



UNIVERZITA MATEJA BELA
Fakulta přírodních věd, Katedra informatiky



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta přírodovědně-humanitní a pedagogická



MEDZINÁRODNÁ KONFERENCIA O VYUČOVANÍ INFORMATIKY

DidInfo 2021

27. ROČNÍK KONFERENCIE

ISBN: 978-80-557-1823-1
ISSN: 2454-051X

BANSKÁ BYSTRICA
2021

Predsedovia programového výboru/General Chairs:

doc. Ing. Jarmila Škrinárová, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
doc. PaedDr. Jiří Vaníček, PhD. *Jihočeská Univerzita, České Budějovice, CZ*

Programový výbor konferencie/Programm Committee:

doc. RNDr. Gabriela Andrejková, CSc. *Univerzita Pavla Jozefa Šafárika, Košice, SK*
Sonya Armstrong, PhD. *West Virginia State University, USA*
Mgr. Jan Berki, PhD. *Technická univerzita v Liberci, CZ*
doc. RNDr. Miroslava Černochová, CSc. *Univerzita Karlova, Praha, CZ*
prof. Dr. Valentina Dagiene *Vilnius University, LT*
Ing. Dana Horváthová, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
prof. Dr. Mirjana Ivanovic *University of Novi Sad, SR*
Ing. Jana Jacková, PhD. *Katolícka univerzita, Ružomberok, SK*
prof. RNDr. Ivan Kalaš, PhD. *Univerzita Komenského, Bratislava, SK*
doc. RNDr. Zuzana Kubincová, PhD. *Univerzita Komenského, Bratislava, SK*
doc. RNDr. Gabriela Lovászová, PhD. *Univerzita Konštantína Filozofa, Nitra, SK*
Ing. Božena Mannová, PhD. *České vysoké učení technické, Praha, CZ*
RNDr. Alžbeta Michalíková, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
RNDr. Pavel Pešat, PhD. *Univerzita Pardubice, CZ*
Prof. Dr. Kate Sanders *Rhode Island College, USA*
prof. Ing. Veronika Stoffová, CSc. *Trnavská univerzita v Trnave, SK*
doc. RNDr. Petr Šaloun, PhD. *Univerzita Palackého v Olomouci, CZ*
doc. RNDr. Lubomír Šnajder, PhD. *Univerzita Pavla Jozefa Šafárika, Košice, SK*
doc. RNDr. Pavel Töpfer, CSc. *Univerzita Karlova, Praha, CZ*
doc. Ing. Ľudovít Trajtel', PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
Dr. Livia Tudor *Petroleum-Gas University, RO*
PaedDr. Patrik Voštinár, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*

Organizačný výbor konferencie/Organizing Committee:

Mgr. Adam Dudáš, PhD.
Katarína Gavaldová
Ing. Dana Horváthová, PhD.
RNDr. Alžbeta Michalíková, PhD.
PaedDr. Patrik Voštinár, PhD.

Recenzenti/Reviewers:

doc. RNDr. Gabriela Andrejková, CSc. *Univerzita Pavla Jozefa Šafárika, Košice, SK*
Mgr. Jan Berki, PhD. *Technická Univerzita v Liberci, CZ*
doc. RNDr. Miroslava Černochová, CSc. *Univerzita Karlova, Praha, CZ*
Mgr. Adam Dudáš, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
Ing. Dana Horváthová, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
Ing. Jana Jacková, PhD. *Katolícka univerzita, Ružomberok, SK*
doc. RNDr. Ľudmila Jašková, PhD. *Univerzita Komenského, Bratislava, SK*
prof. RNDr. Ivan Kalaš, PhD. *Univerzita Komenského, Bratislava, SK*
Mgr. Nika Klimová. *Univerzita Konštantína Filozofa, Nitra, SK*
doc. RNDr. Zuzana Kubincová, PhD. *Univerzita Komenského, Bratislava, SK*
doc. RNDr. Gabriela Lovászová. PhD. *Univerzita Konštantína Filozofa, Nitra, SK*
Ing. Janka Majherová, PhD. *Centrum vedecko-technických informácií, Liptovský Mikuláš, SK*
Ing. Božena Mannová, PhD. *České vysoké učení technické, Praha, CZ*
RNDr. Miroslav Melicherčík, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
RNDr. Alžbeta Michalíková, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
RNDr. Pavel Pešat, PhD. *Technická Univerzita, Liberec, CZ*
PaedDr., Mgr. Vladimír Siládi, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
prof. Ing. Veronika Stoffová, CSc. *Trnavská univerzita v Trnave, SK*
doc. RNDr. Petr Šaloun, PhD. *Univerzita Palackého v Olomouci, CZ*
doc. Ing. Jarmila Škrinárová, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
doc. RNDr. Lubomír Šnajder, PhD. *Univerzita Pavla Jozefa Šafárika, Košice, SK*

doc. RNDr. Pavel Töpfer, CSc. *Univerzita Karlova, Praha, CZ*
doc. Ing. Ľudovít Trajtel', PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
Mgr. Michal Vagač, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*
doc. PaedDr. Jiří Vaniček, PhD. *Jihočeská Univerzita, České Budějovice, CZ*
PaedDr. Patrik Voštinár, PhD. *Univerzita Mateja Bela, Banská Bystrica, SK*

Editori/Editors: Ing. Dana Horváthová, PhD., RNDr. Alžbeta Michalíková, PhD.,
doc. Ing. Jarmila Škrinárová, PhD., PaedDr. Patrik Voštinár, PhD.
Copyright © 2021 autori/authors
ISBN: 978-80-557-1823-1
ISSN: 2454-051X
Recenzovaný zborník/Reviewed Proceedings

Tento zborník vznikol s podporou nasledovného projektu a fondov:

Projekt IT Akadémia IT AKADÉMIA - vzdelávanie pre 21. storočie, ktorý sa realizuje vďaka podpore z Európskeho sociálneho fondu a Európskeho fondu regionálneho rozvoja v rámci Operačného programu Ľudské zdroje, ITMS kód projektu: 312011F057



Záštita nad konferenciou

27. ročník konferencie DidInfo 2021 sa uskutočňuje pod záštitou predsedu Banskobystrického samosprávneho kraja Ing. Jána Luntera.





RNDr. Peter Tomcsányi, PhD.

Práve cez stratu takýchto ľudí, vedľa ktorých roky fungujeme, si uvedomujeme ako vážna je doba, ktorú žijeme...

Zvykli sme si v rámci konferencie Didinfo na jeho každoročnú prítomnosť. Vždy ju obohatil svojimi nenápadnými, skromnými, ale hodnotnými a presvedčivými názormi. Vážime si jeho prínos pre rozvoj vyučovania informatiky na Slovensku i vo svete. Jeho aktivity, ktoré zakaždým prezentoval, poukazujú na to, ako mu záležalo na kvalite vyučovania informatiky na Slovensku. Bol významným učiteľom, ktorý svojimi nemalými nárokmi a očakávaniami posúval svojich študentov ďalej v ich vedomostiach a osobným príkladom ich motivoval k vzdelávaniu. Za mimoriadne významný považujeme jeho prínos do slovenského a svetového hnutia iBobor, ktorého bol dušou a hybným motorom, ako aj do ďalších projektov ako Comenius Logo, Imagine Logo a pod.

Tento zborník je venovaný práve jemu.

Česť jeho pamiatke

Predhovor

Stretávame sa online, no v podstate na pôde Fakulty prírodných vied Univerzity Mateja Bela v Banskej Bystrici na 27. ročníku konferencie DidInfo 2021. Konferencia je už tradične zameraná na vyučovanie informatiky.

Pôvodne slovenská konferencia, neskôr s účastníkmi z Českej republiky sa od roku 2018 pretransformovala na konferenciu, ktorá sa koná každoročne, raz na Slovensku a raz v Českej republike. Je mi ct'ou privítať všetkých účastníkov. Špeciálne vítam medzi nami slovenské i zahraničné osobnosti, ktoré prijali pozvanie a pripravili veľmi inšpiratívne prednášky: prof. Mirjanu Ivanovic z Univerzity v Novom Sade, Srbsko, prof. Valentine Dagiéne z Vilniusekej University, Litva, prof. Ivana Kalaša z Univerzity Komenského v Bratislave a Dr. Livianu Tudor z Polytechnickej Univerzity v Bukurešti, Rumunsko.

Za hlavný cieľ konferencie môžeme považovať rozvoj osobností žiaka a učiteľa v predmete Informatika s presahom do rozvoja algoritmickeho alebo výpočtového myslenia, ktoré ovplyvňuje schopnosť riešiť úlohy z iných odborov. Tento cieľ vnímame z mnohých uhlov pohľadu: vychádzame z moderných technologických trendov, pracujeme v kontexte metodických postupov pri tvorbe algoritmov, ktorý využívame pri analýze, návrhu, tvorbe a overovaní úloh a rôznych informatických aktivít.

Ak sa pozrieme na program tohtoročnej konferencie detailnejšie, uvidíme cenné príspevky zo škôl, ktorých autori vytvorili príklady, aktivity, metódy a softvéry, ktoré súvisia najmä s programovaním. Autori tieto aktivity často zhmotnili do rôznych foriem kartičiek, pracovných listov a metodík, obsahujúcich problémovo orientované a gradujúce úlohy. Úlohy obsahujú prvky tvorivosti a majú kognitívnu náročnosť. Sú koncipované s cieľom primeranou formou začleniť do výučby programovania strojové učenie, umelú inteligenciu, internet vecí, prácu s inteligentnými senzormi, robotickými stavebnicami, mobilnými, webovými a cloudovými aplikáciami. Úlohy majú aj svoj rozmer a aktuálnou požiadavkou je rozvoj myslenia so schopnosťou vnímať úlohu komplexne a dokázať ju rozčleniť na menšie ľahko riešiteľné a aplikovateľné časti.

Rovnako je, pri rozvoji ľudskej osobnosti, dôležitý aj etický rozmer a rovnako aj celospoločenská diskusia v tejto oblasti. Práca študenta a učiteľa informatiky si vyžaduje znalosti z oblasti autorského zákona a zákona o ochrane osobných údajov. To sa prejavuje zručnosťami pri tvorivom písaní odborného textu, ale aj v návrhu, implementácii a používaní nástrojov pre odhaľovanie plagiátorstva.

Už tradične sa autori príspevkov konferencie venujú príprave a organizovaní programátorských súťaží, pričom hlavný dôraz je kladený na rozvoj algoritmickeho myslenia. Súťaže majú svoje nezastupiteľné miesto, kde vzájomné porovnávanie riešení s inými žiakmi pôsobí motivačne. Mimoriadne dôležité sú analýzy súťažných úloh z rôznych pohľadov.

Tradičným lákadlom tejto konferencie sú workshopy. Tento rok sme pripravili 3 workshopy, ktoré sú zamerané na programovanie mikrokontroléra Arduino, programovanie 2D hier v Makecode Arcade a základy práce s micro:bitom.

Doba, ktorú aktuálne žijeme, nám priniesla COVID a izoláciu, ktorá nám neumožňuje poriadne pracovať, byť s deťmi, kvôli ktorým radi robíme svoju prácu a na ktorých môžeme vidieť výsledky nášho snaženia v praxi. Nemôžeme byť fyzicky spolu, radiť sa, diskutovať, riadne prejavovať spolupatričnosť s komunitou, ktorá túto konferenciu po desaťročia vytvára. Preto prinášame dobré príklady ako nám digitálne technológie pomáhajú a otvárajú nové výzvy ako vyučovať informatiku a stretnúť sa na konferencii online.

V ďalšom roku budeme premýšľať, ako značku tejto konferencie ešte posilniť, aby bola atraktívnejšia pre českých a slovenských didaktikov a viac ju prepojiť s praxou učiteľov v teréne.

V týchto časoch je najdôležitejší ľudský prístup, vzájomné porozumenie a tichá spomienka na tých, ktorí nám túto komunitu pomáhali formovať a ktorých nám táto doba zobrala. Preto, na znak úcty, túto konferenciu venujeme ako spomienku na RNDr. Petra Tomcsányiho, PhD.

Želáme konferencii naplnenie stanovených cieľov, aktívnym účastníkom úspech a množstvo inšpirácie do ďalšieho a najmä lepšieho roka!

Marec 2021

doc. Ing. Jarmila Škrinárová, PhD.
doc. PaedDr. Jiří Vaníček, PhD.

predsedovia programového výboru
konferencie DidInfo 2021

Obsah

Pozvané prednášky

Valentina Dagiené	
Informatics Concepts in “Mind-size Bites” for Primary Education.....	11
Liviana Tudor	
Educational Affordances of the Internet of Things Based on Cloud Computing.....	11
Mirjana Ivanovic, Zoran Budimac	
On Use of E-learning 2.0 Collaboration Tools within Teamwork in Delivering Software Engineering Courses.....	12
Ivan Kalaš	
Informatika na križovatke.....	16

Príspevky

Gabriela Andrejková, Mária Baxová, Slavka Blichová	
ICDL/ECDL vo vzdelávaní žiakov stredných škôl.....	26
Daniela Bezáková, Andrea Hrušecká, Roman Hrušecký	
Cestičky – programovanie pre predškolákov.....	34
Andrej Blaho, Ivan Kalaš, Milan Moravčík	
Programovací jazyk v prostredí Emil pre 4. ročník.....	39
Miroslav Blšák, Dana Horváthová	
Mobilná aplikácia na interaktívne vychádzky.....	50
Lucia Budinská, Michal Winczer	
Porovnanie abstraktných úloh v stredoškolských kategóriách súťaže iBobor.....	55
Mária Čujdíková	
Rozvoj informatického myslenia pri hraní videohier.....	59
Václav Dobiáš	
Vztah učiteľů mateřských škol k využívání digitálních technologií při přímé pedagogické činnosti.....	64
Ján Guniš, Ľubomír Šnajder	
Obrátená výučba programovania na vysokej škole.....	69
Jozef Hvorecký	
Učebné materiály podporujúce humor a zábavu.....	75
Hana Hyksová	
Programování robotů na základní škole.....	81
Jana Jacková, Dušan Mráz, Patrik Benyak	
Príprava na medzinárodný benchmarking vo vyučovaní úvodných kurzov programovania v jazyku Python.....	86

<p>Ľudmila Jašková</p> <p>Súťaž iBobor pre nevidiacich žiakov nižšieho sekundárneho vzdelávania: Vedomosti a zručnosti potrebné na riešenie úlohy ako dôležitý faktor ovplyvňujúci jej obťažnosť.....</p>	92
<p>Alžbeta Kanáliková</p> <p>Čo sa skrýva v dátach? Dátová analýza a prezentácia dát (metodika pre SŠ).....</p>	97
<p>Mária Karpielová</p> <p>Faktory zvyšujúce žiacky záujem o programovanie na kurze tvorby počítačových hier.....</p>	102
<p>Nika Klimová, Gabriela Lovászová</p> <p>Tvorivosť v kontexte projektového vyučovania u budúcich učiteľov informatiky.....</p>	107
<p>Jaroslav Knežník, Jarmila Škrinárová, Patrik Voštinár</p> <p>Mobilná aplikácia s umelou inteligenciou pre deti.....</p>	113
<p>Karolína Miková, Andrea Hrušecká</p> <p>Identifikácia gradácie informatických konštruktov pri vyučovaní edukačnej robotiky na 1. stupni ZŠ.....</p>	118
<p>Karolína Miková, Lucia Budinská, Barbora Stenová</p> <p>Analýza edukačných robotických hračiek dostupných na Slovensku.....</p>	123
<p>Marek Nagy</p> <p>Dištančné vzdelávanie GNU/Linux zručností pomocou modulu Multimediálnej čítanky.....</p>	129
<p>Pavol Polonec, Soňa Úradníková</p> <p>Vybrané materiály využívané pri výučbe programovania na ZŠ Spojová.....</p>	134
<p>Ľubo Salanci</p> <p>Porovnávacie štúdiá pravdepodobnostného prístupu a digitálneho vodoznaku pri odhaľovaní plagiátorstva v dištančnom kurze programovania.....</p>	139
<p>Vernika Stoffová, Ján Stoffa</p> <p>Terminologická príprava budúcich edukátorov - Terminological training of future educators.....</p>	145
<p>Václav Šimandl</p> <p>Synchronní distanční výuka informatiky s využitím problémově orientovaných úloh.....</p>	152
<p>Jiří Vaníček, Jan Berki</p> <p>Modelové školní vzdělávací programy pro novou informatiku v Česku.....</p>	158
<p>Patrik Voštinár, Miroslav Melicherčík</p> <p>Keď učiteľ vyrastie na žiaka – vzdelávanie učiteľov ZŠ a SŠ v regióne Banskej Bystrice.....</p>	162
<p>Patrik Voštinár, Dana Horváthová, Ľudovít Trajtel', Jarmila Škrinárová, Alžbeta Michalíková</p> <p>Korona - akcelerátor informatizácie výučby.....</p>	168
<p>Martin Zboran</p> <p>Programovanie robotov vo virtuálnom prostredí.....</p>	173

Krátke príspevky

Silvia Haringová, Nika Klimová

Ukážky úloh z informatických prechádzok.....180

Klára Horváthová

Kognitívna náročnosť úloh v súťaži iBobor.....182

Marek Mansell, Nika Klimová, Martin Cápaj, Magdaléna Bellayová

BBC micro:bit ako súčasť projektu ENTER.....184

Workshopy

Richard Balogh

Základy práce s micro:bitom.....187

Patrik Voštinár, Vladimír Siládi, Michal Vagač

Programovanie mikrokontroléra Arduino.....187

Patrik Voštinár, Jarmila Škrinárová

Programovanie 2D hier v MakeCode Arcade.....187

Organizátori

Sponzori a mediálni partneri

Pozvané přednášky

Invited papers

Informatics Concepts in “Mind-Size Bites” for Primary Education

Valentina Dagiene

Institute of Data Science and Digital Technologies,
Vilnius University,
Vilnius, Lithuania
valentina.dagiene@mii.vu.lt

ABSTRACT

Main goal of informatics at school is to teach how to think – to solve problems by using different ways: with a computer as well as without a computer. Thinking computationally draws on the concepts that are fundamental to computer science, and involves systematically and efficiently processing information held in the tasks. Computational thinking involves defining, understanding, and solving problems, reasoning at multiple levels of abstraction, understanding and applying automation, and analysing the appropriateness of the abstractions.

There are many ways for selecting tasks to be solved by pupils in the classroom. For primary education two types of problem solving are usually declared: (1) hands-on tasks that take more time and cover several topics, and (2) everyday exercises, that are very common in mathematics and grammar lessons. We suggest a third type of problems, the short tasks (as “mind-size bites” according to

S. Papert) with double folded aim: to cover informatics concepts and to be solvable in few minutes.

Solving short informatics tasks is a powerful method that can support a pedagogical shift in the classroom and foster pupils’ engagement and motivation to learn. Problem solving of short tasks can be considered as a systematic process involving pupils into deeper understanding of informatics concepts. Solving short tasks can be one of the strategies that engage and motivates pupils for deeper learning and fosters the deeper thinking skills.

The worldwide Bebras challenge on Informatics and CT is discussed as an example of connecting formal and non-formal informatics education by using thousands of short tasks based on informatics concepts and applying problem solving strategies. The Bebras activities is aimed to promote pupils’ interest in informatics at the early stage of the school education and to motivate pupils to learn deep and master better technology.

Educational affordances of the Internet of Things based on Cloud computing

Liviana Tudor

Politehnica University of Bucharest,
Petroleum-Gas University of Ploiesti,
Romania
tudor.liviana@gmail.com

ABSTRACT

New trends toward the digitization of society focus on the widespread adoption of the Internet of Things and the Industrial Internet of Things. Everywhere there are intelligent sensor-based devices that process data with or without human help. The Internet of Things consists of a multi-layered system having hardware components that transfer data through the Internet. The data are analyzed and transformed using analytical tools or machine learning algorithms.

We will discuss the architecture of the Internet of Things based on the cloud, the connection with industry and education, and examples of successful implementations. We will analyze the integration of cloud services into the architecture of an IoT. Whilst IoT networks use a gateway to gather data and communicate with the cloud, the cloud performs computation and data storage, real-time analytics, machine learning and visualization. We will also present the concepts and technologies involved in Cloud

Computing. In this context, it is worth considering the question of whether an IoT system without cloud components can be reliable.

Some challenges of including IoT and Cloud Computing topics in the Curricula will also be presented. Our discussion will focus on a target group of online courses on IoT and Cloud Computing from universities in the USA and Google company and will consider some existing courses in the European universities. We will follow and discuss the guidelines of some reference papers from the ACM and IEEE databases.

On Use of E-learning 2.0 Collaboration Tools within Teamwork in Delivering Software Engineering Courses

Mirjana Ivanovic
University of Novi Sad
Faculty of Sciences
Trg D. Obradovica 4
21000 Novi Sad, Serbia
mira@dmi.uns.ac.rs

Zoran Budimac
University of Novi Sad
Faculty of Sciences
Trg D. Obradovica 4
21000 Novi Sad, Serbia
zjb@dmi.uns.ac.rs

ABSTRACT

Significant advancements in e-learning have been achieved by use of the Web 2.0 technologies. These technologies converted a passive learner into an active participant in the learning process. Also, the classroom has been transformed and it is not only socially but collaboratively constructed by using different social web tools. Such collaborative ways of learning where communication is multi-directional, and knowledge may be socially constructed, is challenging methodological aspect.

In this paper we briefly report on our long-lasting experience regarding the implementation of Web 2.0 concept in two Software Engineering courses. In particular, this is emphasized in encouragement of students' teamwork, through use of Wiki technology. Attention is also paid to workload, students' feedback and benefits, and potential disadvantages of such approach.

Keywords

E-learning 2.0. Wikis. Teamwork. Software Engineering Courses.

1 INTRODUCTION

Latest research estimates [1] that Internet penetration rate worldwide in total is 63.2%, predominantly in Northern America (89.9%), Europe (87.1%), Latin America/Caribbean (72.6%) decreasing to the least in Africa (46.7%).

Accordingly, Internet search and information retrieval is getting prevalent since early 1990s.

Evidently, the phenomena that significantly influenced last decade is social networking and use of social media, followed also by collaborative content creation. Educators all over the world recognised significance of such technological resources and have been trying to discover could education benefit from such situation. Our opinion is that the situation is definitively affirmative.

Concept of e-learning 2.0 as the unification of e-learning and social networking representing Web 2.0, showed as rather promising methodological approach. Additionally, with Web transformation from 1.0 to 3.0, e-learning has been strongly affected by new ways of thinking. The current situation is that all over the world a lot of universities deploy educational social software in everyday practice. It definitively enhances some important aspects of teaching and learning and significantly influences the creation of new forms of educational activities. One of main advantages of use of social software in everyday educational practice is to allow students to actively collaborate in solving particular problems and tasks [2, 3].

Technologies made a significantly shift from one traditional way of living to another and users in the 21st century are not only passive consumers but active contributors ready to find, share and evolve

information. Web 2.0 and succeeding Web 3.0 encompass forms of social, communication, information sharing and corporative tendencies.

In this paper we will briefly report on our long-lasting experience regarding the implementation of Web 2.0 concept in two Software Engineering courses delivered at our Faculty. In particular, this is emphasized in encouragement of students' teamwork, through use of Wiki technology.

For this purpose, we have been using MOODLE LMS as it offers different functionalities. Wikis are implemented in MOODLE in quite a straightforward way, students learn to use this module rather quickly and soon get familiar with it.

The rest of paper is organized as follows. Second and third sections presents our experiences in delivering Computer Ethics and Software Engineering courses, respectively. Section four gives brief overview of possibilities that e-learning 3.0 offers to future educational processes. The last section concludes the paper.

2 COMPUTER ETHICS RELATED COURSE

At our Faculty we have been delivering Computer Ethics related course for more than 15 years. From the beginning we introduced student essays as a part of exam and at that time essays were prepared either individual or by two students. Soon after several years of course existence and significantly increasing number of enrolled students we switched at collaborative preparation of joint essays. During the course delivery traditional face-to-face discussions directed by the teacher were intensively applied. However, students' feedback was rather poor (shyness, unpreparedness for the topic, and so on). The crucial methodological shift happened with decision to direct course towards cooperative and collaborative learning and building a common knowledge using social tools. Consequently, student activities were divided into two types:

- b) Discussion forums - cover a particular case study connected with a topic.
- c) Wikis - cover a broad topic divided into related articles.

Discussion forums - initially were based on role-playing games technique. Students were given certain roles that reflected different viewpoints of the same ethical problem. On-line discussion forums were particularly challenging for students. In fact, the shy and silent students found themselves very involved in discussions. Forums become more frequent form within the course offering several topics for selection. In such a way created social networks inspired widening of topics. Very often discussions diverged into different directions interesting for students.

Wikis – after positive experiences with discussion forums it was challenging for both participants (teacher and students) to try with Wikis (two types: independent and discussion based). Wikis were accepted as regular activity for most of the students. Independent Wikis served to enable the presentation of a study of one or several different aspects of a topic. Number of students per topic was up to ten and they were grouped according to their preferences. Each team member could actively contribute only to the creation of the chosen topic. However, view of other Wikis was welcomed and encouraged. Students intensively used this opportunity as means of improvements of their text.

Two key differences between independent and discussion-based Wikis were: 1) Each group in discussion-based Wikis are considered as a compact team. The moderator was responsible for the common Wiki. 2) Discussion-based Wikis were supported by parallel discussion forums devoted to the preparation of joint Wikis. “If the use of news forum, and consequently the exchange of information dealing with the final product were sporadic, parallel forums enabled incomparably greater information sharing” [3]. For example, for the creation of the Wiki on piracy, team members of one particular team accessed the forum 625 times, i.e. 41.67 times per participant in average. For final version of the Wiki total number of accesses was 3337, i.e. 37.08 per participant in average.

Collaborative creation of joint outcomes was encouraged as well. To collaborative projects a moderator was assigned and usually it was the best student in the team. Significant number of our students are part-time students and they formed less homogenous groups. In such groups the moderator was the most enthusiastic student who opened the supporting forum. In some cases, the moderator was appointed by the teacher. Initially, each moderator proposed the contents of the essay. During given time interval for preparation of preliminary version the content was created by inserting chunks of text produced by many members of a team.

After that in short period of time (several days), preliminary version was improved and polished. Individual contributions of students were focused on smaller topic, however the amount of individual contribution in the final outcome was rather balanced (7.08 to 11.47 posts per student). The quality of Wikis principally depended on the students who created it.

We also try to include in course chats and blogs however students were not enthusiastic about them.

Workload and students’ impressions - Our experience is that enormous effort of all participants (students and teacher) is needed to achieve Web 2.0 content presented in appropriate manner and of good quality. Students’ activities usually reached their peak at the beginning and at the end of assigned task and during the discussion forums. Second peak is connected to preparation of individual Wikis and students usually behave chaotic and in a hurry. The last activity was discussion-based Wiki (end of Semester) and students generally were less active. The main reason was that these Wikis have been planned in advance and moderators were better organized. For example, during one of semesters average number of activities per student was 675.20 i.e. in average 40.72 times per student weekly. Altogether, they submitted on average 325.67 posts weekly i.e. 55.51 per student. After all we can say that students’ satisfaction was positive. In Figure 1 we present students’ impression regarding the quantity of gained knowledge and different Web 2.0 elements for one typical semester.

Positive effects of this innovative methodological approach are multiple: mutual collaboration, increased awareness, minimized plagiarism, and also satisfaction with adoption of Web 2.0.

I learned the most from ...		Final impression about Web 2.0	
Lectures	72.58%	Unacceptable	0.00%
Individual essays	40.32%	Expected something else	1.61%
Independent discussion forums	33.87%	Partial approval	20.97%
Independent wikis	35.48%	Approval with minor remarks	37.10%
Collaborative creation of joint outcomes	41.94%	Greatest approval	40.32%

Figure 1: Student impression regarding the quality of gained knowledge and Web 2.0 elements

Additionally, use of discussion forums was motivational for students to reveal their own ideas, and it happened that introvert students demonstrate extroversion, students with speech disorders were unsurprisingly active on-line and so on.

However, we noticed some disadvantages: definitely the critical challenge was total transparency i.e. possible threat to privacy and openly revealed personal opinions different from others, withdraw the post soon after publishing it, and so on. These social networking activities were rather exhaustive for the teacher (check the activities several times a day, actively participate and guide actions when it was necessary).

3 SOFTWARE ENGINEERING COURSE - WIKI AS A TOOL FOR STUDENTS’ TEAM ASSIGNMENTS

In this section we will present our experiences gained in delivering a more than a decade course on “Software Engineering”, also with particular attention on Wikis. During that period, we insisted on project-based, team-based, and Wiki-based assignments. Using data collected in Wiki history log we have been assessing individual participation of team members in solving assessments. Use of Wikis in all assignments allowed us to better understand students’ behaviour and their attitudes towards work organization and schedule. Being fourth year students the majority of them had no problem in using Wikis. Also, team members were able to see only their own teams’ solution, i.e. we forbid them to see and copy solution of some other team. Unfortunately, our analysis (based on individual students logs) showed that a significant number of students postponed their work as much as possible and performed their obligations in last few days before the submission deadline.

The topics of 6 obligatory team assignments slightly vary from year to year but they are remarkably similar. For each assignment 2-3 weeks, depending on complexity, are given for finishing them. Additionally, each student is obliged to reach at least 50% of the total number of points per assignment. As self-selected teams seemed as a good practice at the beginning, lately as assignments progressed different problems raised. To solve this, we done selection of the team members randomly by LMS Moodle. The teams usually consisted of 4 (rarely 5) members. Our intention also was to simulate team’s real-life functioning. So once at the semester we allowed them to fire one among the team members if all the other members agree to do it.

As a main orientation of Software Engineering course is to prepare students to the real-life working environments all assignments are created with that intention.

Assignment 1: A realistic “requirements specification” document with a list of software functionalities is prepared by teacher. As in real situations the specification contains a number of mistakes and misunderstandings, some requirements are ambiguous, some unclear. Students are required to recognize the problems, cleared up ambiguous specifications and finally produce much more precise document.

Assignment 2: Use “Function point method” for software cost estimation for the requirement specifications from assignment 1. Teams should estimate needed man-month software implementation and what is the optimal size of the development team. They are required to prepare Wiki document with reasons and explanations of their decisions.

Assignments 3-6: They are of the same type as first two, concentrated on: Formal specifications, Software metrics, and Analysis/creation of structural and OO design.

Some general observations/conclusions of students’ behaviour were noticed:

- The least ambitious students would stop participating in solving assignment as soon as they reach minimum points to pass. A slightly more ambitious students usually completed one more assignment for a little bit higher grade. So, several last assignments were usually solved by only one or two the most ambitious students within team.

- Use of Wiki did not stop live team meetings, we realized that students jointly worked and discussed assignments, created solution and after that entered text into Wiki including individual opinions also. As deadline for submission of the assignment solution was approaching additional discussions and even quarrels were visible in the text. Teachers encouraged such behaviour.

- Results are generally rather disappointing. Students used the minimal time at the beginning of the assignment, postponing the work as long as possible, and performing majority of activities within last several days. Also, more than 80% of the work was done for the last 3 assignments.

- The only thing students liked is collaborative nature of the assignments which led to the production of much better solutions than their individual solutions.

- Students felt that in almost every team there were persons who did not work enough but receiving the higher grades.

In the rest of the section, we will present some numerical data for a typical school year. We can notice that number of posts and views on course/Wiki activities, has peaks short period before the assignments’ submission (Figure 2).

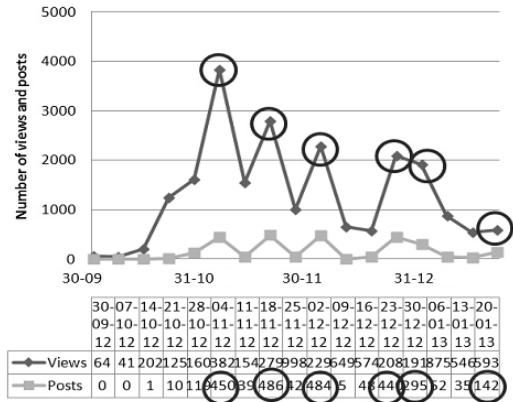


Figure 2: Peak times for assignments

Also, we can have a look into the Wiki assignments’ activities. In the Figure 3 we present for each of ten teams number of edits for assignment 1 (it lasted 17 days).

	Assignment 1									
	Team 1	Team 2	Team 3	Team 4	Team 5	Team 6	Team 7	Team 8	Team 9	Team 10
1 19 10				3		5		6		
2 20 10			1	6		2		1	3	
3 21 10	1		10	3						
4 22 10			7			3				
5 23 10	4		1							
6 24 10						11	5	1		
7 25 10	1			3			3			1
8 26 10	7		1		2	7	10	6		
9 27 10	50	2	17		4	1	9	1	2	2
10 28 10	17		12	6	14	3	6		1	
11 29 10	10		11	2	43	13	3	2		2
12 30 10	6		23	3	44	15	8		2	2
13 31 10	9	11	27	5	25			13		
14 1 11	6	47	10	5	28	15	7	34	19	
15 2 11		10	28	10	18	5		10	45	4
16 3 11	14	21	25	22	20	2	4	11	11	2
17 4 11	1	22	42	19	34	2	11	16	7	38
	126	113	213	87	230	84	66	101	90	51

Figure 3: Assignment 1: Number of edits per team per day

Number of edits is in interval (51, 230). We can notice absence or low activities during early days for assignment solving. As we expected teams were significantly active within the last several days. Figure 4 gives the summary of total numbers and percentages of edits for all teams for Assignment 1.

Day	1	2	3	4	5	6	7	8	9
Edits	14	13	14	10	5	17	8	33	88
%	1.21	1.12	0.86	0.43	1.46	0.69	2.84	7.58	5.08
Day	10	11	12	13	14	15	16	17	All
Edits	59	86	103	90	169	128	132	192	1161
%	5.08	7.41	8.87	7.75	14.6	11.02	11.37	16.54	100

Figure 4: Total number of edits of all teams per day

For this particular example the solution of all assignments required 6672 edits and posts within the Wikis, i.e. average 159 per student. Also, some of students liked to see behaviour of their teammates and we registered additional 5864 views i.e. average 139 per student. Looking at the assignments, the situation is: students needed 26 posts; as a team they needed 106 posts and edits and similar number of viewing on other colleagues’ work - 93.

However, we were aware that for students it was significant to investigate good solutions of other teams we tried to incorporate this methodological aspect as well. We organized at least one class

for presentation of result of “the most provocative/challenging” solution which was not always “the best one”. Students were encouraged to actively discuss, comment, criticise, compare presented with their solution. This gave a chance to students to identify and understand errors of other teams.

4 E-LEARNING 3.0 AND FUTURE EDUCATIONAL PROCESSES

Technological development highly has been influencing further advancements in e-learning and educational processes. Appearance of Web 3.0 technologies and their spreading in wide range of domains and applications caused appearance of e-learning 3.0 as promising educational concept [4]. It is getting more and more popular and challenging in support of future educational processes. However, always there is dilemma is e-learning 3.0 just another buzz word or really new promising educational facilitator. Interesting issues about future of e-learning 3.0 are presented in [5]. Several vital questions of possibilities that e-Learning 3.0 can offer in educational processes are considered: “(1) How Web 3.0 technologies will be incorporated in the existing e-Learning 2.0? (2) Is there a need for a new learning theory for e-Learning 3.0? (3) What are the challenges and issues related to adoption of e-Learning 3.0? “.

In fact Web 3.0 technologies offer significant features that highly influence development of modern educational environments and ecosystems, first of all based on intelligent behaviour and personalized educational actions [6, 7]: Intelligent/semantic Web; 3D Virtualization (personalised avatars); Multi gesture devices and 3D touch user interfaces; AI and machine learning; Socio-Semantic Web, Distributed and Cloud Computing; Openness and interoperability; Control of information; Management of Big Data, and Linked data.

Still e-learning 3.0 systems are not commercially widespread. On the other hand, educational researcher community intensively proposing and implementing such systems as working prototypes and proof-of-concepts.

5 CONCLUSIONS

In this paper our intention was to give a brief overview of our experiences in applying some of e-learning 2.0 capacities (first of all Wikis) and present some essential aspects that e-learning 3.0 can offer in future educational activities [5].

First, we presented some experiences we gained during a long period of delivering two software engineering courses at our Faculty. Currently, use of forums, Wikis, blogs, and instant messages, became a normal practice at a lot of our courses. We presented students’ reactions to the new tool and learning practices it imposes. Also, we analysed students’ behaviour and problems they faced in realization of teamwork, problem-solving approach applying Wikis in finishing assignments. Nevertheless, most of the students were satisfied with the advanced transparency and fairness

of the grading process, as well as with their newly developed competences for successful work in teams.

It is interesting that we have less problems with each new generation of students and seems that they are already prepared and knowledgeable concerning educational technologies.

Definitely we should continue to apply different technological achievements in everyday educational practice. E-learning systems ultimately have been evolving with the evolution of the new technologies and the Web. However, use of them, Web 3.0 and e-learning brings plethora of technological, social, ethical, and legal challenges.

ACKNOWLEDGMENTS

Authors acknowledge financial support of the Ministry of Education, Science and Technological Development of the Republic of Serbia (Grant No. 451-03-9/2021-14/ 200125).

REFERENCES

- [1] World internet stats—Usage and population statistics. (2020). Retrieved from <https://www.internetworldstats.com/stats.htm>.
- [2] BRATIĆ, B., TRIGLIANOS, V., KURBALIJA, V. PAUTASSO C., IVANOVIĆ M. Role of interactive presentation platform ASQ in delivering web design course. In *Smart Learn. Environ.* 2020, vol. 7, paper 15. <https://doi.org/10.1186/s40561-020-00123-w>
- [3] ZDRAVKOVA K., IVANOVIĆ M., PUTNIK Z. Experience of Integrating Web 2.0 Technologies. In *Educational Technology Research and Development*. 2012, vol. 60, no. 2 p. 361-381.
- [4] REGO, H. AHKME eLearning Information System: A 3.0 Approach. Unpublished PhD thesis at University of Salamanca, 2011
- [5] SHALTOU M. S. A., SALAMAH A. I. B. The Impact of Web 3.0 on E-Learning. In *Proceedings of Fourth International Conference on e-Learning "Best Practices in Management, Design and Development of e-Courses: Standards of Excellence and Creativity"*. 2013, p. 227-232, doi: 10.1109/ECONF.2013.70.
- [6] FEHMIDA H. E-Learning 3.0 = E-Learning 2.0 + Web 3.0? In *IOSR Journal of Research & Method in Education*. 2013, vol. 3, no. 3, p. 39-47, p-ISSN: 2320-737X
- [7] EBNER, M., SCHÖN, S., TARAGHI, B., DRACHSLER, H., TSANG, P. First steps towards an integration of a Personal Learning Environment at university level. In: Kwan R., McNaught C., Tsang P., Wang F.L., Li K.C. (eds) *Enhancing Learning Through Technology. Education Unplugged: Mobile Technologies and Web 2.0. ICT 2011. Communications in Computer and Information Science, vol 177.* Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-22383-9_3

Informatika na križovatke. Môj názor

Informatics education at the crossroad. Position paper

Ivan Kalaš

KDMFI FMFI

Univerzita Komenského v Bratislave

ivan.kalas@fmph.uniba.sk

Tento príspevok venujem Peťovi

ABSTRACT

The 2020 and 2021 experience seem to have definitively put an end to discussions about the need to develop digital literacy for pupils and teachers (and, after all, parents). What remains as a truly relevant question is how and where to develop it with pupils and (future) teachers. However, this is only one - although important - topic that must be of interest to our community. It is high time also to discuss openly another problematic aspect of informatics education (certainly not only in our country). Where am I pointing? Each graduate of our primary school has completed six years of compulsory informatics, each graduate of a general education secondary school three more. Are we satisfied with the outcome? To illustrate what I mean I will use an excerpt from the text of a student of our faculty:

... When I think of all the wasted time in schools, my heart aches. ... the hours of informatics that we spent either watching some videos or drawing, I am sick of it. And every time over and over from the beginning. I cannot say I have moved one step forward from the very basics of informatics, not a single step.

I perceive this situation as a crossroads on which we stand. In my keynote, I want to analyse the current situation and propose a discussion on how to react. I believe we as a community will be able to find an escape from this vicious roundabout, from the lost zone of inefficiency and eternal beginnings.

Keywords

computing in school, problems with computing education, digital literacy and computing, programming for all

ABSTRAKT

Skúsenosť z rokov 2020 a 21 zrejme definitívne ukončila úvahy o potrebe rozvíjať digitálnu gramotnosť žiakov, učiteľov, a koniec-koncov, aj rodičov. Naďalej ale zostáva aktuálnou otázka, ako a kde sa má u žiakov a (budúcich) učiteľov rozvíjať. To je však iba jedna – aj keď významná – téma, ktorá musí našu komunitu zaujímať. Je najvyšší čas tiež otvorene hovoriť aj o druhom problematickom aspekte informatického vzdelávania (určite nielen u nás). Kam mierim? Každý absolvent našej základnej školy má za sebou šesť rokov povinnej informatiky a každý absolvent strednej školy zameranej na všeobecné vzdelávanie ešte ďalšie tri. Sme spokojní s výsledkom? Na ilustráciu si pomôžem úryvkom z vyjadrenia študenta našej fakulty:

... keď si spomeniem na premrhaný čas na školách, tak ma až zabolí pri srdci. ... tie hodiny informatiky, ktoré sme buď strávili pozeraním nejakých videí alebo kreslením si, tak mi je až zle. A zakaždým znova a znova od začiatku. Nemôžem povedať, že som sa za celú základku v informatike niekam posunul, absolútne nie.

Tento stav vnímam ako križovatku, na ktorej stojíme. V mojom príspevku chcem analyzovať súčasný stav a navrhnúť diskusiu o tom, ako ďalej. Verím, že sa nám ako komunite podarí nájsť únik z tohto začarovaného kruhového objazdu, zo stratenej zóny neefektívnosti a večných začiatkov.

Kľúčové slova

školská informatika, problémy v informatickom vzdelávaní, digitálna gramotnosť a informatika, programovanie pre každého

1 ÚVOD

Ďalšou etapou v rozvoji informatického vzdelávania, ktorá sa vo vyspelých vzdelávacích systémoch začala okolo a po roku 2012, je snaha rozlíšiť jeho dve zložky – digitálnu gramotnosť a informatické myslenie. Digitálnou gramotnosťou – obzvlášť učiteľov – sa však nemyslia iba bežné počítačové zručnosti, teda napr. schopnosť učiť svoju triedu s využitím niektorého on-line prostredia. Ide o súbor znalostí v odborovej didaktike a zručností, ktoré učiteľom umožňujú produktívne a bezpečne používať rôzne digitálne technológie (vždy, keď je to vhodné, bezpečné a špecificky prínosné) na podporu ich vzdelávacích cieľov a správneho rozvoja žiakov – na každom predmete. Súčasťou týchto zmien je aj (a) rozšírenie povinnej informatiky ako predmetu do primárneho stupňa vzdelávania¹, (b) efektívna integrácia rozvoja digitálnej gramotnosti (v kontexte odborových didaktík) do prípravy budúcich učiteľov² všetkých predmetov, a z nášho pohľadu najmä (c) snaha venovať sa v predmete informatika skutočne rozvoju informatického myslenia – ako to naďalej správne požaduje náš štátny vzdelávací program.

Súčasťou takejto informatiky je celkom prirodzene aj vytváranie predpokladov a základov pre digitálnu gramotnosť žiakov (špeciálne v prvých ročníkoch). Ja sa však v mojom príspevku sústredím len na vlastnú informatiku a predovšetkým jej programovanie. Je totiž najvyšší čas premýšľať a diskutovať o situácii, v ktorej sa v tomto kontexte nachádzame. V súčasnosti má totiž každý absolvent našej základnej školy za sebou šesť rokov povinnej informatiky a každý absolvent strednej školy zameranej na všeobecné vzdelávanie ešte ďalšie tri. Sme spokojní s výsledkom? Napríklad

¹ U nás máme v takomto rozšírení azda úplné prvenstvo.

² Toto je napr. u nás skutočne dlhodobý problém. Vytýčili sme si ho ako hlavný cieľ už v Projekte Comenius v 90-tych rokoch, a potom opäť v celonárodnom projekte Infovek od roku 1999. Zmeny v príprave učiteľov, ktoré si splnenie tohto cieľa vyžaduje,

nastávajú len pomaly. Skúsenosti z pandémie v rokoch 2020 a 21 zrejme definitívne ukončili úvahy o potrebe takejto gramotnosti. Naďalej ale zostáva aktuálnou otázka, ako a kde sa má u učiteľov rozvíjať. A tiež riziko, že „objav“ on-line vzdelávania zatieni komplexnosť tejto problematiky a podporí jej úzke vnímanie.

a práve v oblasti programovania, presnejšie: riešenia problémov programovaním? Obávam sa, že nie a tento stav vnímam ako výzvu na hľadanie príčin a riešení, ako uniknúť z tejto zacyklovej zóny neefektívnosti a večných začiatkov.

Začnem však tým, že sa pokúsím sformulovať, o akej informatike premýšľam a ako vnímam jej miesto v modernom vzdelávaní. Potom sa zameriam na programovanie a skúsím vyjadriť, ako komplexne ho vnímam – celkom určite nie ako „iba programovanie“. Nebudem kritizovať aktuálne kurikulum, pretože ho naďalej považujem za relatívne moderné a správne interpretovateľné³. To, čo ma skutočne trápi, je priepastný rozdiel medzi tzv. *zamýšľaným* kurikulumom v informatike, kurikulumom *realizovaným* a tým skutočne *dosiahnutým*.

2 O AKEJ INFORMATIKE HOVORÍM

Raz za čas si musíme položiť aj takéto všeobecné otázky a zamyslieť sa nad celým priestorom pre naše snaženie v informatickom vzdelávaní, nad našimi cieľmi, prioritami a orientáciou. Samozrejme, týmto témam sa dá venovať aj *priveľa času*. Ale zdá sa mi, že častejší je práve opačný postoj. Preto – len formou odrážok – pomenujem niekoľko východísk, o ktoré sa opieram. Tento zoznam sa v žiadnom prípade nepokúša identifikovať všetky významné aspekty, na to potrebujeme podstatne širšiu diskusiu.

- Musíme striktné – natoľko, ako sa len dá – rozlišovať rozvoj digitálnej gramotnosti (DG) a vlastné informatické vzdelávanie. Obe strany tejto mince sú pre moderné vzdelávanie významné a užitočné, obe musíme v škole starostlivo implementovať a vyhodnocovať. Ale rozvoj DG nesmie dostať príležitosť prekryvať, či dokonca až nahrádzať informatiku. Namiesto ďalších slov tu položím úmyselne nesprávne a provokatívne sformulovanú otázku: *Kde, ako a kedy sa majú žiaci naučiť Excel?*
- Školskú informatiku musíme správne realizovať *pre každého žiaka v triede, pre každé dievča a každého chlapca*⁴. Musíme tiež rešpektovať individualitu a rozdiely medzi žiakmi. Samozrejme, čím vyššie v ročníkoch ideme, tým výraznejšie nastávajú rôzne diverzifikačné procesy. Žiaci majú k jednotlivým oblastiam vzdelávania rôzne postoje, rôznu vybavenosť a rôzne očakávania. Ale našou povinnosťou je urobiť to najlepšie z najlepšieho pre každého v triede, čo len sa dá. Každý má právo urobiť ten svoj *krok vpred*, a to aj v informatike. A my musíme hľadať cesty, teda obsah a formy⁵, ktoré také príležitosti vytvoria **každému**.
- Aké teda majú byť tieto formy a takýto obsah? Použijem tu obrazy označenie *informatická zvedavosť* (computational curiosity): Škola a vzdelávanie má dávať žiakovi príležitosti vystrojiť sa rôznymi nástrojmi⁶ na skúmanie javov a riešenie problémov tohto sveta, nástrojmi, ktoré ich majú pripraviť na zachraňovanie toho, kam svet dospel a čo sme my – predchádzajúce

generácie – stihli pokaziť. Ale tiež na tvorbu nového, na sebaobjavenie, komunikáciu a potešenie. Informatické vzdelávanie ponúka takéto príležitosti v podobe, aká tu ešte nikdy nebola a ktorú zvyčajne nedokáže nič iné nahradiť. A naopak, žiaden z týchto nových nástrojov (a ciest k nim) sa nesmie ani len pokúšať nahradiť prácu žiakov *s lepidlom, nožnicami a papierom, knihou* a pod. (obrazne myslené).

Informatická zvedavosť je jednou z najnovších príležitostí pristupovať k problémom iným spôsobom, prekvapujúco silným na skúmanie textu, hudby, výtvarna, genetických štruktúr, inžinierskych aktivít, matematických problémov, fyzikálnych, chemických, kozmologických alebo spoločenských javov, problémov ekonomických, klimatických, prírodných... K tomuto pohľadu sa vrátim v časti 6.

- Významnou témou tu je aj vnímanie a pedagogická implementácia *programovania*. Nemôžem súhlasiť s tým, že programovanie alebo rozvoj algoritmického myslenia predstavuje jeden z napríklad piatich tematických okruhov informatiky⁷. Vnímam ho inak, širšie, silnejšie a všeobecnejšie, ale tomuto sa chcem venovať až v časti 3.
- Školská informatika – tým, v akom je stave a aká je ešte stále mladá a pedagogicky „neskúsená“, sa na jednej strane rada skrýva za tieto „vysvetlenia“. Ale na strane druhej nám tým dáva aj **jedinečnú príležitosť prichádzať s inovatívnymi pedagogickými postupmi a obsahom**. V dobrom prípade tak môže inšpirovať aj ďalšie vzdelávacie oblasti a predmety⁸.
- *Dobrá školská informatika* (rovnako ako všetky *dobré* vyučovacie predmety) môže, a teda **musí** byť pre žiakov úžasná, naplňujúca, vzrušujúca, zábavná... Máme v našom tíme milé skúsenosti, ktoré potvrdzujú, že sa to dá⁹ – aj v našich školách, s našimi žiakmi a učiteľmi, v súlade so súčasným kurikulumom. Nemá byť „prvou časťou“ univerzitnej informatiky, aj keď musí byť náročná. Má využiť príležitosť byť jedinečným zážitkom na celý život, a zároveň trvalým a významným posunom v informatickom poznaní, postojoch a postupoch neustále sa rozvíjajúcej digitálnej reality. A aj keď to môže znieť nerealizovateľne a naivne, má byť príležitosťou na humanizáciu nášho života a vzdelávania. Áno, dá sa to, využime ju správne.
- Informatika – a zasa rovnako ako všetky iné predmety, má povinnosť prispieť k celostnému rozvoju žiakov. Ani zďaleka nie iba k informatickému mysleniu a poznaniu. Má obrovský potenciál podieľať sa na rozvoji mnohých ďalších kompetencií. Ako príklad spomeniem tri: (a) *kommunikáciu* – tá tu má mať sama o sebe veľa foriem a funkcií a ponúkať rôzne zaujímavé námety, (b) *kolaboráciu* – práca na informatike musí byť kolaboratívna,

³ To však neznamená, že v nasledujúcich rokoch nenastane chvíľa a príležitosť ho modernizovať. Ale bohužiaľ to znamená aj to, že jeho detailnejšia interpretácia pre učiteľov a metodikov nikdy neexistovala a neexistuje.

⁴ Takáto formulácia by sa v súčasnej anglickej kultúre považovala za nevhodnú, pretože implicitne predpokladá rozdiely medzi dievčatami a chlapcami. Korektná formulácia by bola ... *for every learners in the class*.

⁵ Rešpektujúc, prirodzene, „objednávku“ či zadanie štátu vo forme ŠVP či iŠVP alebo inej formy záväzného kurikula.

⁶ Zrejme nemusím zdôrazňovať, že nemám na mysli nástroje v zmysle rôznych softvérových pomocných aplikácií.

⁷ To však primerane odrážalo poznanie a postoje v časoch, kedy napríklad vznikala súčasnosť Štátny vzdelávací program.

⁸ A niekedy takéto moderné tendencie a postoje počúvame na vzdelávaní priamo od učiteľov.

⁹ Pozri napríklad [youtube.com/watch?v=-svFs4IIa3g&t=3s](https://www.youtube.com/watch?v=-svFs4IIa3g&t=3s) alebo [youtube.com/watch?v=VnsDqYTfi0g](https://www.youtube.com/watch?v=VnsDqYTfi0g) alebo [youtube.com/watch?v=DQSV6DVNrQ8&t=10s](https://www.youtube.com/watch?v=DQSV6DVNrQ8&t=10s) a iné, aké má určite každý z nás.

pretože práve to moderná spoločnosť potrebuje¹⁰, a (c) *čítanie s porozumením* – nech to znie akokoľvek netradične, aj túto prednosť opakovane spomínajú učiteľky a učители na vzdelávaníach. S „malou dušičkou“ sme začali na informatike používať už na 1. stupni pracovné zošity a toto rozhodnutie sa nám potvrdzuje v školách ako produktívne a správne. Pracovné zošity a práca s nimi prináša navyše aj niekoľko ďalších pozitívnych javov¹¹.

- (Aj) informatika musí byť na seba náročná a sebakritická. Neučme zo zotrvačnosti žiakov to, čo bolo samozrejmosťou pred niekoľkými rokmi alebo to, čo učili nás. Buďme dôsledne kritickí a reflektívni – pripravení rozlúčiť sa s čímkoľvek, ak sa vývoj ubera inam¹².
- Aj v školskej informatike prijímame skutočnosť, že **globálne javy sveta sa týkajú všetkých** a každá forma a zložka vzdelávania na ne musí reagovať. V tomto duchu vydalo v roku 2018 OECD stanovisko pod názvom *The Future of Education and Skills* [1]. Tu je citát z Predslovu (úmyselne som ho nepreložil do slovenčiny):

We are facing unprecedented challenges – social, economic and environmental – driven by accelerating globalisation and a faster rate of technological developments. At the same time, those forces are providing us with myriad new opportunities for human advancement. The future is uncertain and we cannot predict it; but we need to be open and ready for it. The children entering education in 2018 will be young adults in 2030. Schools can prepare them for jobs that have not yet been created, for technologies that have not yet been invented, to solve problems that have not yet been anticipated. It will be a shared responsibility to seize opportunities and find solutions.

OECD v tejto a ďalších správach poukazuje na potrebu reagovať v aktuálnych kurikulumoch – na každom stupni vzdelávania a v každej vzdelávacej oblasti – na rýchle, hlboké a zle predvídateľné zmeny. Nech to znie akokoľvek neadekvátne, týka sa to aj nás v informatike. V časti 6 sa pokúsim predostrieť jeden možný smer ďalšieho rozvoja, ktorý možno naznačuje príležitosti urobiť posun aj týmto smerom. OECD vo svojej správe uvádza tri veľké výzvy tohto meniaceho sa a ohrozeného sveta:

- **Výzva ekologická** – klimatické zmeny a decimovanie prírodných zdrojov vyžaduje urgentnú reakciu.
- **Výzva ekonomická** – bezprecedentné inovácie vo vede a technológiách otvárajú základné otázky o nás samotných. Rastúca produktivita a efektívnosť ekonomiky enormne zvyšuje nároky na vytváranie a využívanie dát a otázky bezpečnosti a vyťahuje nás rôznym neznámym rizikám.
- **Výzva spoločenská** – rastúca populácia, zanikajúce zdroje klimatické zmeny, často umocňované populistickými politickými praktikami, prinášajú masívne a nekontrolovateľné migračné vlny a ohrozujú dôveru k vládam a štandardnému spoločenskému poriadku. To otvára ďalší priestor hrozbám vojen a terorizmu.

Spomínané javy umocňujú potrebu adekvátnych znalostí, zručností a postojov. Nemôžeme sa domnievať, že tieto skutočnosti sú mimo

informatického či matematického vzdelávania. Sme povinní uvažovať o tom, ako na ne reagovať aj v našich koncepciách a obsahu školskej informatiky.

3 PROGRAMOVANIE NIE JE IBA PROGRAMOVANIE

Chcem sa pokúsim objasniť, ako vnímam programovanie. Zrejme sa zhodneme na tom, že cieľom informatiky nie je naučiť sa ten alebo onen programovací jazyk či stať sa programátormi. V časti 2 som napísal, že jeden z cieľov informatiky – a školského programovania – spočíva v objavovaní nových „nástrojov“ na riešenie problémov (problémov už známych alebo takých, o ktorých ešte ani nevieme). Staršia verzia anglického kurikula používala slová *učiť sa očakávať neočakávané*. Pre mňa je programovanie spôsob premýšľania¹³, je to jedna zo stratégií, ako môžeme pristupovať k poznávaniu, prostriedok¹⁴ na rozvoj informatického myslenia, ale aj inštrument na jeho vyjadrenie.

Programovanie vnímam skutočne široko. Som ovplyvnený postojom, ktorý Papert prezentoval v [2]: Program je objekt, ktorým premýšľame a o ktorom premýšľame. Možno práve tým sa informatika vymedzuje proti matematike: Objektom nášho záujmu je program (istým spôsobom sformulovaný plán riešenia) ako vyjadrenie myšlienky. Použijem tu dva krátke úryvky z úvodu našej Metodické príručky k vzdelávaciemu obsahu Emil 4, pozri [3], určenej pre učiteľky a učiteľov 1. stupňa:

Jazykom a kostrou školskej informatiky, ak sa nemá redukovat' na rozvoj základných počítačových zručností, je programovanie. Nie programovanie ako remeslo či zručnosti v práci s niektorým programovacím jazykom, ale programovanie ako nový nástroj na premýšľanie o problémoch a ich riešeníach, nový nástroj na poznávanie, skúmanie a naprávanie sveta. Je to užitočná príležitosť na rozvoj tvorivosti, komunikácie a spolupráce – a opäť povieme – pre každého žiaka v triede.

... programovanie je komplexná forma učenia sa a riešenia problémov, pri ktorej žiaci čítajú (aj text, aj iné reprezentácie a záznamy či plány postupu), premýšľajú, diskutujú, porovnávajú, opravujú, vyjadrujú a vysvetľujú, plánujú a vykonávajú, rozpoznávajú obmedzenia. Vytvára efektívne príležitosti, ako sa učiť vytrvať pri riešení problému, spolupracovať a komunikovať, sústrediť sa, mať radosť z objavy, mať radosť zo spoločnej práce a učenia sa.

Zoznam takýchto „Bloomových sloviac“ by mohol pokračovať ďalej: *riešiť, zahrať, vysvetliť a nechať si vysvetliť, predstaviť si, preskúmať (analyzovať), zmeniť alebo zmeniť reprezentáciu, opraviť, doplniť, zopakovať...* Čo všetko žiaci *analyzujú* pri takto vnímaných programovacích aktivitách? Bývajú to rôzne reprezentované dáta, obrázky, aké majú vzniknúť, obrázky, aké vznikli, rôzne typy situácií, rôzne obmedzené prostriedky na vyriešenie takýchto situácií alebo pravidiel správania (riadenia) a pod.

Ak chápeme a implementujeme školské programovanie takto široko, vytvárame pre žiakov (všetkých vzdelávacích stupňov) vhodné okolnosti na systematické a presné uvažovanie, rozvoj tvoriv-

¹⁰ Individuálnu prácu žiakov na informatike pomocou pripravených pracovných listov metódou *každý postupuje vlastným tempom* považujem za zmeškanie najfantastickejšieho vlaku či lietadla alebo rakety, v akej sa mohli žiaci odvieť.

¹¹ Pripravujeme v našom tíme o tom inú publikáciu.

¹² Tým nemám na mysli prispôbovať sa každej technologickej a možno dočasnej odchýlke. Mám na mysli to, že o každom jednom jave, postupe a téme sa musíme opakovane a kriticky pýtať sami seba, či si aj naďalej zaslúži v škole priestor.

¹³ A to aj o ostatných tematických okruhoch.

¹⁴ Môj dedko by povedal *šikovný vercajch*. Pred sto rokmi mal 31.

vosti, abstraktného myslenia, dekompozície a syntézy, algoritmického myslenia atď. Chcem tu však zdôrazniť: Takýmto sa programovanie stáva iba prostredníctvom správnych didaktických zásad a postupov¹⁵, aké by sme mali používať pri práci so vzdelávacím obsahom.

Tá bohatá oblasť možností, ktoré pre jednoduchosť označujeme ako programovanie, sa s postupujúcimi ročníkmi, teda až na strednú školu nie zužuje. Presne naopak – významne rozširuje. Ako sa pokúsím sformulovať v časti 6, aj na strednej škole by mala žiakov pozývať na atraktívne mosty vedúce do ďalších oblastí vzdelávania, vedy a výskumu, včítane spoločenských.

Ak by som zotrval pri poetike typu *informatická zvedavosť* a nechal sa inšpirovať jazzom, poviem, že *programovanie je (pre mňa) životným pocitom a postojom*¹⁶.

3.1 Výkonový štandard

Kvôli cieľu, ku ktorému sa chcem dostať ďalej, urobím teraz malú odbočku do kurikula: Sumarizujem tu (nie doslovne a len v bodoch), čo stanovuje inovovaný ŠVP ako výkonový štandard¹⁷ pre každého žiaka na záver 8. ročníka ZŠ v predmete informatika, a to v tematickom okruhu *Algoritmické riešenie problémov* – prípadne v inom okruhu, ak takýto výkon považujem za jasne prepojený na algoritimizáciu. Dočítame sa, že žiak na konci 8. ročníka základnej školy dokáže/vie:

- *vyhľadávať a ziskávať informácie, rozpoznať a popísať vzťahy, zakódovať, dekodovať, vyhľadať, použiť, uvažovať o vlastnostiach informácií a o ich reprezentáciách,*
- *pracovať so štruktúrou (tabuľky, grafy, postupnosti obrázkov, čísel, ...) – vytvárať, manipulovať, používať, interpretovať,*
- *uvažovať o vzťahoch a pracovať so vzťahmi medzi dátami v štruktúre,*
- *riešiť problém, naprogramovať... pomocou postupnosti, opakovania a rozhodovania,*
- *rozumieť opakovaniu a uvažovať o počte opakovaní, opakujúce sa vzory v dátach, v programe,*
- *rozumieť jednoduchým podmienkam a rozhodovaniu, zostaviť, zapísať, využiť pri riešení, interpretovať,*
- *pracovať s premennou, rozpoznať, čo je premenná, ako si uchovať hodnoty (vstup), interpretovať, upravovať, pracovať s výrazom,*
- *uvažovať o jazyku, aplikovať pravidlá a obmedzenia a ďalšie konštrukcie jazyka (vetvenie a podmienky),*
- *uvažovať o riešení a programe, o rôznych riešeniach, vlastnostiach programu,*
- *uvažovať o vzťahu vstup – algoritmus – výsledok.*

Takto formulovaný ambiciózny výkonový štandard povinnej informatiky na záver 2. stupňa ZŠ nadobudol účinnosť 1. septembra 2015, a to postupne od 1. a 5. ročníka. Uvedomujem si, že to nie je dlho – každý žiak, ktorý ukončil ôsmu triedu v júni 2019 alebo neskôr (teda vlastne „teraz“), už má za sebou štyri roky takejto novej informatiky. Pozrime sa teda hlbšie v čase, pohľad do staršieho ŠVP¹⁸ sumarizuje Tabuľka 1. Bez toho, aby som analyzoval tieto

vzdelávacie programy – alebo ich hodnotil – chcem zdôrazniť niečo iné a skutočne závažné, a to **priepravný rozdiel medzi zamýšľaným (intended), realizovaným (implemented) a dosiahnutým (attained) kurikulumom v informatike**¹⁹. Presne to je hlavnou témou môjho príspevku.

Tabuľka 1: Vzdelávacie požiadavky v informatike v ŠVP vo verzii z roku 2008 v okruhu *Postupy, riešenie problémov, algoritmické myslenie*

Pojmy	Vlastnosti a vzťahy, postupy a metódy
<ul style="list-style-type: none"> • <i>postup riešenia, etapy riešenia problémov,</i> • <i>programovací jazyk, elementárny príkaz, postupnosť, procedúra, cyklus,</i> • <i>zložitost riešenia problému</i> 	<ul style="list-style-type: none"> • <i>v detskom programovacom prostredí riešenie úloh s opakovaním nejakých činností, zoskupovanie častí riešenia do procedúr,</i> • <i>porovnanie času trvania rôznych riešení problému</i>
Obsahový štandard	Výkonový štandard
<p><i>Postup riešenia, formálny zápis riešenia, etapy riešenia problémov.</i></p> <p><i>Programovací jazyk, elementárny príkaz, postupnosť, cyklus, procedúra, parametre, premenná, hodnota, priradenie.</i></p> <p><i>Zložitost riešenia problému.</i></p>	<p><i>Žiak:</i></p> <ul style="list-style-type: none"> • <i>dokáže zapisovať a interpretovať postupy do formálneho zápisu (napríklad zápis matematických algoritmov, algoritmus na jednoduché zašifrovanie textu),</i> • <i>demonštruje v detskom programovacom prostredí riešenie úloh s opakovaním nejakých činností, zapamätávanie výpočtov do premenných, zoskupovanie častí riešenia do procedúr,</i> • <i>dokáže porovnať čas trvania rôznych riešení problému.</i>

4 ZAČIATKY INFORMATIKY A JEJ SÚČASNÝ STAV

Pripomínam pre úplnosť aj ostatné roky: povinná školská informatika sa na strednej škole na Slovensku zaviedla v roku 1985²⁰, povinná informatika na 2. stupni v roku 2005 a na 1. stupni v roku 2008. To znamená, že každý žiak, ktorý skončil 1. stupeň ZŠ približne v roku 2010 alebo neskôr už absolvoval dva alebo tri roky informatiky a každý žiak, ktorý skončil 8. ročník ZŠ v roku 2014 a neskôr aspoň šesť rokov. Znamená to tiež, že prakticky všetci dnešní bakalári už majú po svojej ZŠ a SŠ (ak bola všeobecnevzdelávacia) za sebou informatiky deväť rokov.

¹⁵ To, samozrejme, zahŕňa rôzne aspekty vyučovania, napr. formy interakcie medzi učiteľom a žiakmi a žiakmi navzájom, ale aj výskumné metódy používané pri vývoji a overovaní obsahu, a následne pri jeho skúmaní.

¹⁶ Takéto niečo však zvyčajne počujeme iba od jazzmana, ktorý má (medzi iným) aj úžasnú techniku a hlboké porozumenie hudby a rytmu.

¹⁷ Teda záväzný a minimálny súbor obsahu a výkonov **všetkých žiakov**.

¹⁸ Je zrejmy výrazný a moderný posun od verzie z roku 2008 k verzii súčasnej. Ale už aj požiadavky z roku 2008 sú vlastne ambiciózne a očividne kontrastujú s citátmi, ktoré uvádzam v časti 4.

¹⁹ To ale nie je špecifický problém 2. stupňa základnej školy.

²⁰ S malými úpravami v rozsahu za celých týchto tridsaťšesť rokov.

To, na čo poukazujem v mojom príspevku, je obraz celkovej (a nedostatočnej) podpory učiteľov za celé tie dlhé roky²¹ – na všetkých možných úrovniach. V žiadnom prípade ale necítim dôvod ukazovať prstom na učiteľov z praxe. A rovnako na nikoho. Jediné, čo môže pomôcť, je premýšľať a diskutovať o tom, ako nájsť v školskej informatike únik zo začarovaného kruhového objazdu, zo stratenej zóny neefektívnosti a večných začiatkov, v ktorej sme uviazli.

S kolegami, s ktorými realizujeme na fakulte výučbu niektorých predmetov v magisterskom programe aplikovanej informatiky v bloku Tvorba softvéru pre vzdelávanie, sme v zimnom semestri 2020/21 aj pre potreby tohto príspevku robili anketu o tom, ako si spomínajú²² na svoje informatické vzdelávanie na základnej a strednej škole. Celkovo sme zatiaľ získali 27 odpovedí, z ktorých vyberám 14 citátov (každý od iného študenta, tých zvyšných v našej vzorke som neofiltroval preto, že by obsahovali opäť postoj. Citát, ktorý som použil v abstrakte, tu už neopakujem):

- *ide o mrhanie časom, roky sme strávili wordom a excelom, a stále ho ani moc dobre neovládame,*
- *informatika na školách je veľmi slabá, a je potrebné, aby sa zaviedol systém a neučilo sa stále odznova alebo neučilo sa vlastne nič,*
- *na strednej škole sme sa na informatike učili robiť vo Worde. Bol to naozaj stratený čas,*
- *informatika na školách je veľmi slabá. Ja som síce mala dobrú informatiku aj na základnej aj strednej, ale čo som vždy počúvala kamarátov či decká z iných škôl, tak to bolo veľmi zlé,*
- *aj z vlastnej skúsenosti viem, že informatika na školách je veľmi slabá, bolo by vhodné aby sa zaviedol systém a informatika by sa vyučovala na potrebnej úrovni,*
- *informatika je často zanedbávaná na školách ..., patrí medzi prvé hodiny, ktoré sa rušia, ak je potreba robiť v škole niečo iné,*
- *bola som na gymnáziu a informatika tam nebola takmer žiadna. Vlastne v poslednom roku nejaký nepodarený pokus ale ako žiadna sláva, čiže ani s wordom a excelom to nebolo niečo extra. Ak by som nebola rok na strednej škole v zahraničí, tak informatiku na strednej škole nemám žiadnu,*
- *moja stredoškolská informatika bola na nadpriemernej úrovni, ale aj tak mám pocit, že to mohlo byť oveľa lepšie. Z nejakého dôvodu sa stále na informatiku pozerá ako na zábavu pri počítači,*
- *čo sa týka informatiky na základnej škole, tak som ju prakticky nezažil. Miesto učenia sa sme sa len hrali nejaké počítačové hry. Na strednej škole to bolo lepšie...,*
- *učili sme sa stále dokola to isté a nikdy nič poriadne,*
- *v rámci našej školy bola informatika veľmi slabá, až na posledný ročník na strednej škole. Nepamätám si, že by sme mali informatiku na základnej škole...,*
- *až keď som nastúpila na výšku, som si uvedomila, že základy programovania nie sú len o písaní kódu, ale hlavne o spôsobe myslenia – logického myslenia,*
- *absenciou výučby programovania (na ZŠ, vložil I.K.) trpí celé vysokoškolské programovanie. Nakol'ko ani stredné školy nie*

²¹ To sa prostredníctvom rôznych projektov začína meniť. Aj keď pri niektorých mimo univerzitných iniciatívach vnímam možno dobrý zámer, ale didaktickú nekompetenciu. To je však téma samo o sebe.

všetky poskytujú kvalitný základ programovania, tak sa prvé semestre, resp. aj viacero semestrov učia základy, ktoré ak by študenti mali zmáknuté už počas ZŠ či SŠ, tak by sa vysokoškolské štúdium mohlo zameriavať na komplexnejšie veci,

- *... milujem umenie, rada chodím do divadla, do múzeí a galérií, rada šijem, vyšívam, vyrábam rôzne veci, a vlastne kvôli tomu ma baví aj programovanie, lebo vďaka nemu niečo vytváram.*

Domnievam sa, že k týmto citátom netreba viac slov a je zrejme, prečo pokračujem pokusom o identifikovanie príčin tohto javu a signalizovaných problémov. A iniciovaním diskusie o možných zmenách.

4.1 V čom cítim problémy súčasnej školskej informatiky

Opäť priznávam: nie je to pokus o dáky systematický a starostlivo zoradený zoznam bodov, je veľmi ťažké a nejednoznačné správne oddeliť javy, príčiny, dôsledky, prejavy a námety na opatrenia... Vo vystúpení sa zameriam na týchto šesť:

- Reálna úroveň dosahovaných výsledkov v školskej informatike (toto nie je validný výsledok výskumu, ale „iba“ neprehliadnuteľné hlasy učiteľov 2. stupňa – o 1. stupni, učiteľov SŠ o 2. stupni, nás učiteľov na fakultách o SŠ, a koniec-koncov aj študentov informatiky, ktorí spomínajú na svoje školské roky). Sumarizujem slovami, ktoré som už použil aj skôr: **prívelký rozdiel medzi zamýšľaným kurikulumom, realizovaným kurikulumom a tým dosiahnutým.**
- Obraz, ako vnímajú školskú informatiku niektorí učelia (aj informatiky, aj iných predmetov), vedenie škôl, žiaci, rodičia, štát, verejnosť, novinári či politici. Príčasť ako zábavu a „voľné“ hodiny za počítačom. Príčasť ako prácu s textom, prezentáciami a tabuľkami. Ale prečo sa potom treba aj k základom týchto oblastí znova a znova vracieť, niekedy až po úroveň univerzity?
- Snaha o prenášanie skúseností z „krúžkovej informatiky“ do informatiky pre každého žiaka, niekedy aj zamieňanie si týchto dvoch prostredí²³. Aj keď oboch veľmi dôležitých, ale iba jeden z nich predstavuje všeobecné vzdelanie pre všetkých, má štátne kurikulum a má sa gradovať po ročníkoch či stupňoch cez celé vzdelávanie.
- Niekedy didakticky rozpačitá alebo až nekompetentná „pomoc“ s vyučovaním informatiky a vzdelávaním učiteľov (a to aj učiteľov na 1. stupni) zo strany rôznych iniciatív, niekedy aj IT spoločnosti. Ale pracovať v IT a kvalifikovane robiť didaktiku informatiky nie je to isté.
- Neexistencia „strednej vrstvy“ didaktických dokumentov. Myslí tým súbor materiálov orientovaných na učiteľa, školu, metódičov a autorov vzdelávacieho obsahu, ktoré by detailnejšie interpretovali štátne kurikulum, teda by zapadali medzi kurikulum a reálne vyučovaný obsah informatiky na škole. Takéto dokumenty by mali o.i. obsahovať aj gradovaný plán vzdelávacích cieľov školskej informatiky (napr. podľa takých cyklov, aké používa iŠVP v súčasnosti, teda pre 3. a 4. ročník, 5. a 6. ročník, potom 7. a 8., a napokon 10. až 12. Vzhľadom na situáciu v škol-

²² Tieto výpovede, samozrejme, musíme interpretovať opatrne. Ale vzhľadom na otázky orientované na spomienky pred niekoľkými rokmi azda nie sú veľmi poznačené pandemickou náladou.

²³ Zvyčajne však v zmysle vnímania školskej informatiky ako niečoho, čo sa „dá robiť podobne“ ako krúžok či klub.

skej informatike, napr. aj s ohľadom na percento v súčasnosti ne-kvalifikovaných učiteľov učiacich informatiku na školách, cítim však potrebu napísať – po ročníkoch).

- Dostávam sa k tomu najhorúcejšiemu: problém večných začiatčovníkov v informatike. Obávam sa, že sme na mnohých (primnohých) školách uviazli v akejsi slučke a nevieme sa z nej vy-motať. Na vyššom stupni sa vraciame do skoro toho istého štartu, ale – odhadujem – **so stále menším a menším tímom súťažia-cich**²⁴. Vraciame nie v zmysle *prípomeňme si, zopakujme si*, ale v zmysle *musíme začať od nuly*.

4.2 Čo sú naproti tomu úspechy v našej ko-munite

Nebolo by konštruktívne, ak by som nezdôraznil to pozitívne, čo sa už v správnom smere zlepšenia tejto situácie v našej komunite deje.

Väčšina z našich pracovísk je viac alebo menej zapojená do rôz-ných projektov s cieľom podporovať učiteľov informatiky z praxe. V ostatných rokoch vznikajú súbory metodických materiálov pre učiteľov informatiky²⁵. Prebiehajú série webinárov s vedením škôl, učiteľmi a začínajúcimi koordinátormi digitálnej transformácie. Navrhujeme, aby sme v práci s nimi prezentovali moderný a konzis-tenčný, ale náročný pohľad na školskú informatiku. Takú, ktorá podporuje a pomáha rozvoju digitálnej gramotnosti žiakov (a kole-gov učiteľov), ale zdravo dôrazne si chráni svoj priestor primárne pre vlastnú informatiku. A v nej pre programovanie, ktoré vo veľ-kej gradácii sleduje jednu líniu a podporuje premyslený poznávací proces v informatickom vzdelávaní na ZŠ a SŠ.

Ďalšia veľká – a čoraz silnejšia devíza – je rozrastajúca sa skupina mladých doktorandov a výskumníkov na našich katedrách. Bez najmenších pochybností hovorím, že aj kvalita ich výstupov sa z roka na rok zvyšuje. My starší, napr. ich školitelia a kolegovia z akademického prostredia, im musíme byť nápomocní a nároční na ich prácu zároveň. Musíme sa snažiť ďalej aktívne rozširovať túto skupinu a pomôcť jej rásť. Bol by som rád, ak by sme v blízkej budúcnosti dali pri voľbe tém dôraz na školskú informatiku pre všetkých. Aj podpora talentovaných žiakov a krúžkovej činnosti je významná a má na Slovensku bohatú tradíciu. Teraz je ale v ďalšom rozvoji školskej informatiky a jej výskume²⁶ (napr. edukačnom výskume vývojom [4]) „každá ruka dobrá“. V časti 7 sa ešte vrátim k úvahám o možnom smerovaní našej komunity, ktoré by snáď pomohlo ďalšiemu rozvoju školskej informatiky.

5 KDE ZAČÍNAME A...

Celý informatický oblúk sa vo formálnom vzdelávaní začína na 1. stupni a končí 3. ročníkom na SŠ (v prípade všeobecno-vzdeláva-cej školy). Deti však žijú vo viac alebo menej digitalizovanom pro-stredí už od malička. Domnievam sa preto, že je dôležité a zaujímavé spolupracovať už s materskou školou a skúmať, aké formy používania digitálnych technológií sú pre túto vekovú skupinu bez-pečné, vývinovo primerané a prospešné. V rámci týchto aktivít (s dôrazom na „predškolský“ ročník detí) si kladieme tri ciele:

(a) metódami pedagogického výskumu vyvíjať a overovať také ak-tivity, ktoré vhodne prospejú rozvoju detí v tomto veku a ich prí-prave na základnú školu, ďalej (b) poskytnúť pedagógom, deťom a ich rodičom príklady *dobrého používania DT*, a tiež (c) vytvoriť pre deti príležitosť na získavanie prvých *predinformatických* skú-seností. Medzi naše didaktické a vývojové princípy v práci s touto vekovou skupinou patrí zásada, že deti nikdy nepracujú s tabletom, robotickou hračkou alebo iným zariadením tohto typu individuálne. V každej našej aktivite pracuje malá skupina detí²⁷ aktívne a spolu, napr. sediac alebo stojac (teda skôr poskakujúc) pred interaktívnou tabuľou. Skupiny sa zvyčajne striedajú po 10 alebo 12 minútach. Takýmto aktivitami chceme prispievať k rozvoju ich rôznych kompetencií a získaniu silných prospešných zážitkov²⁸.

Záväzný začiatok informatického vzdelávania sa však odohráva na 1. stupni ZŠ. Vo vývojovej práci nášho tímu sme prijali stratégiu, že tomuto prvému formálnemu kroku treba venovať veľkú pozor-nosť – a koniec-koncov to vyžaduje aj štátny vzdelávací program. Domnievam sa, že ak sa tu informatika nerozbehne obsahovo a vý-vinovo primerane postavená a realizovaná, ohrozíme jej vnímanie aj na 2. stupni. A že by sme tak prispievali k tomu spomínanému *začarovanému kruhovému objazdu*. Uvedomujeme si, že na 1. stupni musí informatika prispievať aj k rozvoju základov digitálnej gramotnosti žiakov. S tým ale súvisí aj ďalší bod našej stratégie: Vzdelávací obsah pre 1. stupeň – včítane prvých rokov programo-vania – tvoríme tak, aby ho chcel a mohol a trúfol si realizovať pe-dagóg primárneho vzdelávania vo svojej vlastnej triede²⁹. To nepo-važujeme za obmedzenie, ale jedinečnú príležitosť budovať infor-matické základy moderným spôsobom, aj s celkom prirodzeným prepojením na rozvoj matematického myslenia a ďalších predme-tov či oblastí vzdelávania. A navyše, domnievame sa, že učiteľka alebo učiteľ, ktorého si tento prístup získa, bude náchylnejší **pri každej opodstatnenej príležitosti** použiť DT aj na svojich iných predmetoch. Našu dôveru v takýto prístup budovania *informatiky pre každého*³⁰ posmeľujú nielen reakcie učiteľov, ale napr. aj Papert vo svojom predhovore v [7] (uvádzam tu len tri z jeho dvanástich charakteristík logovskej kultúry):

- *Vychádzame tiež z predpokladu, že deti dokážu programovať už v pomerne ranom veku.*
- *Predpoklad, že deti dokážu programovať, vyjadruje veľmi veľa: v tejto kultúre veríme (opravujem sa: vieme), že deti pochádzajúce z akéhokoľvek sociálneho prostredia³¹ a v každom veku do-kážu omnoho viac, než sa o nich predpokladá (zvýraznil I.K.). Stačí, ak im poskytneme vhodné nástroje a príležitosti.*
- *Rovnakú dôveru, že deti dokážu viac, než sa očakáva, máme aj voči učiteľom a učiteľkám.*

Prikladám tu ešte krátky citát z našej Metodickéj príručky pre pe-dagógov [3]:

... začať s programovaním ... už v 3. ročníku a systematicky po-kračovať v ďalších ročníkoch má okrem kognitívneho opodstat-nenia aj zdôvodnenie afektívne či emočné. Ak v nižšom veku žiaci

²⁴ Teda žiakov, ktorí to ešte „nevzdali“.

²⁵ Zrejme najpočetnejšie teraz v projekte IT Akadémie, ale určite aj v ďalších menších projektoch, napr. projektoch KEGA alebo VEGA.

²⁶ Napr. edukačnom výskume vývojom [4].

²⁷ Zvyčajne tretina alebo polovica skupiny.

²⁸ Zdôrazňujem, že nám ide aj o to, aby sme rodičom poslali signál, aké užitočné môžu byť DT. Príklady sme už videli aj na konfe-rencii Didinfo, napr. [5] alebo [6] a iné.

²⁹ Teda v triede, kde učí skoro všetky predmety. A teda učiteľ alebo učiteľka, ktorá nie je špecializovaná na informatiku. Ak sa nám ju nepodarí získať, je to pravdepodobne naša chyba. Skúsajme lepšie.

³⁰ Už na 1. stupni a s významnou prítomnosťou takého programo-vania, o akom som písal v časti 3.

³¹ Potvrďujú tu napr. aj naše vlastné zážitky z práce s rómskymi žiakmi, pozri www.youtube.com/watch?v=-svF5411a3g

riešia primerane náročnú gradáciu programovacích úloh a darí sa im, získavajú pozitívny citový vzťah k informatike, programovaniu, riešeniu problémov a logickému mysleniu.

Vzdelávací obsah a metóda, ktorú sme vyvinuli pre 1. stupeň, je určená žiakom 3. ročníka (resp. 3. a 4. ročníka) a 4. ročníka (resp. 4. a 5. ročníka). Touto určitou voľnosťou chceme prispieť k prepájaniu školskej informatiky na 1. a 2. stupni. V súčasnosti zahajujeme vývoj vzdelávacieho obsahu, ktorý bude pokračovaním dvoch predchádzajúcich „dielov“, teraz však už s jazykom Scratch.

6 ... KAM CHCEME PRÍSŤ

Na základe našich terajších skúseností z 1. stupňa, pozri napr. [8], a predchádzajúcich skúseností z ročníkov 5 a 6 s programovaním pre každého v projekte ScratchMaths, pozri napr. [9] alebo [10], považujeme za realistické očakávať, že po absolvovaní 8. ročníka bude každý žiak v triede³² vedieť:

- používať základné príkazy, vytvárať, opravovať a upravovať príkazy vlastné (zložený príkaz v Emilovi 4, môj blok v jazyku Scratch a pod.), a to aj so vstupom, cielene používať a vytvárať jednoduché funkcie v Pythone buď bez alebo aj so vstupom),
- zostaviť, vykonať, upraviť či opraviť postupnosť príkazov alebo opakovanie skupiny príkazov (pevné, nekonečné a s podmienkou),
- rozpoznať, zostaviť a vyhodnotiť podmienku a využiť ju na podmienené vykonanie skupiny príkazov,
- vytvoriť, upraviť a vyhodnotiť premennú, výraz s premennou,
- získať a použiť vstupy a zadať výstupy rôznych typov (bloky otázka, odpoveď, bublina a myšlienka v Scratchi, a to napr. aj v kontexte riadenia micro:bitu v prostredí Scratch alebo inom, získať a používať vstupné dáta v Pythone a pod.),
- vytvárať a využívať základné udalosti, udalosťami riadený program,
- plánovať, vytvárať a vykonávať paralelné scenáre pre jednu postavu,
- plánovať správanie násobných postáv v Scratchi, plánovať a využívať komunikáciu a spoluprácu medzi násobnými postavami,
- zostavovať a využívať zoznamy jednoduchých hodnôt (napr. v jazykoch Scratch a Python).

Uvedomujem si, že porozumenie a prijatie určitého javu alebo postupu (napr. abstraktného myslenia) je pre rôzne vekové úrovne žiakov na rôznej úrovni, že sa ku všetkým týmto výkonom musíme opakovane vracáť, opakovane ich utužovať a využívať v rôznych situáciách a prostredia, s ktorými pracujeme, a neustále o nich viesť podnetné diskusie medzi žiakmi. Ale verím, že nebudeme musieť s nimi na strednej škole (opäť) začínať prakticky od nuly – za predpokladu, že sa nám podarí tento postoj rozšíriť a realizovať na jednotlivých stupňoch vzdelávania tak, ako sa to viac-menej úspešne darí v matematike a ako to vyžaduje štátny vzdelávací program v informatike. Nemali by sme ale práve pre takýto (nový, ale zároveň už dlhšie vyžadovaný) postoj urobiť všetko, čo je v našich silách? Aby sme na strednej škole nemuseli opäť začínať od začiatku (potichu zmierení s tým, že už si aj tak nemalú časť triedy pre takýto obsah nikdy nezískame)?

Je teda celkom prirodzené, že sa zamýšľam aj nad tým, kam vlastne má prísť informatika na druhom „konci“ spomínaného informatického vzdelávacieho oblúka. Uvedomujem si tiež, že je to náročný

proces a že určite budeme potrebovať mať „vo vrecku“ aj dočasné a náhradné a skrátene „obchádzky“. Ale prihovám sa za to, aby sme sa o tento správny krok pokúšali ako komunita.

6.1 Stredná škola ako (zrejme posledná) príležitosť

Na strednú školu všeobecno-vzdelávacieho typu sa pozerám aj ako na „poslednú príležitosť“ rozvíjať si poznanie do šírky, získavať kvalitný pohľad na rôzne prírodovedné a spoločenskovedné oblasti – skôr ako sa na „cestu“ do zamestnania alebo na vysokú školu začneme užšie špecializovať. Taktiež si myslím, že je to v istom zmysle jedinečná príležitosť získať pohľad na informatiku a jej programovanie ako na silný a užitočný nástroj na skúmanie prekvapujúco rôznorodých problémov, od využitia v matematike, cez prácu z dátami, genetiku, výskum vesmíru, analýzu a syntézu reči, textu, hudby... Kde sa končí tento zoznam? Nebola by škoda nevyužiť takúto šancu?

Áno, uvedomujem si, že som tu na tenkom lade, v neprebádanom a nedostatočne premyslenom teréne. To ma však neodradí, pretože takúto príležitosť považujem za potenciálne unikátny a presvedčivý „dôkaz“ pre žiakov SŠ, že informatika je silná, užitočná a má zmysel. Zdá sa mi, že to môže byť cesta, ako rozvíjať produktívnu spoluprácu medzi predmetmi, medzi učiteľmi a medzi žiakmi. Uvedomujem si tiež, že sa tu vynára veľa nových otázok, o ktorých treba diskutovať, hľadať a overovať inovátné riešenia. Uvedomujem si tiež, že ak sa pokúsime ísť touto cestou, narazíme na veľa nečakaných prekážok. Aj tak si myslím, že je to výzva a azda správne smerovanie a že stoj za námahu.

Ďalej už ostanem len pri niekoľkých odrážkach námetov a predstáv, z akých vývojových princípov by sa tu dalo vychádzať. Ide zatiaľ len o náznaky a nedomyslené, neprediskutované predstavy, sny a východiská:

- snažme sa robiť stredoškolskú informatiku ako príležitosť pre moderné inovátné vzdelávanie, ktoré otvára mosty do rôznych oblastí a poskytuje (veľmi dobre naplánované, premyslené a gradované) príležitosti a úžasné príklady úžasného učenia (sa),
- skúsme zažiť so žiakmi to, že programovanie je silné a má zmysel – nie samo pre seba, ale pre bádanie a poznávanie, tvorbu, komunikáciu, sebavyjadrenie. Vychádzam pritom z (dnes odvážneho) predpokladu, že každá žiačka a každý žiak si už na základnej škole vytvoril veľmi dobré základy,
- stredná škola všeobecno-vzdelávacieho typu v svojej nematuritnej etape nemá zabíhať do vysokoškolského štúdia, ale vynikajúco, široko a otvorene rozvíjať mladých ľudí (pokiaľ nejde o špecializované štúdium, rôzne zamerania a pod.),
- mali by sme mať odvahu prijať princíp, že žiaci sa veľmi veľa dokážu naučiť jeden od druhého, v tímoch, pri zaujímavých tvorivých činnostiach – ak majú možnosť mať voľbu oblastí, postupov a cesty, ktorou sa chcú v takýchto bádaniach pustiť. To neznamená, že takýto proces nemá byť starostlivo naplánovaný a rôznymi formami synchronizovaný tak, aby mali z práce každého tímu osoh všetci v triede,
- uvažujem teda o systéme či schéme projektovo a tímovo realizovaných „blokov“, možno polročných alebo trojmesačných. Aj keď by mali ponúkať bohatú rozmanitosť, mali by byť akoby „zavesené“ na dobre premyslenú gradáciu, ktorá definuje určité

³² Samozrejme, v podobnom zmysle, ako vie podľa iŠVP každý žiak na konci 9. ročníka napr. ... *vyriešiť jednoduchú lineárnu rovnicu s jedným výskytom neznámej.*

- požiadavky na (a) záväzný (nie veľký, a teda nie prevládajúci) príspevok k informatickému obsahu samotnému, (b) záväzný (opäť nie veľký, teda nie prevládajúci) príspevok k programovacím javom, štruktúram, dátam, poznaniu...
- mohli by sme poskytnúť žiakom počas stredoškolskej informatiky napr. v štyroch alebo piatich etapách vždy dve alebo tri alternatívne témy pre tímovú prácu, s výborným pomocným a inšpirujúcim materiálom pre každú tému (a metodický materiál pre každú z tém pre učiteľa),
 - tieto zadania tém by mali určité etapy a niektoré predpísané ciele a prvky, poskytovali by ale veľkú voľnosť v prístupe, delbe práce, metódach a pod.,
 - úloha učiteľa by bola pomáhať, konzultovať a radiť pri vyhľadávaní zdrojov, synchronizovať proces, podporovať príležitosti, **aby sa každý veľa naučil od všetkých ďalších v triede**. K tomu by mali prispieť vzájomné prezentácie nastudovaných tém, použitých postupov a získaných objavov a riešení.

Na záver tejto časti chcem ešte priložiť neúplný, nezoradený a opäť zatiaľ absolútne neprediskutovaný zoznam niekoľkých možných námetov na takéto témy či moduly. Určite je ich už teraz priveľa, mnohé ďalšie tu nie sú, iné nie sú vhodné alebo vyžadujú neprimeraný odborný aparát... Je to zatiaľ iba náznak. Jeden možný príklad potom v časti 6.1.1 rozvediem detailnejšie.

- Informatika a hudba (analýza, komponovanie...)
- Informatika a text (analýza textu, generovanie náhodných textov, básničiek, príbehov...)
- Informatika a výtvarné umenie
- Informatika a počítačová grafika
- Informatika pomáha matematike
- Umelá inteligencia, robotika, VR
- Programovacie jazyky
- Logické obvody, počítačové systémy
- Internet vecí. Programovanie pre web
- Komunikácia, siete, prenosy, spolupráca systémov
- Modelovanie, simulácie
- Informatika a medicína, genetika, bioinformatika
- Práca s dátami, big data, vizualizácia dát
- Programovanie v aplikáciách typu Office (makrá, customisation, Visual Basic v aplikáciách)
- Etický Hacking. Informatika a bezpečnosť

6.1.1 Výpočtová zvedavosť a abstraktné výtvarné umenie 20. storočia

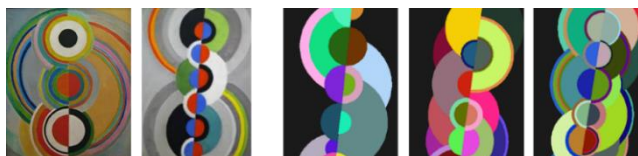
Ako ilustráciu toho, o čom premýšľam v súvislosti s možným smerovaním programovania na SŠ, chcem použiť predmet, ktorý už štyri roky realizujeme s kolegyňou Bezákovou na našej fakulte v magisterskom programe Aplikovaná informatika. V súčasnosti teda ide iba o nepriamy pokus a neadekvátne overovanie. Predsa to však považujem za dôležitý a užitočný krok a ilustračný príklad v mojom príspevku. Tu je niekoľko faktov, východísk a cieľov:

- Pôvodná myšlienka a inšpirácia pochádza od J. Claysona [11]. V prvej verzii svojej knihy používal Terrapin MacLogo, neskôr obsah transformoval do Comenius Loga³³ a v rokoch 2013-15 ho

s podporou kolegov Blaha a Tomcsányiho transformoval do jazyka Python a overoval na Deep Springs College v USA [12].

- Podstatou tohto vzdelávacieho obsahu (a teraz aj nášho predmetu na fakulte) je zapojiť žiakov (v našom prípade zatiaľ študentov) do skúmania niektorých smerov abstraktného výtvarného umenia prostredníctvom programovania. Naším cieľom je sprístupniť im nový zážitok³⁴, iný (a pre študentov vždy prekvapujúci) pohľad na programovanie ako na nástroj na skúmanie niečoho neznámeho a z pohľadu bežnej praxe nezvyčajného.
- Prostredníctvom programovania skúmame a modelujeme kľúčových autorov týchto výtvarných prúdov, akými sú napr. Mondrian, Malevich, Delaunay, Rothko, Klee, Calder a pod. Kladieme si otázky typu *Čo robí Soniu Delaunay Soniou Delaunay?* Keď ich diela skúmame a modelujeme prostredníctvom programovania, prijímame a programujeme rôzne teórie, ktoré postupne rozvíjame, vizuálne vyhodnocujeme a vylepšujeme. Takto orientujeme aj párové projekty našich študentov, ktoré počas semestra robia a ktoré sú pre väčšinu z nich a v každom behu predmetu opäť aj prekvapením, aj výzvou, a radi sa jej chopia.
- Matematika, ktorú v predmete používame, je v plnom rozsahu stredoškolská (niekedy dokonca až základnoškolská).
- Z pohľadu vlastného programovania ide zväčša o bežnú znalosť Pythonu, aká zodpovedá požiadavkám strednej školy³⁵. Používame iba základné konštrukcie, číselné hodnoty, funkcie, žiadne dátové štruktúry a žiadne pokročilé programovacie postupy.
- Používame štandardný modul Turtle, využívame násobné korytnačky a vymýšľame a vylepšujeme niekoľko (z pohľadu computer science nie náročných ale zaujímavých) algoritmov.

Predmet naši študenti prijímajú s prekvapením, ale skoro všetci (zatiaľ v každom zo štyroch opakovaní) pozitívne. Napriek tomu, že sa z pohľadu programovania nenaučia v tomto stave poznania už nič nové, vo svojich záverečných reakciách uvádzajú iné pozitíva.



Obrázok 1: Štúdie diela S. Delaunay v projekte jednej z dvojice študentov

Obrázok 1 ilustruje najprv dve zo známych diel výtvarníčky³⁶, potom vpravo sériu vlastných programovaných štúdií (teda iterovaných teórií a pokusov) jednej z dvojice študentov o tom, čo sú v týchto dielach určujúce prvky, ich umiestnenie a vzťahy medzi nimi, rovnováhy, farebné palety a pod. Takéto vizuálne pokusy si nekladú za cieľ napodobniť zvolené výtvarné dielo, ale premýšľať o nich a skúmať ich, teda učiť sa novému pohľadu na výtvarné dielo, osvojiť si nové bádateľské postupy – prostredníctvom programovania – a mať príležitosť nahliadnuť do oblastí umenia, ktorá je nanešťastie informatikom často pomerne vzdialená.

Nie je takýto vzdelávacie obsah aspoň čiastočne ilustráciou toho, čo by mohli žiaci zažiť na modernej všeobecno-vzdelávacej strednej škole? Aj keď nás len čaká transformácia a overenie takto oriento-

³³ Vtedy sme sa ako tím autorov Comenius Logo s profesorom J. Claysonom z American University of Paris aj zoznámili.

³⁴ Na aký už zrejme mnohí z nich nikdy nebudú mať „čas a príležitosť“.

³⁵ Avšak nie celkom na začiatku strednej školy.

³⁶ Pozri Musée National d'Art Moderne, Paris, www.mam.paris.fr

vanej intervencie pre skupiny stredoškôľakov, vnímam to ako príležitosť, ktorá môže:

- ďalej rozvíjať programovacie zručnosti a porozumenie (ak by bola napr. cieľom práca s násobnými perami či korytnačkami, paralelnými procesmi alebo kódovanie farieb a ich transformácie medzi rôzne vhodnými reprezentáciami),
- pre niektoré skupiny žiakov atraktívne ilustrovať silu programovania ako užitočného nástroja na skúmanie a modelovanie,
- a tiež spoznať niektoré výtvarné smery a kultúrno-spoločenské pozadie doby, v ktorej vznikali.

7 AKO ĎALEJ

V záverečnej časti sa chcem zamerať na úvahy, ako prispieť k zmene, o ktorej hovorím. Myslím si, že sa dá nájsť cesta von zo spomínaného metaforického cyklu (a svojou prácou sa o to pokúšam). Niektoré možnosti som už naznačoval, niektoré ešte nie:

- Prvým vhodným krokom by bola inicializácia diskusie v rámci odbornej komunity. Priznal som v texte niekoľkokrát, že ide iba o prvé myšlienky. V častiach, kedy som uvažoval o 1. a 2. stupni, sa síce opieram o veľmi povzbudzujúce a rozsiahle skúsenosti ostatných šiestich či siedmich rokov³⁷, z projektov ScratchMaths a Informatika s Emilom, aj keď sme v oboch narazili a ešte narazíme na rôzne problémy a prekážky. V úvahách o pokračovaní na strednej škole momentálne zatiaľ prevažne iba „extrapolujem“, čo treba brať s rezervou. Bez diskusie sa ďalej nikam neposunieme. Budem veľmi rád, ak sa k podobným témam vrátíme (najneskôr) aj na nasledujúcich konferenciách Didinfo.
- To, o čo sa môžeme pri prichádzajúcom akreditačnom procese pokúsiť na našich katedrách, sú nové alebo obnovované študijné programy pre budúcich učiteľov informatiky na 2. a 3. stupeň. Teraz máme naliehavú príležitosť premýšľať, či by sme nemali urobiť určité úpravy – možno len obsahové, možno strategické – v našich študijných programoch. Celkom určite by som napr. navrhoval byť náročnejší vo vyžadovanej úrovni bežnej digitálnej gramotnosti a očakávaných zručnostiach pri práci s textom, tabuľkami a pod. Ale primárne mám na mysli náš prístup najmä k didaktike informatiky v príprave budúcich učiteľov a v ich didaktike programovania.
- Veľký priestor na lepšie porozumenie, inovácie a zmeny poskytujú našim pracoviskám rôzne projekty a výskumné aktivity – naše a našich doktorandov. Sám za seba si hovorím, že moje ďalšie aktivity tohto druhu sa už budú zameriavať iba na *školskú informatiku pre všetkých* – lebo práve v nej ide o každého žiaka. Možno by sme ako katedry mohli uvažovať aj o spoločnom projekte, zameranom na spomínané problémy.
- Niežby som teraz objavoval niečo nové, čo v našej komunite nerobíme: chcem spomenúť vývoj metodických a podporných materiálov pre učiteľov z praxe a ich priebežné vzdelávanie. Samozrejme, že materiály budú rôzne, pretože máme rôzne autor-ské prístupy a rôzne koncepcie. Ale to, čo by ich malo pomerne záväzne prepájať, je platné kurikulum. A možno – na základe ďalšej diskusie – aj niektoré pohľady a postoje, ktorým som sa venoval v predchádzajúcich častiach tohto príspevku. A ešte jedna teraz mimoriadne aktuálna téma: Domnievam sa, že všetky

formy doplnujúcich, rozširujúcich a atestačných štúdií v informatike by mali patriť výlučne do kompetencie akreditovaných fakúlt, ktoré musia pravidelne a opakovane dokladovať v tejto oblasti svoje profesionálne kompetencie.

- A napokon: Aby som nenavrhol unáhlenú revoltu, ale prospešnú – aj keď dôraznú – zmenu pre žiakov a ich učiteľov, potrebujeme zrejme rôznorodé štandardy a požiadavky, ale zároveň aj rôzne „skratky“ a „dobiehačky“ vo forme podporných materiálov a doplnujúcich aktivít (napr. *mikrokurzov*).

Prihovám sa za náročný a rýchly posun, ktorý sa ale žiakom oplatí a prinesie im radosť z učenia sa a úspechu.

8 POĎAKOVANIE

Ďakujem agentúre VEGA, ktorá prostredníctvom nášho projektu 1/0602/20 prispela k tomu, že tento príspevok môžem publikovať na konferencii Didinfo 2021.

BIBLIOGRAFICKÉ ODKAZY

- [1] OECD The Future of Education and Skills. Education 2030. OECD 2018, 21 p., [oecd.org/education/2030-project/contact/E2030_Position_Paper_\(05.04.2018\).pdf](https://oecd.org/education/2030-project/contact/E2030_Position_Paper_(05.04.2018).pdf)
- [2] PAPERT, S. *Mindstorms. Children, Computers, and Powerful Ideas*. Basic Books, 1980, New York, 230 p.
- [3] KALAŠ, I., BLAHO, A., MORAVČÍK, M. (2021) *Informatika s Emilom 4. Programovanie pre 4. a 5. ročník základnej školy. Metodická príručka*. Indícia, s.r.o. (v tlači), 77 str.
- [4] PLOMP, T., NIEVEEN, N. (Eds.) *Educational Design Research: An Introduction*. SLO Enschede 2013, 203 p.
- [5] BEZÁKOVÁ, D. Informatika v cirkuse. Zborník konferencie *Didinfo 2019*, Banská Bystrica, pp. 22-25.
- [6] BEZÁKOVÁ, D. Cestičky – programovanie pre predškôľakov. Príspevok prijatý na konferenciu *Didinfo 2021*, Banská Bystrica.
- [7] PAPERT, S. *What is Logo? And Who Needs It?* Preface in: *Logo Philosophy and Implementation*. LCSII, 1999, 164 p.
- [8] BLAHO, A., KALAŠ, I., MORAVČÍK, M. Programovací jazyk v prostredí Emil pre 4. ročník. Príspevok prijatý na konferenciu *Didinfo 2021*, Banská Bystrica.
- [9] BENTON, L. KALAS, I., HOYLES, C., NOSS, R. Designing for learning mathematics through programming: A case study of pupils engaging with place value. *Int J of Child-Computer Interaction*, vol. 16, 2018, pp. 68-76.
- [10] KALAS, I., BENTON, L. Defining Procedures in Early Computing Education. In: Tatnall A., Webb M. (eds) *Tomorrow's Learning: Involving Everyone. Learning with and about Technologies and Computing*. WCCE 2017. IFIP Advances in Information and Communication Technology, vol 515. 2017, Springer, pp. 567-578.
- [11] CLAYSON, J. *Visual Modelling with Logo*. A structural Approach to Seeing. The MIT Press, Cambridge, MA, 1988. 392 p.
- [12] CLAYSON, J. *A Computational Eye*. unpublished manuscript, Deep Springs College 2015. 379 p.

³⁷ Nič to nedokazuje, ale pre určitú predstavu chcem odhadnúť, že za obdobie spomínaných rokov a projektov sme intenzívne s kolegami zo štyroch rôznych tímov v rôznych krajinách pravidelne pracovali s veľkým počtom žiakov v našich ôsmich partnerských

vývojových školách (tzv. *design schools*) a vzdelávali minimálne 1500 učiteľov v rôznych krajinách. Všetky tieto príležitosti považujem za jedinečný a nenahraditeľný zdroj skúseností a skutočné východisko, z ktorého sa dá čerpať.

Príspevky

Papers

ICDL/ECDL vo vzdelávaní žiakov stredných škôl

ICDL/ECDL in the education of secondary school students

Gabriela Andrejková
Ústav informatiky
Prírodovedecká fakulta UPJŠ
Jesenná 5
040 01 Košice
Slovensko
gabriela.andrejkova@upjs.sk

Mária Baxová
Slovenská informatická spoločnosť
FMFI UK, Mlynská dolina 5692
842 48 Bratislava
Slovensko
baxova@ecdsl.sk

Slavka Blichová
Univerzita Pavla Jozefa Šafárika
v Košiciach
Centrum celoživotného vzdelávania
a podpory projektov
Šrobárova 2, 041 80 Košice
Slovensko
slavka.blichova@upjs.sk

ABSTRACT

One of the possibilities to verify systematic knowledge and skills in the field of computer technology is provided by the international system ICDL/ECDL. In some thematic areas, it allows the verification of knowledge and skills at two levels of difficulty, namely the standard, which is needed to work in normal positions, and the advanced, which is important to support more complex and demanding professional activities. As part of the national project IT Academy – education for 21st century, 6 765 secondary school pupils from 104 schools were involved in the activities to obtain the ICDL/ECDL certificate at the standard level, 2 663 of them have so far participated in the verification in 4 key modules, which provides us with a sufficient number for statistical analysis of areas in which students make the most mistakes. In the article, we present an analysis of these errors and make recommendations for consultants who prepare students for testing in order to reduce these errors.

Keywords

ECDL. ICDL. Word processing. Cyber security. IT Security. Computer essentials.

ABSTRAKT

Jednu z možností na overenie systematických znalostí a zručností v oblasti práce s výpočtovou technikou poskytuje medzinárodný systém ICDL/ECDL. V niektorých tematických oblastiach umožňuje overenie znalostí a zručností na dvoch úrovniach náročnosti, a sice na štandardnej, ktorá je potrebná pri práci na bežných pracovných pozíciách, a na pokročilej, ktorá je dôležitá na podporu odborných komplexnejších a náročnejších činností. V rámci národného projektu IT Akadémia – vzdelávanie pre 21. storočie bolo do aktivít na získanie certifikátu ICDL/ECDL na štandardnej úrovni zapojených (pripravovaných) 6 765 žiakov zo 104 stredných škôl, 2 663 z nich sa doposiaľ zúčastnilo overovania v 4 kľúčových moduloch, čo nám poskytuje dostatočný počet na štatistickú analýzu oblastí, v ktorých sa žiaci dopúšťajú najviac chýb. V článku uvidíme analýzu týchto chýb a uvidíme odporúčania pre konzultantov, ktorí pripravujú žiakov na testovanie, s cieľom tieto chyby redukovat.

Kľúčové slová

ECDL. ICDL. Spracovanie textu. Kybernetická bezpečnosť. Bezpečnosť pri práci s IKT. Základy práce s počítačom.

1 ÚVOD

ICDL Foundation (do mája 2019 ECDL Foundation) je medzinárodná organizácia zriadená CEPISom (rada európskych profesionálnych informatických spoločností), ktorej cieľom je

vytvárať štandardy digitálnych kompetencií potrebných pre bežných používateľov počítača na pracovnom trhu a aj pre aktívny život, poskytovať systém ich overovania a podporu pri vzdelávaní. Týmto chce docieľiť posilňovanie jednotlivcov, organizácii a celej spoločnosti v efektívnej výkonnosti a konkurencie schopnosti. V spolupráci s národnými garantmi systému ICDL/ECDL zo zapojených štátov a ich akreditovanými testovacími centrami je tvorcom a globálnym poskytovateľom certifikačných programov. Programy umožňujú nadobudnúť a overiť prenositeľné štandardné aj pokročilé digitálne kompetencie, ktoré sú nevyhnutné na efektívne využívanie osobných počítačov a ďalších digitálnych nástrojov na bežných i expertných pracovných pozíciách, prípadne doplniť si niektoré špecifické základné digitálne a odborné znalosti [2]. Keďže sa jedná o medzinárodné štandardy (v mnohých štátoch zaradené aj do národných kvalifikačných rámcov), ktoré prispievajú k medzinárodne porovnateľnej digitálnej gramotnosti žiakov a študentov, vzdelávanie a overovanie digitálnych kompetencií podľa systému ECDL bolo zaradené do projektu IT Akadémia.

Výskum digitálnej gramotnosti na Slovensku poukazuje na to, že sice počet digitálne gramotných obyvateľov rastie, ale stále sú rezervy v jednotlivých oblastiach [1]. Napríklad, podľa tabuľky G15 v [1] až 27 % respondentov nemá skúsenosti s textovým editorom, čo je základná digitálna gramotnosť, a až 44 % nemá skúsenosti s vyhľadávaním informácií v počítačových sieťach. U žiakov stredných škôl (SŠ) a študentov vysokých škôl (VŠ) je pravdepodobne táto situácia podstatne lepšia, avšak doterajšie skúsenosti s testovaním podľa ECDL ukazujú, že je potrebné venovať viac pozornosti niektorým témam s cieľom upevniť terminológiu a zefektívniť prácu pri využívaní počítačov.

V našom príspevku veľmi stručne pripomenieme informácie o systéme ICDL/ECDL a urobíme analýzu doterajších výsledkov testovania v troch moduloch s odporúčeniami pre ich zlepšenie.

2 SYSTÉM ICDL/ECDL V PROJEKTE IT AKADÉMIA

Systém ICDL/ECDL je modulárny. Ucelená tematická oblasť IT, v ktorej je možné vzdelávať sa a vykonať skúšku, sa v systéme ECDL nazýva **modul**. Pre každý modul je pripravený detailný štruktúrovaný sylabus, ktorý je jednotný pre celý svet a je podkladom pre medzinárodné overovacie (certifikačné) skúšky.

Sylaby sú na Slovensku dostupné v anglickom a slovenskom jazyku, skúšky majú jednotnú štruktúru, pozostávajú z praktických úloh v zvolenom konkrétnom softvérovom prostredí, sú v slovenskom jazyku, pričom kľúčové moduly M2, M3, M7 majú aj verziu s maďarským podtextom. V prípade zdravotných

znevýhodnení, je možné štandardnú skúšku absolvovať v tzv. individuálnom režime. Skrátene ukážky testov sú verejnosti k dispozícii [2].

Okrem toho systém ICDL/ECDL poskytuje zaregistrovaným záujemcom o skúšky referenčné príručky, z ktorých niektoré sú dostupné aj v slovenskom jazyku [2], s ukázkami realizácie úkonov v konkrétnych softvérových aplikáciách, ktoré sú zoradené podľa bodov daného sylabu.

V rámci projektu IT Akadémia [3, 4] sme sa rozhodli ponúknuť žiakom SŠ a študentom VŠ všetky dostupné moduly zo skupiny modulov Base (základné, rozumej kľúčové moduly so štandardnou náročnosťou) a skupiny modulov Standard (rozšírenie základu, rozumej rozšírenie kľúčových zručností o zručnosti z ďalších potrebných oblastí IT), ktoré boli pri príprave projektu na Slovensku k dispozícii:

- M2 Základy práce s počítačom / Computer Essentials,
- M3 Spracovanie textu / Word Processing,
- M4 Tabuľkový kalkulátor / Spreadsheets,
- M5 Používanie databáz / Using Databases,
- M6 Prezentácia / Presentation,
- M7 Základy práce online / Online Essentials,
- M9 Práca s obrázkami a grafikou / Image Editing,
- M12 Bezpečnosť pri využívaní IKT / IT Security.

Žiaci a študenti si z ponuky 8 modulov vyberajú 4 moduly podľa svojho uváženia a zamerania, v ktorých absolvujú dištančnú prípravu formou e-learningu pomocou vytvorených študijných materiálov (dostupných na stránke vzdelávacieho portálu [4]) a samotné testovanie v Akreditovaných testovacích centrách ECDL (akredituje ich národný garant ICDL/ECDL, Slovenská informatická spoločnosť), ktorými sú Školské výpočtové strediská CVTI SR (pre žiakov SŠ) a partnerské univerzity UPJŠ a ŽU (pre študentov VŠ). V rámci prípravy žiaci SŠ využívajú konzultácie k vybraným témam (konzultantmi sú učitelia SŠ). Úspešní absolventi získavajú Certifikát ECDL Profile s medzinárodnou platnosťou.

Počty žiakov SŠ, ktorí sa doposiaľ zúčastnili testovania sú nasledujúce: M2 – 2 354 žiakov, M3 – 2 663, M4 – 981, M5 – 65, M6 – 2 250, M7 – 1 871, M9 – 100, M12 – 241. Väčšina žiakov si vybrala moduly M2, M3, M6 a M7.

V nasledujúcom poukážeme na chyby, ktorých sa žiaci dopúšťajú v moduloch M2, M3 a M12. Chybosť testovania v ďalších piatich moduloch uvádzame v prílohe 1.

3 ANALÝZA CHÝB V JEDNOTLIVÝCH MODULOH

Analýza odpovedí na otázky v testoch jednotlivých modulov nám umožňuje upozorniť konzultantov, ktorým témam je potrebné venovať dôslednejšiu prípravu. Ukazuje sa, že teoretické otázky (a terminológia) sú pre testovaných ťažšie, preto je potrebné nájsť metódy na upevnenie vedomostí. Všetky otázky si vyžadujú dôsledné čítanie s porozumením a presné vykonávanie úloh podľa zadania. Zároveň je dôležité, aby boli žiaci pripravení aj na časový rámec, nakoľko na vykonanie jedného testu je stanovených 45 minút.

Pri analýze výsledkov testovania uvedených v Tabuľkách 1-3 v texte a 5-8 v prílohe vychádzame z očíslovaných položiek sylabov. Prvé číslo v položke sylabu definuje modul (2 – M2, 3 – M3, 12 – M12), druhé číslo definuje kategóriu v danom module

a posledné číslo súbor zručností. Údaje v tabuľkách sú usporiadané podľa percentuálnej chyby v klesajúcom poradí.

3.1 Modul M2 – Základy práce s počítačom

Modul M2 je zameraný na znalosti a prácu s operačným systémom, správou súborov, prácu s počítačovou sieťou a externými zariadeniami. Je to čiastočne praktický a čiastočne teoretický modul. Je to jeden zo vstupných modulov s náročnosťou primeranou pre svet práce, ktorý je určený pre širokú verejnosť [2].

Úspešný absolvent skúšky by mal mať znalosti podľa nasledujúcich kategórií uvedených v sylabe [2]:

1. **Počítače a zariadenia** – základné pojmy, ktoré sa týkajú IKT, počítačov, príslušných zariadení a softvéru, spustenie a ukončenie práce s počítačom
2. **Pracovná plocha, ikony, nastavenia** – efektívna práca s pracovnou plochou počítača, využívanie ikon a práca s oknami, základné nastavenia operačného systému, funkcie pomocníka
3. **Výstupy** – práca s jednoduchým textom a tlač
4. **Správa súborov** – základy práce so súbormi a priečkami, ich usporiadanie, uchovávanie a kompresia údajov
5. **Siete** – základné pojmy počítačových sietí, prístupy do siete
6. **Bezpečnosť a ochrana zdravia** – ochrana údajov a zariadení, škodlivý softvér, význam zálohovania, ochrana zdravia a životného prostredia pri IT

Tabuľka 1: Analýza chybovosti podľa kategórií v module M2

Položka sylabu	Oblasť znalostí a zručností	Počet osôb	Počet úloh	Chyba [%]
2.1.3	Softvér a licencie (Počítače a zariadenia)	2354	4723	26.8
2.4.3	Uchovávanie a kompresia údajov (Správa súborov)	2354	7806	19.0
2.4.1	Základy práce so súbormi a priečkami (Správa súborov)	2354	14545	18.4
2.2.3	Nástroje a nastavenia (Pracovná plocha, ikony, nastavenia)	2354	11375	17.9
2.1.2	Hardvér (Počítače a zariadenia)	2354	4708	16.9
2.5.1	Základné pojmy počítačových sietí (Siete)	2354	5114	14.9
2.6.3	Ochrana zdravia a životného prostredia pri IT (Bezpečnosť a ochrana zdravia)	1970	2765	14.8
2.2.1	Pracovná plocha a ikony (Prac. plocha, ikony, nastavenia)	1626	3633	14.4
2.1.4	Zapnutie / spustenie počítača, vypnutie (Počítače a zariadenia)	404	404	14.4
2.3.2	Tlač (Výstupy)	2354	3098	14.1
2.1.1	IKT (Počítače a zariadenia)	1208	1208	13.8
2.5.2	Prístup do siete (Siete)	2354	3525	13.3
2.6.1	Ochrana údajov a zariadení (Bezpečnosť a ochrana zdravia)	2354	3141	13.1

2.4.2	Usporiadanie súborov a priečinkov (Správa súborov)	2354	7839	12.2
2.2.2	Práca s oknami (Pracovná plocha, ikony, nastavenia)	2354	2735	8.0
2.3.1	Práca s textom (Výstupy)	2354	7061	7.4
2.6.2	Škodlivý softvér (malvér) (Bezpečnosť a ochrana zdravia)	2354	3535	6.7

Analýza výsledkov testovania v module M2 Základy práce s počítačom (Tabuľka 1) ukázala, že žiaci tento modul zvládajú veľmi dobre. Celková chybovosť je iba 14.5 %, resp. celková úspešnosť je 85.5 %, a to aj napriek tomu, že tento modul obsahuje skoro 53 % teoretických otázok. V rámci teoretických otázok sa striedajú formulácie typu „urči pravdivé tvrdenie“, „urči nepravdivé tvrdenie“, „čo nemôže nastať“, a pod., takže je veľmi dôležité čítať otázku a možné odpovede s porozumením a dostatočne rýchlo vybrať správnu odpoveď.

V rámci prvej kategórie **Počítače a zariadenia** žiaci zvládajú tri súbory zručností na celkom dobrej úrovni – *IKT* na 86.2 %, *Hardvér* na 83.1 % a *Zapnutie / spustenie počítača, vypnutie* na 85.6 %, napriek tomu, že sú to skoro všetko teoretické otázky. S uvedenými pojmi sa stretávajú priebežne na všetkých stupňoch škôl, nakoľko vzdelávací štandard predmetu informatika obsahuje časť Softvér a hardvér. Celkovo najväčšia chybovosť je práve v súbore zručností *Softvér a licencie*, až 26.8 %, čiže viac ako jedna štvrtina úloh bola riešená nesprávne. Asi najväčším problémom sú práve rôzne typy softvérových licencií aj napriek tomu, že vzdelávací štandard predmetu informatika obsahuje časť *Informačná spoločnosť – legálnosť používania*. Tejto oblasti by bolo potrebné venovať dostatočnú pozornosť využívajúc napr. riadený rozhovor o softvéroch, ktoré majú žiaci nainštalované na svojich domácich počítačoch.

V druhej kategórii **Pracovná plocha, ikony, nastavenia** bola najväčšia chybovosť v súbore zručností *Nástroje a nastavenia* – 17.9 %. Táto chybovosť môže byť spôsobená aj tým, že žiaci veľmi často sami nenastavujú napr. rozlíšenie, dátum a čas, nemenia predvolený jazyk, resp. neprikladajú jazykové nastavenia klávesnice. Uvedené zručnosti teda nemajú zautomatizované, pretože ich nevykonávajú opakovane a pravidelne. Ak to porovnáme napr. so súborom zručností *Práca s oknami*, kde je chybovosť iba 8.0 %, je evidentné, že s oknami pracujú pravidelne, a preto riešenie úloh z tejto oblasti je pre nich oveľa ľahšie.

Tretia kategória **Výstupy** je najúspešnejšia, nakoľko celková chybovosť je 9.45 %, t. j. z celkového počtu 10 159 úloh žiaci nesprávne vyriešili iba 960 úloh.

Vo štvrtej kategórii **Správa súborov** sa ukázalo, že žiaci nemajú väčšie problémy s úlohami zo súboru zručností *Usporiadanie súborov a priečinkov*, kde je chybovosť 12.2 %, ale riešenie úloh z ďalších dvoch súborov zručností už nedopadlo až tak dobre. Súbor zručností *Základy práce so súborami a priečinkami* má chybovosť 18.4 % a *Uchovávanie a kompresia údajov* 19.0 %. Najčastejšie chyby sa opäť týkajú väčšinou zručností, s ktorými sa nestretávajú pravidelne – meniť spôsob zobrazenia súborov a priečinkov, vyhľadávanie súborov na základe vlastností (názov, formát, obsah, dátum poslednej zmeny,...), rýchly prístup.

Piata kategória **Siete** obsahuje dva súbory zručností *Základné pojmy počítačových sietí* a *Prístup do siete*. Celková chybovosť je 14.2 %. Táto kategória obsahuje teoretické úlohy, ktoré kladú veľké nároky na čítanie s porozumením, ale je evidentné, že väčšinu pojmov z tejto oblasti majú žiaci zvládnuté.

Šiesta kategória **Bezpečnosť a ochrana zdravia** je druhá najúspešnejšia, nakoľko celková chybovosť je 11.2 %, t. j. z celkového počtu 9 441 úloh žiaci nesprávne vyriešili 1 057 úloh. Z troch súborov zručností najhoršie dopadol súbor zručností *Ochrana zdravia a životného prostredia pri IT*, kde chybovosť je 14.8 %. Napriek tomu, že táto kategória obsahuje teoretické úlohy, je evidentné, že žiaci majú vcelku uspokojivé znalosti z oblasti bezpečnosti a ochrany zdravia.

3.2 Modul M3 – Spracovanie textu

Spracovanie textu sa stalo každodennou súčasťou ľudí nielen v pracovnom procese, ale aj v osobných aktivitách. Spracúvame oficiálne listiny pre obchodný styk, resp. pre styk s úradmi, ale aj osobné dokumenty napr. na prezentovanie našich zámerov, resp. na uchovanie našich myšlienok.

Aplikácie na spracovanie textu ponúkajú množstvo nástrojov na zefektívnenie práce, ktoré neboli k dispozícii v ére písacích strojov. Je dôležité, aby sme o týchto nástrojoch vedeli, ak chceme využívať kapacitu ponúkaných možností pri vytváraní a spracovávaní textových dokumentov, a tým dosahovať nielen vyššiu vlastnú pracovnú efektívnosť, ale dať možnosť jednoduchej úpravy svojich dokumentov aj iným, ak sa dokument stáva napr. súčasťou väčšieho celku alebo súboru.

Sylabus modulu M3 definuje základné pojmy a zručnosti, ktoré by mal bežný človek ovládať, aby mohol aplikáciu na spracovanie textu využívať efektívne pre každodennú potrebu.

V rámci vzdelávacieho štandardu predmetu informatika je práca s textom súčasťou oblasti *Reprezentácie a nástroje* už na 1. stupni základnej školy, pokračuje na 2. stupni základnej školy a tiež na strednej škole. Na každom stupni je výkonový a obsahový štandard na vyššej úrovni tak, aby sa žiaci naučili efektívnym a tvorivým spôsobom využívať informačno-komunikačné technológie, informačné zdroje a možnosti aplikačného programového vybavenia.

Úspešný absolvent skúšky by mal mať znalosti podľa nasledujúcich kategórií uvedených v sylabe [2]:

1. **Používanie aplikácie** – základná práca s dokumentmi – vytváranie nového dokumentu, vytváranie dokumentu zo šablóny, ukladanie na konkrétne miesto, ukladanie v rôznych formátoch, zvyšovanie produktivity vhodným nastavením aplikácie
2. **Tvorba dokumentu** – vytvorenie a úprava textových dokumentov, ich zdieľanie a distribuovanie
3. **Formátovanie** – spôsoby formátovania textu, odseku, používanie štýlov
4. **Objekty** - vkladanie tabuliek, obrázkov a grafických objektov do dokumentov
5. **Hromadná korešpondencia** – príprava dokumentov pre hromadnú korešpondenciu a výstupy hromadnej korešpondencie
6. **Príprava výstupov** – nastavenia dokumentu pred tlačou (stránkovanie, okraje, práca s hlavičkou a pätičkou), kontrolovanie a oprava pravopisu pred finálnou tlačou dokumentov

Výsledky testovania sú uvedené v Tabuľke 2. V rámci prvej kategórie **Používanie aplikácie** základnú *Prácu s dokumentmi* zvládajú žiaci na celkom dobrej úrovni, nakoľko chybovosť je iba

12.5 %, t. j. zo 4 623 úloh bolo nesprávne vyriešených približne 578 úloh. Najčastejšie chyby v tomto súbore zručností často vychádzajú z problémov žiakov čítať zadania textov s porozumením a vykonať zadanie presne podľa pokynov – uloženie na určené miesto na disku, uloženie pod iným menom, v inom formáte. Tieto zručnosti sa často považujú za samozrejmosť a je možné, že sa im nevenuje dostatočná pozornosť v zmysle presného vykonania úlohy.

Tabuľka 2: Analýza chybovosti podľa kategórií v module M3

Položka sylabu	Oblasť znalostí a zručností	Počet osôb	Počet úloh	Chyba [%]
3.5.2	Výstupy (Hromadná korešpondencia)	2633	2633	46.3
3.5.1	Príprava (Hromadná korešpondencia)	2633	5266	39.5
3.1.2	Zvyšovanie produktivity (Používanie aplikácie)	1994	1994	37.3
3.3.3	Štýly (Formátovanie)	2633	3302	30.8
3.3.2	Odseky (Formátovanie)	2633	16133	23.9
3.6.1	Nastavenia (Príprava výstupov)	2633	8911	20.2
3.6.2	Kontrola a tlač (Príprava výstupov)	2633	3275	17.5
3.4.2	Formátovanie tabuľky (Objekty)	2633	4904	16.5
3.1.1	Práca s dokumentmi (Používanie aplikácie)	2633	4623	12.5
3.2.1	Vkladanie textu (Tvorba dokumentu)	2633	3657	10.9
3.2.2	Označovanie, upravovanie (Tvorba dokumentu)	2633	8839	10.5
3.4.3	Grafické objekty (Objekty)	2353	4050	7.3
3.3.1	Text (Formátovanie)	2633	11513	6.9
3.4.1	Vytváranie tabuľky (Objekty)	2633	5266	5.0

V rámci súboru zručností *Zvyšovanie produktivity* je chybovosť podstatne vyššia – 37.3 %, t. j. z 1 994 úloh bolo nesprávne vyriešených približne 744 úloh. Najčastejšie chyby v rámci týchto úloh sú pri používaní pomocníka a pri navigácii v dokumente. Vo všeobecnosti pri práci s ktoroukoľvek aplikáciou by žiaci mali byť vedení k tomu, aby si vedeli pri problémoch pomôcť vyhľadáním informácií cez pomocníka alebo na odborných diskusných fórach. Takáto flexibilita pri práci s akoukoľvek aplikáciou bude pre nich veľkou devízou neskôr v pracovnom procese. Techniky navigácie v dokumente nepatria do obsahového štandardu na strednej škole. Na 2. stupni základnej školy majú vo výkonovom štandarde uvedené „používať nástroje na vyhľadanie a nahradenie textu“, ale nie navigácia v dokumente. Nakoľko väčšinou žiaci aj na strednej škole pracujú s krátkymi, resp. kratšími dokumentami, je možné, že sa priamo s navigáciou v dokumente, ktorá sa používa hlavne pri dlhých dokumentoch, ani nestretli, a preto tieto schopnosti nemajú utvrdené a zautomatizované. Ak sa pozrieme na prvú kategóriu celkovo, je možné vyhodnotiť, že chybovosť žiakov je 20.0 %, t. j. z celkového počtu 6 617 úloh v tejto kategórii žiaci nesprávne vyriešili 1 322 úloh.

Druhá kategória **Tvorba dokumentu** dopadla veľmi dobre, nakoľko súbor zručností *Vkladanie textu* aj súbor zručností *Označovanie, upravovanie* zvládajú žiaci na približne 90.0 %. Chybovosť v oboch častiach je 10.9 % a 10.5 %, t. j. z celkového počtu 12 496 úloh v tejto kategórii žiaci nesprávne vyriešili 1 326 úloh. Z uvedeného vyplýva, že týmto zručnostiam a schopnostiam sa v obsahu školskej informatiky venuje dostatočná pozornosť a drobné nepresnosti, resp. nesprávne riešenia často vyplývajú z už spomenutých problémov žiakov čítať zadania textov s porozumením a vykonať zadanie presne podľa pokynov (napr. vloženie textu na správne miesto v dokumente, vymazanie konkrétneho textu, presun textu na presne určené miesto v dokumente,...).

Tretiu kategóriu **Formátovanie** možno z pohľadu úspešnosti rozdeliť na dve časti. Súbor zručností *Text* zvládajú žiaci na približne 93.0 %, pretože chybovosť je iba 6.9 %, t. j. z celkového počtu 11 513 úloh nesprávne riešili iba 794 úloh. Znamená to, že s formátovaním samotného textu nemajú väčšie problémy. Domnievame sa, že je to spôsobené aj tým, že už na 1. stupni základnej školy výkonový štandard v predmete informatika obsahuje „použiť konkrétne nástroje editora na tvorbu a úpravu textu“. Na vyšších stupňoch škôl sa tieto zručnosti len prehľbujú a utvrdzujú a žiaci si ich zautomatizovali.

Ďalšie dva súbory zručností *Odsek* a *Štýly* už nedopadli tak dobre. Chybovosť pri *Odseku* je 23.9 %, t. j. z celkového počtu 16 133 úloh žiaci nesprávne riešili 3 856 úloh a pri *Štýloch* je chybovosť až 30.8 %, t. j. z celkového počtu 3 302 úloh žiaci nesprávne riešili až 1 017 úloh. Ukázalo sa, že žiaci majú veľké medzery v práci s odsekom – nepoznajú mäkké zalomenie riadku, medzery medzi odsekmi, odsadzovanie odseku, nastavovanie zarážok tabulátorov. Naopak väčšinou bez problémov vedia nastaviť riadkovanie, zarovnávať text, pridávať a odstraňovať odrážky a číslovanie v jednoduchých zoznamoch, nastavovať orámovanie odseku. Pri zručnostiach s využívaním štýlov väčšinou vedia používať nástroj na kopírovanie formátu, ale často nevedia aplikovať existujúci štýl znakov na vybraný text. A to aj napriek tomu, že obsahový štandard pre prácu s textom na strednej škole (konkrétne na gymnáziu) obsahuje „štýl ako nástroj na formátovanie textu“ a dokonca aj „úpravu štýlov“, čo nie je zručnosť, ktorú vyžaduje sylabus M3.

Ak sa teda pozrieme na tretiu kategóriu celkovo, je možné vyhodnotiť, že chybovosť žiakov je 18.3 %, t. j. z celkového počtu 30 948 úloh v tejto kategórii žiaci nesprávne vyriešili 5 667 úloh.

Štvrtá kategória **Objekty** je najúspešnejšia, nakoľko celková chybovosť je 9.6 %, t. j. z celkového počtu 14 220 úloh žiaci nesprávne vyriešili iba 1 368 úloh. Žiaci bez problémov zvládli úlohy na *Vytvorenie tabuľky* – 95.0 % úloh bolo vyriešených úspešne. Pri *Formátovaní tabuľky* už bola chybovosť podstatne vyššia – 16.5 %, t. j. zo 4 904 úloh bolo nesprávne vyriešených približne 809 úloh. Boli to hlavne úlohy na úpravu šírky stĺpca alebo výšky riadku, kde neboli dodržané presne uvedené hodnoty zo zadania, resp. opäť žiaci nečítali zadania pozorne a nespĺnili presné pokyny, čím strácali body za jednotlivé úlohy. Úlohy zo súboru zručností *Grafické objekty* vyšli opäť s nízkou chybovosťou len 7.3 %, t. j. približne 92.7 % úloh žiaci zvládli úspešne. Medzi úlohy, ktoré boli problematické, patrili úlohy na zmenu veľkosti objektu so zachovaním alebo bez zachovania pomeru, ale väčšinou to opäť vyplývalo nie z neznalosti danej zručnosti, ale nepozorného čítania zadania úlohy.

Vysoká úspešnosť tejto kategórie je opäť naviazaná na to, že prvýkrát sa s vkladáním objektov, obrázkov stretávajú už žiaci na 1. stupni základnej školy, a keďže sa tieto zručnosti pravidelne

systematicky opakujú, žiaci s úspešným vykonaním jednotlivých úloh nemajú väčšie problémy.

Piata kategória **Hromadná korešpondencia** dopadla najhoršie, nakoľko celková chybovosť je 41.8 %, t. j. z celkového počtu 7 899 úloh žiaci nesprávne vyriešili 3 299 úloh. Úlohy na zistenie schopnosti zvládnuť hromadnú korešpondenciu boli na seba nadväzujúce, čiže kto nedokázal riešiť úlohy zo súboru zručností *Príprava*, nemal šancu pokračovať na riešení úloh zo súboru zručností *Výstupy*. Časť žiakov zvládla úlohy na prepojenie hlavného dokumentu a distribučného zoznamu, ale už nezvládla úlohy na vloženie dátových polí do hlavného dokumentu, a tým pádom ani nemohli dokončiť akýkoľvek výstup. Mnohí žiaci ale tieto úlohy vôbec neriešili a úplne ich preskakovali. Sú dve príčiny, prečo sa to dialo. Prvou príčinou je, že sa žiaci vôbec nestretli s hromadnou korešpondenciou, a tak tieto úlohy úplne vynechali. Dôvodom môže byť aj to, že hromadná korešpondencia sa nenachádza v obsahovom štandarde v predmete informatika ani na strednej škole. Druhou príčinou môže byť časový stres pri vypracovávaní testu, nakoľko otázky z kategórie hromadnej korešpondencie sú skoro na konci testu. Žiaci si častokrát neuvedomujú ako rýchlo uteká čas a zdržia sa pri niektorých úlohách na začiatku testu a k posledným piatim – šiestim úlohám sa vôbec nedostanú. Takže pri príprave na testovanie je okrem samotných zručností a schopností pracovať s aplikáciou veľmi dôležité pracovať efektívne a pripravovať sa aj na časové zvládnutie jednotlivých úloh a celého testu.

V rámci šiestej kategórie **Príprava výstupov** je v oboch súboroch zručností približne rovnaká chybovosť: súbor zručností *Nastavenia* má chybovosť 20.2 %, t. j. z 8 911 úloh bolo nesprávne vyriešených približne 1 800 úloh a súbor zručností *Kontrola a tlač* má chybovosť 17.5 %, t. j. z 3 275 úloh bolo nesprávne vyriešených približne 573 úloh. Najčastejšie chyby v rámci súboru zručností *Nastavenia* boli pri práci s vkladaním, resp. vymazávaním zlomu strany, resp. používaním viacnásobného klávesu Enter a s pridávaním polí v hlavičkách alebo päťach dokumentu. Pri *Kontrola a tlač* žiaci väčšinou bez problémov zvládali kontrolu pravopisu, ale pri tlači a nastavovaní parametrov tlače (napr. len stranu 2) opäť zaúradovala nesústreďenosť alebo stres a v úlohách prehľadali toto zadanie. Uvedené nám ukazuje, že tlačiť dokument žiaci zvládajú, len sa musia naučiť riešiť úlohu presne podľa zadania – nastaviť správne parametre.

3.3 Modul M12 – Bezpečnosť pri využívaní IKT

Na základe sylabu tento modul je zameraný na pochopenie kľúčových princípov bezpečného používania IKT v každodennom živote, využívanie zodpovedajúcich techník a aplikácií na udržiavanie bezpečného pripojenia do počítačovej siete, spoľahlivé a bezpečné využívanie internetu a zodpovedajúce spracovanie údajov a informácií. Je to mierne náročný modul, ktorý sa venuje praktickým aj teoretickým znalostiam [2]. Modul M12 je pomerne nový, lokalizovaný bol v roku 2013, kedy sa ukázala potreba komplexnejšie sa venovať problematike bezpečnosti pri využívaní IKT. Aj vzdelávací štandard predmetu informatika obsahuje oblasť Informačná spoločnosť – bezpečnosť a riziká, kde sa žiaci oboznamujú so základmi bezpečnosti pri práci s IKT a rozličnými rizikami. Tento modul považujeme za veľmi dôležitý z hľadiska kybernetickej bezpečnosti.

Úspešný absolvent skúšky by mal mať znalosti podľa nasledujúcich kategórií uvedených v sylabe [2]:

1. **Pojmy z oblasti informačnej bezpečnosti** - základné pojmy z oblasti informačnej bezpečnosti - ohrozenia údajov, osobnej bezpečnosti, bezpečnosti súborov
2. **Škodlivý softvér (malware)** - ochrana počítača, prislúchajúcich zariadení, počítačovej siete pred škodlivým softvérom a neautorizovaným prístupom
3. **Bezpečnosť počítačových sietí** - rozpoznávať typy počítačových sietí a pripojení a rozumieť špecializovaným pojmom z ich ochrany
4. **Bezpečná práca s webom** - zásady bezpečného pohybu po internete a komunikácie na internete
5. **Komunikácia** - bezpečnostné otázky z oblasti elektronickej komunikácie vrátane elektronickej pošty a komunikácie v reálnom čase
6. **Bezpečná správa údajov** - správne zálohovanie, obnovovanie a bezpečná likvidácia údajov

Tabuľka 3: Analýza chybovosti podľa kategórií v module M12

Položka sylabu	Oblasť znalostí a zručností	Počet osôb	Počet úloh	Chyba [%]
12.5.2	Komunikácia v sieti v reálnom čase (Instant Messaging, IM) (Komunikácia)	122	244	39.8
12.4.1	Prezeranie webových stránok (Bezpečná práca s webom)	241	1219	30.4
12.1.3	Osobná bezpečnosť (Pojmy z oblasti informačnej bezpečnosti)	241	666	26.4
12.1.4	Bezpečnosť súborov (Pojmy z oblasti informačnej bezpečnosti)	193	193	25.1
12.2.3	Ochrana (Škodlivý softvér (malware))	241	368	25.0
12.1.2	Hodnota informácie (Pojmy z oblasti informačnej bezpečnosti)	241	897	23.1
12.5.1	Elektronická pošta (E-mail) (Komunikácia)	241	596	22.7
12.1.1	Ohrozenie údajov (Pojmy z oblasti informačnej bezpečnosti)	188	445	22.5
12.2.1	Definície a funkcie (Škodlivý softvér (malware))	241	241	20.7
12.6.1	Bezpečnosť a zálohovanie (Bezpečná správa údajov)	241	543	20.1
12.2.2	Typy (Škodlivý softvér (malware))	241	409	14.7
12.3.3	Bezpečnosť bezdrôtovej siete (Bezpečnosť počítačových sietí)	241	418	14.5
12.3.1	Počítačové siete (Bezpečnosť počítačových sietí)	241	552	13.8
12.4.2	Sociálne siete (Bezpečná práca s webom)	241	241	12.0

12.6.2	Bezpečná likvidácia (Bezpečná správa údajov)	182	182	9.3
12.3.2	Pripájanie sa k sieti (Bezpečnosť počítačových sietí)	188	257	7.0
12.3.4.	Riadenie prístupu k sieti (Bezpečnosť počítačových sietí)	182	300	5.3

Modul M12 je viac teoretický, obsahuje približne 72 % teoretických otázok a iba 18 % praktických úloh. V rámci teoretických otázok, podobne ako v module M2, sa striedajú formulácie rôzneho typu, takže na žiakov sú kladené vysoké nároky na čítanie s porozumením a dostatočne rýchly výber správnej odpovede. Je to ešte o to ťažšie, že v tomto module sa vyskytuje veľa anglických pojmov, rôzne skratky vytvorené z anglických slov a tiež náročnejšie slovenské odborné pojmy, ktoré si žiaci môžu ľahko navzájom „vymeniť“. Referenčná príručka a študijné materiály dostupné vo vzdelávacom portáli [4] vysvetľujú jednotlivé pojmy, avšak tu odporúčame o týchto pojmoch diskutovať, nespoliehať sa na to, že žiaci prečítanému porozumeli. Je dôležité využívať také metódy osvojovania týchto teoretických pojmov, ktoré aktivizujú žiakov, napr. kartičková hra – jednotlivé pojmy sú napísané na kartičkách, každý žiak si vytiahne kartičku a vysvetlí na nej napísaný pojem a začne tak diskusiu k uvedenému pojmu. Je dobré, ak konzultant tu upozorní tiež na zákon o ochrane údajov, ktorý je na Slovensku platný od roku 2018 [5].

Výsledky testovania sú uvedené v Tabuľke 3. Analýza podľa kategórií ukázala, že najhoršie (chybovosť viac ako 30 %) dopadli súbor zručností *Komunikácia v sieti v reálnom čase (Instant Messaging, IM)* z kategórie **Komunikácia** – chybovosť 39.8 %, t. j. z 244 teoretických úloh bolo až 97 úloh zodpovedaných nesprávne, a súbor zručností *Prezeranie webových stránok* z kategórie **Bezpečná práca s webom** – chybovosť 30.4 %, t. j. z 1 249 teoretických aj praktických úloh bolo nesprávne zodpovedaných, resp. riešených až 371 úloh. V rámci praktických úloh práve v tomto súbore zručností mali mnohí žiaci problém s rôznymi nastaveniami webového prehliadača, odstránením súkromných údajov z webového prehliadača, nastavení pre cookies. V kategórii **Komunikácia** v súbore zručností *Elektronická pošta (E-mail)* im zase robili problémy praktické úlohy súvisiace s digitálnym podpisom.

Priemerná chyba cez všetky kategórie je 20 %, teda až 1554 úloh z celkového počtu 7771 bolo zodpovedaných nesprávne.

4 ZÁVER

Analýza chybovosti podľa jednotlivých kategórií v rámci všetkých modulov (aj výsledky modulov uvedených v prílohe v Tabuľkách 5-8) nám ukázala, ktoré zručnosti a schopnosti v jednotlivých moduloch naši žiaci stredných škôl zvládajú veľmi dobre a kde sú medzery, ktoré majú prepojenie aj na výkonový a obsahový štandard vzdelávacieho štandardu predmetu informatika na základných a stredných školách. Modul M9 je najmenej vyberaným modulom, má priemernú chybovosť 34.5 %, preto príprave žiakov na testovanie v tomto module je potrebné venovať veľkú pozornosť.

Pri príprave na teoretické otázky je dôležité využívať také metódy osvojovania teoretických pojmov, ktoré aktivizujú žiakov a striedať rôzne formulácie otázok (testové úlohy so záporom, testové úlohy s grafickými prvkami, poradie správnej

alternatív, ...). Zároveň sa ukázalo, že žiaci majú často problém čítať zadania s porozumením a riešiť úlohu presne podľa zadania. Taktiež zvládnutie časového rámca testu je dôležitou súčasťou prípravy na samotné testovanie. Príprava žiakov na testovanie, nad rámec vzdelávacieho štandardu predmetu informatika, formou samoštúdia (e-learning) a následných konzultácií je preto veľmi dôležitá. K úspešnému zvládnutiu testov by mohli prispieť okrem vzdelávacieho portálu IT Akadémie aj príručky a nahrávky webinárov uvedené v bibliografii [6-12]. Uvedené príručky sú zdrojom ďalších informácií. Nahrávky obsahujú analýzy vzorových testov s upozornením na najčastejšie chyby.

Zaradenie systému ICDL/ECDL do ponuky pre žiakov SŠ a študentov VŠ bolo krokom k štandardizácii znalostí a zručností v IT na Slovensku, vzhľadom na to, že systém sa v nasledujúcich rokoch stal napr. v UNESCO Institute for Statistics jedným z celosvetových etalónov pre porovnávanie vzdelávacích obsahov IT v Ázii a Afrike. V roku 2017 získal pečať kvality a bol zaradený do ponuky vo vzdelávacom systéme v USA, stal sa súčasťou viacerých národných kvalifikačných rámcov v Európe i vo svete a tiež je v niektorých štátoch ponúkaný ako jedna z možností pre praktickú časť maturitnej skúšky z IT.

POĎAKOVANIE

Tento článok bol vytvorený v rámci národného projektu IT Akadémie – vzdelávanie pre 21. storočie, ktorý sa realizuje vďaka podpore z Európskeho sociálneho fondu a Európskeho fondu regionálneho rozvoja v rámci Operačného programu Ľudské zdroje.

BIBLIOGRAFICKÉ ODKAZY

- [1] VELŠIČ, M. *Digitálna gramotnosť na Slovensku, Inštitút pre verejné otázky*, 2020, https://www.ivo.sk/buxus/docs//publikacie/subory/Digitalna_gramotnost_2020.pdf
- [2] ECDL na Slovensku, Slovenská informatická spoločnosť, <https://www.ecdl.sk/>
- [3] Národný projekt IT Akadémie – vzdelávanie pre 21. storočie, 2017-2021, <https://itakademia.sk/>
- [4] Národný projekt IT Akadémie – vzdelávanie pre 21. storočie, vzdelávací portál, 2017-2021, <https://vzdelavanie.itakademia.sk>
- [5] Zákon o ochrane osobných údajov a o zmene a doplnení niektorých zákonov, 2018, https://www.slovlex.sk/static/pdf/2018/18/ZZ_2018_18_20190901.pdf
- [6] BARVÍŘ, T., MELIŠOVÁ, Š., HAMPL, J. *ECDL – manuál pro začátečníky a příprava ke zkouškám*, GRADA, 2011.
- [7] LAPÁČEK, J., SIMR, P., DANNHOFFEROVÁ, J., SÝKOROVÁ, K. *ECDL – Průvodce přípravou na testy*, Computer Press, 2016.
- [8] PECINOVSKÝ, J., PECINOVSKÝ, R. *Office 2016 Průvodce uživatele*. Grada Publishing, a.s., Praha, 2016.
- [9] KLATOVSKÝ, K. *Word 2016 nejen pro školy*. In Computer Media s.r.o. 2016.
- [10] ŽITNIAK, J. *Microsoft Office 2016 Podrobná uživatelská příručka*. Computer Press, a.s. Brno, 2017.
- [11] NĚMEC, P. *GIMP 2.8 – Uživatelská příručka pro začínající grafiky*. Computer Press, Brno, 2013.
- [12] Blichová, S., Kalakay, R. *Nahrávky webinárov s analýzou vzorových testov*, 2020, <https://portal.ccvapp.upjs.sk/videoarchiv/category/17>

Príloha 1. Chybovosť v ďalších moduloch

Tabuľka 4: Chybovosť podľa kategórií v module M4 – Tabuľkový kalkulátor

Položka sylabu	Oblasť znalostí a zručností	Počet osôb	Počet úloh	Chyba [%]
4.7.2	Kontrola a tlač (Príprava výstupov)	981	1224	34.8
4.7.1	Nastavenia (Príprava výstupov)	982	1720	29.4
4.3.1	Riadky a stĺpce (Práca s pracovnými hárkami)	982	1476	23.5
4.4.2	Funkcie (Vzorce a funkcie)	982	4059	23.4
4.2.1	Vkladanie, vyberanie, označovanie (Bunky)	982	1462	21.4
4.4.1	Aritmetické vzorce (Vzorce a funkcie)	982	4779	19.3
4.5.1	Čísla / dátumy (Formátovanie)	982	1718	18.7
4.5.2	Obsah (Formátovanie)	982	1721	18.0
4.5.3	Zarovnávanie, efekty orámovania (Formátovanie)	982	1587	13.5
4.6.2	Upravovanie (Grafy)	982	2464	12.8
4.6.1	Vytváranie (Grafy)	982	1952	12.7
4.2.2	Upravovanie, usporiadanie (Bunky)	982	2221	12.3
4.1.1	Práca s tabuľkovým kalkulátorom (Používanie aplikácie)	982	1239	11.0
4.3.2	Pracovné hárkky (Práca s pracovnými hárkami)	754	1123	11.0
4.2.3	Kopírovanie, presúvanie, vymazávanie (Bunky)	982	2340	10.6
4.1.2	Zvyšovanie produktivity (Používanie aplikácie)	737	737	9.1

Tabuľka 5: Chybovosť podľa kategórií v module M5 – Používanie databáz

Položka sylabu	Oblasť znalostí a zručností	Počet osôb	Počet úloh	Chyba [%]
5.1.3	Vzťahy (Databázová filozofia)	52	52	53.8
5.3.3	Vzťahy (Tabuľky)	65	82	51.5
5.6.2	Tlač (Výstupy)	38	38	46.1

5.4.2	Dotazy (Získavanie informácií)	65	407	37.3
5.6.1	Výstupy (Výstupy)	65	243	37.2
5.5.1	Používanie formulárov	65	227	33.5
5.4.1	Základné operácie Získavanie informácií	65	105	31.4
5.1.2	Organizácia databázy (Databázová filozofia)	65	213	29.6
5.2.2	Základné operácie (Používanie aplikácie)	65	133	28.9
5.3.2	Návrh (Tabuľky)	65	333	21.9
5.1.1	Základné pojmy (Databázová filozofia)	65	105	21.9
5.3.1	Záznamy (Tabuľky)	65	65	10.8
5.2.1	Práca s databázou (Používanie aplikácie)	48	65	7.7

Tabuľka 6: Chybovosť podľa kategórií v module M6 - Prezentácia

Položka sylabu	Oblasť znalostí a zručností	Počet osôb	Počet úloh	Chyba [%]
6.2.3	Predloha pre snímku (Vytváranie prezentácie)	2250	5089	38.1
6.1.2	Zvyšovanie produktivity (Používanie aplikácie)	300	300	33.7
6.6.2	Kontrola a samotná prezentácia (Príprava výstupov)	1974	2567	31.0
6.2.1	Zobrazenia prezentácií (Vytváranie prezentácie)	1203	1203	27.6
6.6.1	Príprava (Príprava výstupov)	2250	6816	19.2
6.5.1	Vkladanie, upravovanie (Grafické objekty)	2250	8447	18.1
6.4.1	Používanie grafov (Grafy)	2250	6803	14.6
6.5.2	Kreslenie (Grafické objekty)	2250	7302	14.1
6.4.2	Organizačné diagramy (Grafy)	2250	2750	12.8
6.1.1	Práca s prezentáciami (Používanie aplikácie)	2250	3074	12.7
6.2.2	Snímky (Vytváranie prezentácie)	2250	9537	11.9
6.3.1	Práca s textom (Text)	2250	6178	11.3
6.3.2	Formátovanie (Text)	2250	7074	10.9
6.3.3	Tabuľky (Text)	2250	3911	10.3

**Tabuľka 7: Chybovosť podľa kategórií v module M7 –
Základy práce online**

Položka syllabu	Oblasť znalostí a zručností	Počet osôb	Počet úloh	Chyba [%]
7.5.5	Používanie kalendárov (Používanie el. pošty (e-mailu))	1871	3742	34.4
7.1.1	Kľúčové pojmy (Pojmy z oblasti prezerania webu)	1871	2795	28.1
7.4.1	Virtuálne (online) komunity (Základné pojmy z komunikácie)	1871	2795	23.4
7.5.3	Prijímanie el. správy (e-mailu) (Používanie elektronickej pošty (e-mailu))	1871	4686	22.7
7.2.3	Záložky / Obľúbené položky (Prezeranie webových stránok)	1871	3762	21.2
7.3.3	Autorské práva (copyright) (Informácie z webových stránok)	950	1264	16.6
7.3.1	Vyhľadávanie (Informácie z webových stránok)	1871	3742	14.5
7.5.4	Organizovanie el. správ (Používanie elektronickej pošty (e-mailu))	1871	4074	14.4
7.1.2	Bezpečnosť a ochrana (Pojmy z oblasti prezerania webu)	1871	4997	14.1
7.5.2	Prijímanie elektronickej správy (e-mailu) (Používanie elektronickej pošty (e-mailu))	1871	4672	13.5
7.2.2	Nástroje a nastavenia (Prezeranie webových stránok)	1871	4050	13.0
7.4.3	Elektronické správy - základné pojmy (Základné pojmy z komunikácie)	1871	4375	12.5
7.5.1	Odosielanie el. správy (e-mailu) (Používanie elektronickej pošty (e-mailu))	1871	9023	12.4
7.3.2	Kritické hodnotenie (Informácie z webových stránok)	1576	1576	11.0

7.4.2	Komunikačné nástroje (Základné pojmy z komunikácie)	1871	1871	10.0
7.2.4	Výstupy z webu (Prezeranie webových stránok)	1871	2818	9.3
7.2.1	Používanie webového prehliadača (Prezeranie webových stránok)	1871	8408	3.1

**Tabuľka 8: Chybovosť podľa kategórií v module M9 –
Práca s obrázkami a grafikou**

Položka a syllabu	Oblasť znalostí a zručností	Počet osôb	Počet úloh	Chyba [%]
9.6.2	Tlač (Príprava výstupu)	80	80	54.4
9.1.3	Farebné modely/farebný priestor (Pojmy z oblasti zobrazovania)	100	281	51.6
9.4.1	Výber oblasti /výber (Práca s obrázkami)	100	248	47.2
9.4.5	Efekty a filtre (Práca s obrázkami)	100	371	44.3
9.6.1	Nastavenie (Príprava výstupu)	37	37	40.5
9.5.2	Nástroje na maľovanie (Kreslenie a maľovanie)	100	300	39.7
9.1.1	Digitálne obrázky (Pojmy z oblasti zobrazovania)	100	181	33.7
9.4.3	Vrstvy (Práca s obrázkami)	100	419	32.5
9.4.2	Manipulácia s obrázkami (Práca s obrázkami)	100	352	32.2
9.5.1	Nástroje na kreslenie (Kreslenie a maľovanie)	100	100	32.0
9.3.3	Zvýšenie produktivity (Používanie grafického editora)	88	88	31.5
9.3.2	Nastavenia (Používanie grafického editora)	80	80	31.2
9.4.4	Text-Práca s obrázkami	100	245	29.6
9.1.2	Grafické formáty-Pojmy z oblasti zobrazovania	85	109	24.8
9.3.1	Vytváranie obrázkov (Používanie grafického editora)	100	426	18.8
9.2.1	Získavanie obrázkov (Získavanie obrázkov)	100	155	7.1

Cestičky – programovanie pre predškolákov

Paths – Computing in Pre-primary Education

Daniela Bezáková

Katedra didaktiky matematiky, fyziky
a informatiky, FMFI UK

Mlynská dolina
842 48 Bratislava

Slovensko

daniela.bezakova@fmph.uniba.sk

Andrea Hrušecká

Katedra didaktiky matematiky, fyziky
a informatiky, FMFI UK

Mlynská dolina
842 48 Bratislava

Slovensko

andrea.hrusecka@fmph.uniba.sk

Roman Hrušecký

Katedra didaktiky matematiky, fyziky
a informatiky, FMFI UK

Mlynská dolina
842 48 Bratislava

Slovensko

roman.hrusecky@fmph.uniba.sk

ABSTRACT

In our paper we present an educational software Paths aimed for computing in pre-primary education. In Paths environment children can meet with controlling an agent by giving it instructions, with symbolic notation of given instructions and programming. In our paper we present the software itself and some pedagogical materials dealing with it: an exercise book for children and a methodology for teachers. We also write about some early experiments with the software and additional materials realized before and during the Covid19 pandemia. The Paths software is a part of upcoming conception of informatics from preschool to upper secondary school.

Keywords

Informatics. Kindergarten. Educational computing. Propaedeutic of programming.

ABSTRAKT

V článku predstavíme prostredie Cestičky určené na rozvoj algoritmickeho myslenia a propedeutiky programovania pre deti predškolskeho veku. V prostredí Cestičky sa deti stretnú s riadením vykonávateľa pomocou príkazov, „symbolickým“ zaznamenávaním vykonaných príkazov a plánovaním. V článku predstavíme samotný softvér a ďalšie materiály k nemu: pracovný zošit pre deti a metodiku pre učiteľov s návrhom vedenia jednotlivých stretnutí so softvérom a metodickými poznámkami k aktivitám v pracovnom zošite. V článku sa taktiež venujeme overovaniu softvéru a spomínaných materiálov v období pred pandémiou aj počas pandémie. Softvér Cestičky je súčasťou pripravovanej koncepcie informatiky od predškolskeho veku až po maturitu.

Kľúčové slová

Informatika. Materská škola. Edukačný softvér. Propedeutika programovania.

1 ÚVOD

V posledných dvoch desaťročiach bolo vyvinutých niekoľko nástrojov na podporu propedeutiky programovania pre deti z MŠ, a tiež vzniklo niekoľko štúdií o programovaní v predškolskom veku, napr. použitím Bee-Botov [1, 2], ScratchJr [3, 4] alebo KIBO [4, 5]. Programovateľného robota Bee-Bot asi predstavovať netreba. ScratchJr. [4, 6] na rozdiel od Scratchu obsahuje menšiu množinu blokov a program sa tvorí zľava doprava, nie zhora nadol. Deti môžu pridávať agentov, meniť ich zobrazenie, programovať ich správanie. KIBO [4] je robotická hračka, ktorú programujeme fyzicky spájaním priliehajúcich drevených kociek. Každá kocka má čiarový kód, ktorý zoskenujeme vstavaným skenerom robota. Po stlačení tlačidla potom robot zoskenovaný program vykoná.

Pozreli sme na možnosti, ktoré poskytujú tieto tri nástroje z pohľadu troch kritérií podľa prof. Kalaša [7]: *agent, ktorého riadime (reálny alebo virtuálny), druh ovládania (priama manipulácia, priame riadenie, programovanie) a interakcia (riadenie izolovaného agenta, alebo viacerých interagujúcich agentov)*. V spomínaných troch technológiách nájdeme reálnych (Bee-Bot, KIBO) aj virtuálnych (Scratch) agentov, riadenie jedného (Bee-Bot) aj viacerých agentov (Scratch) a takisto sú prítomné všetky tri druhy ovládania.

V rámci pripravovanej koncepcie informatiky sa snažíme o systematický prístup k procesu učenia sa programovania od predškolskeho veku až po maturitu. Pre tretí a štvrtý ročník ZŠ boli vyvinuté prostredia Emil 3 a Emil 4 s gradovanými úlohami, pracovnými zošitmi, metodikou a školeniami pre učiteľov ([8, 9]). Prostrediami Cirkus a Cestičky sme sa snažili vytvoriť akéhosi prirodzeného predchodcu robota Emila, vhodného pre deti predškolskeho veku, čo sa týka programátorských konceptov aj úrovne riadenia. Náš agent v prostredí Cestičky je virtuálnym izolovaným agentom, ktorého najskôr riadime priamo, neskôr jeho správanie programujeme.

2 CESTIČKY – SOFTVÉR A MATERIÁLY K NEMU

2.1 Prostredie

Edukačný softvér Cestičky je po Cirkuse [10] druhým softvérom pre materské školy v rámci spomínanej koncepcie informatiky od predškolskeho veku až po maturitu [11]. Prostredie Cestičky je určené na rozvoj algoritmickeho myslenia predškolákov. Je inšpirovaný predchádzajúcimi úspešnými projektami Cirkus šaša Tomáša a Výlety šaša Tomáša [12, 13] (na rozdiel od týchto softvérov je prispôbený tak, aby ho bolo možné používať na rôznych zariadeniach).



Obrázok 1 - Prostredie Cestičky

Ústrednou postavou prostredia Cestičky je už známy robot Emil, ktorý šoféruje autíčko po farebných cestách farmy. Emil na svojej ceste navštevuje zvieratká a v niektorých aktivitách môže na svojom autíčku prevážať mláďatká. Deti riadia Emila klikaním na farebné cestičky.

Cestičky sú rozdelené do piatich úrovní, ktoré sa líšia mapami (rozložením cestičiek, križovatiek, stanovišť so zvieratkami), spôsobe ovládania Emila a v prevážení zvieratiek. Prehľad odlišností jednotlivých úrovní nájdeme v Tabuľke 1.

Na každej mape je graf, ktorý má päť stanovišť so zvieratkami (listov) a niekoľko križovatiek. V križovatkách sa stretávajú vždy maximálne 4 cestičky, každá inej farby. Pri budovaní nejakej cesty vieme v každej križovatke podľa farby jednoznačne povedať, kade pokračovať.

V prvých štyroch úrovniach riadime Emila priamo, t.j. klikaním na symboly cestičiek vľavo v ponuke (nie priamo na cestičky v mape). Keďže zo žiadneho vrcholu nevedú dve cestičky rovnakej farby, je takéto ovládanie jednoznačné. V poslednej úrovni Emil spí a riadime ho nepriamo – najskôr mu pripravíme plán cesty, potom ho zobudíme a on postupuje podľa plánu.

Tabuľka 1 - Rozdiely medzi úrovňami¹

Úroveň	Mapa	Križovatky	Ovládanie	Prevážanie
1	bez cyklov ₂	3	priame	iba Emil
2	bez cyklov	3	priame	aj mláďatko
3	cyklus	4	priame	iba Emil
4	cyklus most	4	priame	aj mláďatko
5	cyklus most	4	plánovanie	iba Emil

Príkazy, ktoré Emilovi dávame, sa zaznamenávajú v hornej časti obrazovky.

V úrovniach, ktoré umožňujú prevážanie mláďatiek, pribudnú pri stanovištiach zvieratiek nástupištia a v ponuke tlačidlo s padákom, pomocou ktorého si privoláme mláďatko. Mláďatko prileti v niektoré stanovište so zvieratkami. Emil musí najskôr mláďatko vyzdvihnúť (mláďatko automaticky naskočí na auto, ak je auto prázdne) a potom ho zaviesť tam, kam chceme. Ak je stanovište prázdne, mláďatko automaticky vyskočí z auta. V zázname vidno, či išiel Emil sám alebo s mláďatkom.



Obrázok 2 – Ukážka záznamu s prevážaním mláďatka

Ďalšou postavičkou v programe je krtko, ktorý na cestách kope jamy. Tým môžeme ovplyvňovať riešiteľnosť úloh, prípadne deti prinútiť uvažovať nad iným riešením.

Rovnako ako softvér Cirkus aj Cestičky sú navrhnuté tak, aby s ním pracovali deti pri interaktívnej tabuli v malej skupine. Riešenie detí

program nekontroluje (ani keď sa prevádzajú mláďatká), kontrola sa robí iba diskusiou a argumentáciou detí medzi sebou a učiteľkou.

2.2 Pracovný zošit

Okrem samotného softvéru sme pre deti pripravili pracovný zošit, ktorý obsahuje päť sád aktivít (päť pracovných „listov“), každá sada k jednej úrovni softvéru Cestičky. Súčasťou pracovného zošita je aj päť máp a obrázky zvieratiek zo softvéru Cestičky. Keďže aktivity v pracovných listoch nadväzujú na aktivitu v príslušnej úrovni softvéru s interaktívnou tabuľou, odporúčame ich riešiť do týždňa po príslušnom stretnutí so softvérom, aby deti nezabudli, čo v príslušnej úrovni softvéru robili.

Úlohy v pracovnom zošite sa riešia rôznymi spôsobmi, napr. nalepením obrázka na dané miesto, dokreslením farebnými ceruzkami, zakrúžkovaním, nakreslením bodiek (keďže deti ešte nemusia vedieť písať čísla) atď.

Každé dieťa by malo mať vlastný pracovný zošit. Väčšinu úloh rieši každé dieťa samostatne, niektoré úlohy budú riešiť vo dvojiciach (prípadne ako skupina). Aj v takom prípade si však riešenie do pracovného zošita zaznačí každé dieťa.

Zadanie by mal deťom prečítať a vysvetliť učiteľ. Väčšina aktivít sa skladá z viacerých rovnakých podúloh, pričom prvá z nich je už vyriešená a slúži ako vzor. Túto riešenú úlohu môže učiteľ využiť na vysvetlenie zadania, pričom riešenie nepredkladá, ale po vysvetlení zadania deti požiada, aby dané riešenie skontrolovali, čím si zároveň overí, že ho pochopili. Po vyriešení každej aktivity (jej jednotlivých podúloh) si deti spoločne porovnávajú riešenia a diskutujú o nich. Až potom prechádzajú na ďalšiu aktivitu.

2.3 Metodický materiál

Pre učiteľov MŠ sme pripravili metodický materiál k softvéru Cestičky. Obsahom tohto materiálu je:

- stručné predstavenie softvéru,
- návrh na organizáciu práce so softvérom,
- metodický postup vedenia piatich stretnutí (vyučovacích jednotiek) so softvérom s využitím interaktívnej tabule: ich ciele, návrh aktivít, otázky na diskusiu a pod.,
- metodické komentáre k úlohám z pracovných listov.

3 AKTIVITY

3.1 Aktivity pri interaktívnej tabuli

Cieľom **prvého stretnutia** s prostredím je zoznámiť deti s robotom Emilom, preskúmať prvú úroveň softvéru (čo vidíme na mape, čo môžeme robiť, na čo sa dá klikáť a pod.) a riadiť Emila tak, aby navštívil jednotlivé zvieratká na mape. Mali by si začať uvedomovať súvislosť medzi a) príkazmi, ktoré dávame Emilovi, b) symbolmi, ktoré sa zapisujú do záznamu a c) cestičkami, po ktorých Emil išiel. Učíme deti záznam čítať a zistiť z neho základné informácie, ako kde Emil štartoval, kde skončil, po akých cestičkách išiel.

Počas **druhého stretnutia** sa deti (skúmaním) oboznámia s tým, ako funguje prevážanie zvieratiek. Budú prevážať mláďatká k ich rodičom, naučia sa v zázname rozlišovať časť cesty bez mláďatka a časť trasy s mláďatkom, vyčítať zo záznamu podrobnejšie

postupnosti cestičiek chodiť „dokola“, v mape bez cyklu takáto postupnosť nie je.

¹ Výrazy, ktorými popisujeme prostredie v tomto článku, sa deti nedozvedia.

² Pojmom cyklus myslíme cyklus v grafe, ktorý reprezentuje mapu, t.j. v prípade mapy s cyklom môžeme po istej

informácie o Emilovej ceste. Taktiež budú skúmať, kam sa dokáže Emil dostať a kam nie, ak niektorú cestu zničí krtko.

Na **treťom stretnutí** sa deti stretnú s mapou s cyklom. Cieľom stretnutia je uvedomiť si, že sa vieme dostať od zvieratka k zvieratku rôznymi spôsobmi, rozlišovať, kedy ideme po väčšom a kedy po menšom počte cestičiek. Aj v tomto stretnutí budeme skúmať, kam sa Emil vie dostať a kam nie, ak krtko zničí niektorú cestu.

Na **štvrtom stretnutí** budeme s Emilom zase preväzovať zvieratká, na mape s cyklom pritom pribudol aj most. Najskôr budeme trasu, ktorou pôjde Emil, voliť ľubovoľne (zvolia si ju deti) a po vyložení mláďatka (alebo aj pribežne) ju budeme skúmať – počet cestičiek, farby cestičiek, či šiel po moste alebo popod most. Toto skúmanie budeme robiť pre celú trasu, ale aj samostatne pre trasu bez mláďatka a trasu s mláďatkom. Neskôr budeme voliť trasu pre Emila tak, aby spĺňala zadanú požiadavku (napr. daný počet cestičiek, nesmie ísť popod most, musí ísť s mláďatkom po červenej a pod.). Nepovinnou doplnujúcou aktivitou je preväzovanie zvieratiek za predpokladu, že krtko nám zničí až dve cestičky.

Počas **piateho stretnutia** prejdeme k plánovaniu Emilovej cesty. V prvej aktivite len skúmame, ako teraz budeme riadiť Emila. V nasledujúcich tvoríme plány cesty na navštívenie jednotlivých zvieratiek, najskôr bez obmedzenia, neskôr pridáme na trasu nejakú požiadavku. Nakoniec skúsime vytvoriť plán takej cesty, počas ktorej Emil navštívi až dve zvieratká.

3.2 Aktivity v pracovných listoch

Úlohy v jednotlivých pracovných listoch nadväzujú svojim obsahom aj náročnosťou na jednotlivé stretnutia so softvérom. V tejto časti nebudeme popisovať podrobne obsah jednotlivých pracovných listov ani gradáciu úloh, uvedieme len typy úloh.

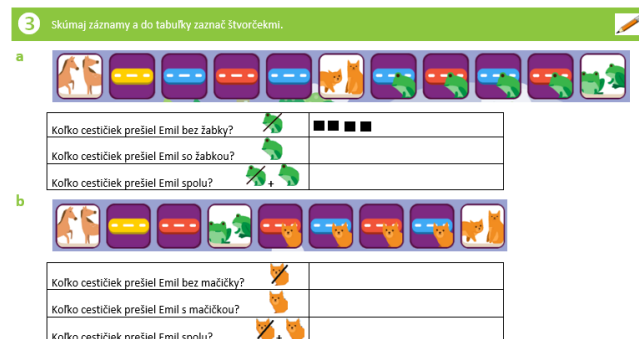
Prvým typom aktivít sú úlohy, v ktorých deti dostanú neúplný záznam a podľa mapy hľadajú odpovede. Sem zaradíme úlohy, v ktorých majú deti určitú chýbajúcu štart, resp. cieľ cesty alebo chýbajúcu cestičku, neskôr dokreslíť celú postupnosť cestičiek (prípadne rôzne postupnosti) pre daný štart a cieľ, alebo nájsť k danej ceste spätočnú cestu. K náročnejším patria úlohy, kde majú deti zistiť, či daný záznam popisuje správnu cestu, nájsť chybu v zázname a prípadne ju opraviť.



Obrazok 3 – Ukážka úlohy z pracovného zošita

Ďalší typ aktivít tvoria také, kde deti skúmajú len záznam a mapu nepotrebujú vôbec (prípadne ju môžu využiť ako pomôcku). V úlohách tohto typu zisťujeme zo záznamu rôzne informácie, ako

napr. počet cestičiek po ktorých šiel Emil, počet cestičiek konkrétnej farby, alebo naopak po ktorej cestičke šiel Emil daný počet krát. V záznamoch s preväzanim mláďatka hľadáme podobné informácie samostatne pre časť cesty bez mláďatka a časť cesty s mláďatkom. Deti sa tu stretnú aj so zapisovaním informácií do tabuliek. Z kompletného záznamu tiež určujeme, pri ktorom zvieratku Emil štartoval, pri ktorom vyzdvihol mláďatko a kde ho vyložil.



Obrazok 4 – Ukážka úlohy z pracovného zošita

V treťom type aktivít hľadajú deti riešenie len pomocou mapy. Ide o úlohy ako napr. nájsť zvieratká, od ktorých vedie cestička danej farby, či je možné dostať sa od daného zvieratka k danému bez použitia cestičky konkrétnej farby alebo ku komu sa z daného štartu Emil dostane, ak pôjde práve po troch cestičkách.



Obrazok 5 – Ukážka úlohy z pracovného zošita

4 OVEROVANIE

4.1 Overovanie pred Koronou

Overovanie softvéru, aktivít pri interaktívnej tabuli, aj papierových aktivít, sme začali v jednej materskej škole v Bratislave s deťmi predškolského veku. Takéto stretnutia s deťmi sme absolvovali tri, viac nám nedovolila aktuálna situácia. Všetkých stretnutí sa zúčastnili traja členovia nášho tímu (ďalej výskumníci) a učiteľka príslušnej triedy. Samotné overovanie a aktivitu pri interaktívnej tabuli viedol člen nášho tímu.

Prvé stretnutie prebehlo cez prázdniny a bolo zamerané predovšetkým na samotný softvér, jeho funkčnosť, ovládanie a na orientáciu detí na rôznych mapách. Zúčastnili sa ho dve štvorčlenné skupiny detí, ktoré po prázdninách nastupovali do základnej školy. Takúto vzorku sme vybrali cielene. Keďže sme po prázdninách chceli pokračovať v overovaní aktivít, nechceli sme vybrať deti, ktoré by sa mohli potenciálne týchto overovaní zúčastniť a boli by tak ovplyvnené týmto stretnutím.

Ďalšie dve stretnutia prebehli na začiatku školského roka. Prebiehali našim overeným spôsobom z overovania Cirkusu. Zúčastnilo sa ich približne 20 detí, väčšina v predškolskom veku.



Obrázok 6 - Overovanie jednej z prvých verzií prostredia

Druhá a tretie stretnutie sme začali úvodnou diskusiou so všetkými deťmi, potom sme pracovali v skupinách a na záver sme sa opäť všetci zišli pri záverečnej diskusii. Deti boli rozdelené do štyroch skupín, ktoré sa striedali pri jednotlivých aktivitách. Jedna aktivita bola so softvérom pri interaktívnej tabuli, druhá za stolom, kde deti riešili papierové úlohy zamerané na prácu s mapou a cestičkami (podľa pripravovaného pracovného zošita). Zvyšné dve aktivity pripravila pani učiteľka a týkali sa aktuálnej témy, ktorú práve preberali v materskej škole.

Na druhom stretnutí sme overovali prvú aktivitu s jednoduchou mapou a bez prevážania zvieratiek. Deti so zadanými úlohami nemali problém. Pri práci na úlohách si však často nevšimli zaznam cestu, ktorý sa vytvára, preto sme niektoré úlohy zadávali tak, aby odpoveď na ne museli deti zistiť práve zo záznamu.



Obrázok 7 - Overovanie druhej aktivity

Ani na treťom stretnutí, na ktorom už Emil prevážal aj zvieratká, deti s úlohami nemali problémy. Občas sa však stalo, že na vyriešenie úlohy im nestačil záznamový riadok. Bolo to vtedy, keď sa po ceste pomýlili a museli niektorý úsek cesty opakovať.

Papierové aktivity deti robili podľa vytlačenej mapy. Nemali s nimi väčšie problémy. Na mape sa dobre orientovali. Asi najväčším problémom bola pozornosť detí. Občas sa stalo, že cestu, ktorú mali do záznamu doplniť, nalepili do iného zadania a nie toho, ktoré práve riešili.



Obrázok 8 - Papierové aktivity

Celkovo deti so samotným prostredím nemali problémy, niektoré automaticky začali Emila ovládať pomocou cestičiek v ponuke (niektoré z týchto detí už mali skúsenosť s Výletmi šaša Tomáša), niektoré na to postupne prišli (skúšali klikat' na cesty v mape, na križovatky, prípadne ťahať Emila po ceste). Jednu skupinu sme museli naviesť na to, že na ľavej strane je panel, ktorý môžu využiť. Po tom už nebol s ovládaním softvéru žiaden problém.

Navrhnuté ovládanie softvéru nebolo nutné meniť. Po prvom overovaní sme si ujasnili, aké zložité mapy sú vhodné. Príliš veľa križovatiek znamená príliš dlhý záznam krokov, ktoré sa nemusia zmestiť do záznamového riadka. V mapách, ktoré sa využívajú, preto máme maximálne štyri križovatky.

4.2 Overovanie počas Korony

Pandemická situácia nám nedovolila pokračovať v overovaní v materskej škole, preto sme museli zmeniť spôsob overovania. Keďže sme už mali skúsenosti z overovania softvéru Cirkus a aj nejaké z overovania Cestičiek, mali sme predstavu, čo deti zvládajú. Pri tomto druhom overovaní sme sa hlavne zamerali na porozumenie zadaniu úloh, ktoré by sa mali robiť pri interaktívnej tabuli a následne úloh v pracovnom zošite.

Overovanie sme robili on-line s jednou predškoláčkou z našej testovacej materskej školy, ktorá so softvérom pracovala na počítači. Absolvovala overovanie v materskej škole, teda prvé dve úrovne softvéru jej boli relatívne známe, aj keď medzi týmito overovaniami ubehol dlhší čas. Metodika stretnutí a úlohy v pracovnom zošite sa medzitým tiež mierne zmenili.

S predškoláčkou sme absolvovali päť online-stretnutí tak, ako boli navrhnuté v metodike. Na každom stretnutí najskôr pracovala so softvérom Cestičky a hneď potom riešila úlohy z pracovných listov – už bez použitia softvéru (potrebnú mapu mala k dispozícii). Stretnutia viedol jeden z nášho výskumného tímu. So súhlasom rodičov sme nahrávali obrazovku aj zvuk.

Pri overovaní sme si všimli, že pri riešení úloh si záznam často nevšima a na otázky o prejdenej ceste odpovedá podľa toho, ako úlohu riešila. Ale v prípade, keď nevidela, kade Emil šiel (videla len hotový záznam), a potom mala zodpovedať na otázky ohľadom jeho cesty, vtedy už záznam využívala.

Pri mape s cyklom, ak sme cestu pokazili krtkom, jej nerobilo problém určiť, kam sa môže a kam sa nemôže Emil dostať. Takisto jej nerobilo problém, vymyslieť alternatívnu cestu.

Dobre tiež chápala a riešila úlohu s konkrétnou požiadavkou na cestu, hoci takáto cesta nebola najkratšia (v zmysle počtu cestičiek).

V piatej úrovni (plánovanie trasy) nerobilo veľký problém, ak dostala úlohu nielen sa dostať ku konkrétnemu zvieratku, ale aby aj po ceste robot Emil navštívil iné konkrétne zvieratko. Toto považujeme za najnáročnejšie v tejto úrovni, lebo celá trasa je už dosť dlhá (zvyčajne aspoň 8 cestičiek) a medzizastávku si deti musia uvedomiť.

Z úlohu v pracovnom liste jej robili najväčšie problémy úlohy, v ktorých bolo treba doplniť, kde Emil začínal. Tento typ úlohu je náročný v tom, že pokiaľ nemáte k dispozícii mapu, treba pozerať cestu od konca, a to jej evidentne nešlo. Ak však podobnú úlohu riešila s pomocou mapy, dokázala ju vyriešiť tak, že hľadala zvieratko, od ktorého ide cestička rovnakej farby, ako prvá cestička v zázname. Testerke taktiež robilo problémy nájsť chybu v zázname trasy, ak bola táto chyba bližšie k začiatku trasy – z jej reakcií bolo evidentné, že mala úplne inú predstavu o trase, ktorá nekorešpondovala s pokazenou cestou v zadaní.

5 Zhrnutie

Obe overovania nám ukázali, že skúmaním deti dokážu veľmi rýchlo odhaliť ovládanie softvéru a nerobí im žiadne komplikácie. V druhej a štvrtej úrovni (prevádzanie mláďat) bolo prirodzené, že mláďatko musíme zaviesť rodičom. Deti si postupne zvykali na záznam a kým v prvej úrovni ho pri odpovediach na otázky často ignorovali, v neskorších úrovniach ho už využívali.

Druhé overovanie nám pomohlo odhaliť niektoré nedostatky najmä v zaniatiach úloh v pracovných listoch. Výhodné bolo, že sme pripravené zadanie mohli hneď preformulovať a tak zistiť, ktorá formulácia je pre dieťa prístupnejšia. V niektorých prípadoch sme mali pripravené dve verzie rovnakej úlohy, ktoré sa líšili spôsobom zadávania odpovede a zistili sme, ktorý spôsob je pre dieťa vhodnejší či pochopiteľnejší. Taktiež sme zmenili poradie niektorých úloh na základe toho, aké náročné boli pre našu testerku, prípadne sme niektoré typy úloh úplne vynechali. Overili sme si tiež, že metóda *vysvetliť zadania a dať dieťaťu overiť predložené riešenie* bola dobrým nápadom. Naozaj sme pomocou toho zistili, či dieťa úlohu pochopilo alebo nie.

Pri viacerých úlohách si uvedomujeme, že nemôžeme zovšeobecňovať postrehy. Každé dieťa môže reagovať inak, a takisto deti od seba navzájom môžu „odkukávať“ a nie všetko sami objaviť.

Na záver by sme chceli dodať, že pri tvorbe metódik a úloh v pracovných listoch, ako aj pri samotnom overovaní, sme si dávali pozor na slovník, ktorým sa s deťmi rozprávame. Pojmy, ktoré sme používali v tomto článku ako graf, cyklus, či najkratšia cesta, sme s deťmi úmyselne nepoužívali.

POĎAKOVANIE

Chceli by sme sa poďakovať kolegom z nášho výskumného tímu (špeciálne Andrejovi Blahovi, Ivanovi Kalašovi a Milanovi Moravčíkovi) a našej testovacej materskej škole. Taktiež ďakujeme neziskovej organizácii Indícia a projektu VEGA 1/0602/20 za podporu nášho výskumu.

BIBLIOGRAFICKÉ ODKAZY

- [1] PEKÁROVÁ, J.: Using a Programmable Toy at Preschool Age: Why and How? In *Workshop Proceedings of SIMPAR 2008*. Venice, 2008, p. 112-121. ISBN 978-88-95872-01-8
- [2] PALMÉR, H.: Programming in Preschool --With a Focus on Learning Mathematics. In *International Research in Early Childhood Education*. 2017, vol. 8, no. 1, p. 75-87. ISSN 1838-0689.

- [3] FLANNERY, L-P., KAZAKOFF, E-R., BONTÁ, P., SILVERMAN, B., BERS, M-U., RESNICK, M.: Designing ScratchJr: support for early childhood learning through computer programming. In *Proceedings of the 12th International Conference on Interaction Design and Children*. New York : ACM, 2013, p.1–10.
- [4] BERS, M. U.: Coding, playgrounds and literacy in early childhood education: The development of KIBO robotics and ScratchJr. In: *2018 IEEE Global Engineering Education Conference (EDUCON)*. 2018, p. 2094-2102. ISSN: 2165-9567.
- [5] BERS, M.U., HORN, M.S.: Tangible programming in early childhood: Revisiting developmental assumptions through new technologies: In *Berson IR, Berson MJ, editors, High-Tech Tots: Childhood in a Digital World*. Greenwich, CT: Information Age Publishing, 2010. p. 49-70.
- [6] ScratchJr Homepage. <https://www.scratchjr.org>.
- [7] KALAŠ, I.: On the road to sustainable primary programming. In: *Proceedings of the Constructionism in Action: Constructionism*. Bangkok : Suksapattana Foundation, 2016, p. 184–191.
- [8] KALAŠ, I.: Programming in lower primary years: design principles and powerful ideas. In *Proc of Constructionism Computational Thinking and Educational Innovation*. Vilnius, 2018, p.71-80. ISBN 978-609-95760-1-5.
- [9] KALAŠ, I., BLAHO, A., MORAVČÍK, M.: Exploring Control in Early Computing Education. In: *Sergei N. Pozdniakov and Valentina Dagienė (eds.) Informatics in Schools. Fundamentals of Computer Science and Software Engineering*. Springer, LNCS 11169, p. 3-16. ISBN 978-3-030-02750-6.
- [10] BEZÁKOVÁ, D., HRUŠECKÁ, A., HRUŠECKÝ, R. Informatika v Cirkuse, In *Didinfo 2019*. Banská Bystrica 2019, p. 22-25. Dostupné tiež na https://io.fpv.umb.sk/didinfo/Zbornik_Didinfo_2019.pdf
- [11] Indícia. O informatike s indíciou. <https://www.indicia.sk/aktualne-skolenia/informatika-s-emilom/o-informatike-s-emilom>
- [12] KABÁTOVÁ, M., KALAŠ, I., TOMCCSANYIOVÁ, M.: Programming in Slovak Primary Schools. In *Olympiads in Informatics*. Vilnius, 2016, vol. 10. P. 125-159.
- [13] EDULAB. Výlety šaša Tomáša. <http://www.skolkahrou.sk/vylety-sasa-tomasa/>

Programovací jazyk v prostredí Emil pre 4. ročník

Programming language in Emil for Year 4

Andrej Blaho

KAI FMFI

Univerzita Komenského v Bratislave
Slovensko an-

drej.blaho@fmph.uniba.sk

Ivan Kaláš

KDMFI FMFI

Univerzita Komenského v Bratislave
Slovensko

ivan.kalas@fmph.uniba.sk

Milan Moravčík

EDIX, s. r. o.

Bratislava
Slovensko

milan.moravcik@edix.sk

ABSTRACT

Informatics in primary school is mandatory in Slovakia since 2008. Nevertheless, its *enacted curriculum* remains unsatisfying. Teachers are getting no pre-service informatics education nor any continuous in-service support. So, in the Computing with Emil project we set out to develop and verify a series of pedagogical interventions for each Year of informatics, starting in primary. Our overall intention is to support every pupil's systematic and constructivist learning process with an emphasis on programming and intense connections to mathematics and other subjects. For Years 3 and 4 we completed new interventions Emil 3 and Emil 4. In our paper we present in detail programming language of Emil 4. However, based on the analysis of Emil 4's design process we concluded that summarising the language with related computational concepts would provide only partial picture. Thus, we decided to include also *operations*, which pupils perform with the concepts, and their gradation, which reflects our classification of the operations' *cognitive demands*.

Keywords

programming at primary level, Computing with Emil, design research, computational concepts and constructs

ABSTRAKT

Informatika je na Slovensku povinným predmetom na 1. stupni ZŠ od roku 2008, ale jej skutočne realizovaný vzdelávací obsah¹ zostáva neuspokojivý. Učiteľia nedostávajú zodpovedajúce iniciálne vzdelávanie, ani priebežnú podporu. V projekte Informatika s Emilom sme si preto zaumienili vytvárať a overovať nový vzdelávací obsah, ktorý začne na 1. stupni a bude postupovať do vyšších ročníkov. Jeho cieľom je podporiť u každého žiaka v triede systematický a konštruktivistický poznávací proces s dôrazom na programovanie, s bohatým preháňaním s matematikou a ďalšími oblasťami. Pre 3. a 4. ročník sme vyvinuli úplné vzdelávacie obsahy (skrátene Emil 3 a Emil 4) a v tomto príspevku detailne predstavujeme programovací jazyk prostredia Emil 4. Na základe analýzy procesu nášho vývoja sme však usúdili, že prezentovať iba jazyk a koncepty, ktoré rozvíja, je nedostatočné. Zameriavame sa preto aj na operácie, ktoré s konceptami robia žiaci, a ich gradáciu, ktorá zodpovedá našim zisteniam o ich kognitívnej náročnosti.

Kľúčové slová

programovanie na 1. stupni ZŠ, informatika s Emilom, výskum vývojom, programovacie koncepty a konštrukty

1 ÚVOD

Roky so znepokojením registrujeme neuspokojivú a neriešenú situáciu s vyučovaním informatiky na 1. stupni ZŠ, a to predovšetkým v oblasti programovania. Aj keď je tu informatika povinný predmet už od roku 2008 a tej dobe zodpovedajúce národné kurikulum je z pohľadu algoritmického myslenia a programovania pomerne detailné, kvalitné a náročné, realita v príprave budúcich učiteľov a na školách tomu vôbec nezodpovedá. Podobnú situáciu však pozorujeme aj v iných krajinách, napr. Anglicku². Za najväčšiu slabinu reálneho informatického vzdelávania na 1. stupni považujeme postoj pedagogických fakúlt³, neexistujúcu podporu pedagógov a neexistujúci systematický vzdelávací obsah, opäť najmä v oblasti programovania. Preto sme sa v spolupráci s neziskovou organizáciou Indícia a ďalšími partnermi, napr. i našou fakultou, rozhodli skúsiť túto situáciu zmeniť, a to intenzívnou spoluprácou so základnými školami a dlhodobým vývojom vzdelávacieho obsahu a vzdelávania učiteľiek a učiteľov z praxe. Tento bežiaci projekt pod názvom Informatika s Emilom sme zahájili v apríli 2017. V decembri 2018 sme ukončili – a Indícia začala šíriť – nový vzdelávací obsah s názvom Emil 3 (pre 3. ročník⁴), v marci 2020 potom vzdelávací obsah Emil 4 (pre 4. ročník). V súčasnosti zahajujeme vývoj obsahu pre 2. stupeň. Jedným z určujúcich princípov nášho nadväzujúceho vývoja bolo rozhodnutie orientovať sa na „bežného“ primárneho pedagóga, teda neočakávať a nevyžadovať žiadne špeciálne informatické zameranie. Toto rozhodnutie sa nám dnes naďalej javí ako správne, praktické a produktívne.

2 VÝCHODISKÁ

Problémom, na ktorý sme sa rozhodli zamerať, je skutočnosť, že ani v našej, ani zahraničnej literatúre a reálnej situácii na školách nám nie je známy vyvinutý a overený pokus o vytvorenie *ucelenej postupnosti krokov pri budovaní základov programovania* pre túto vekovú skupinu, ktorý by a) rešpektoval vývinovú primeranosť obsahu, metód a používaných programovacích prostredí; b) efektívne využíval potenciál týchto prostredí (nie len, ale najmä) na budovanie informatického poznania a myslenia⁵, c) systematicky podporoval poznávací proces žiakov v informatike v rozsahu viacerých ročníkov (ako to bežne robí matematika), d) a to s jasným zameraním na formálne vzdelávanie a každého žiaka (dievča a chlapca) v triede a e) rešpektujúce primárneho pedagóga a učebnú klímu typickú pre primárne vzdelávanie.

¹ Anglicky *enacted curriculum*.

² Ich národné kurikulum je veľmi stručné, jasné a ešte ambicioznejšie ako naše – a to už v prvom cykle (pre žiakov vo veku 5 až 7 rokov), ešte viac v druhom cykle (pre žiakov od 7 do 12 rokov). Ich nový národný program vzdelávania vstúpil do platnosti v roku 2014, ale aj to je už sedem školských rokov.

³ Ten sa však začína meniť k lepšiemu.

⁴ Presnejšie: Emil pre 3. a 4. ročník a Emil pre 4. a 5. ročník.

⁵ Zdôrazňujeme to preto, že niekedy – napr. v prípade programovateľnej včely Bee-Bot – sa aktivity a ich metodické materiály (často veľmi dobré) zameriavajú najmä na využitie týchto zariadení v inom predmete, resp. skôr s dôrazom na medzipredmetové

Výskumníci sa často zameriavajú na rozvoj informatického myslenia alebo algoritmického myslenia a programovania v podmienkach mimo bežnej triedy, v kluboch, na krúžkoch či letných školách, pozri napr. [1], [2], prípadne sa zameriavajú na vyššie vekové skupiny, pozri [3], [4] a ďalšie. A pritom už poznáme: *...compelling reasons for mandatory computing beyond economics and employment. Access to a high-quality education can be seen as an equity issue*, ako píše Sentence [5] a nabáda, aby sme rozvíjali... *computing for all*. Presne to sa stalo pre nás v projekte Informatika s Emilom najväčšou výzvou, spolu so všetkými princípmi (a) až (e), ktoré sme spomenuli vyššie.

2.1 Koncepty a konštrukty

Ak sa výskumníci snažia implementovať takúto informatiku pre každého v triede, zvyčajne sa zameriavajú na skúmanie *programovacích konceptov* a ich *kognitívnej náročnosti* (podľa Blackwella [6] *cognitive demands*), pozri napr. [7], [8] alebo [3]. Perlman už v roku 1976 referovala o tom, aké... *stratené boli deti, keď sa oboznamovali s rôznymi konceptami... pri práci s naším výskumným programovateľným zariadením Tortis Slot Machine*. Poučená týmito pozorovaniami začala vo svojej práci skúmať práve kognitívnu náročnosť v pozadí rôznych aspektov programovania, pozri [10], [11]. Preto sme si v našom projekte kládli otázky ako: *Ktoré koncepty sú vývinovo primerané pre žiakov 3. a 4. ročníka? V akom poradí ich zaraďovať do nášho vzdelávacieho obsahu tak, aby si ich žiaci a ich učitelia osvojili a porozumeli im? Čo sú tie primerané malé kroky, ktoré máme v aktivitách implementovať, aby zvolené koncepty žiaci dokázali objavovať sami, bez výkladu učiteľa a aby si pritom konštruovali ich hlboké a trvácne porozumenie? V čom vlastne spočívajú tieto kroky?*

Podobnými otázkami sme sa zaoberali aj v predchádzajúcom projekte ScratchMaths, pozri napr. [11] či [8], kedy sme si uvedomili, že slovník, ktorý sa bežne používa pri definovaní vzdelávacích cieľov informatiky, a obzvlášť programovania, vo väčšine národných kurikul považujeme za „prihrubý“ na to, aby vyjadril drobné rozdiely v tom, ako si žiaci postupne rozvíjajú, teda graduujú svoje porozumenie týchto konceptov⁶. Dnes už napr. vieme, že použitie príkazu s vopred nastavenou hodnotou vstupu, ako napr. *opečiatkuj kruh*, je jednoduchšie a má predchádzať neskoršiemu objavu, že *opečiatkuj* môže použiť jeden z niekoľkých tvarov ako svoj vstup (parameter), z ktorých si žiak vie sám vybrať⁷. Preto venujeme v ostatných rokoch mimoriadnu pozornosť tomu, aby sme identifikovali a overili, čo je správna (určite jedna z viacerých) *gradácia operácií s daným konceptom*, ktoré postupne spoznáva. Odporúčame nevyberať v jednotlivých krokoch kurikul koncepty, ktoré sa majú žiaci naučiť, ale koncepty a s nimi súvisiace operácie, ktoré majú spoznať v danom vzdelávacom cykle a ktoré sú vývinovo primerané. Na označenie súboru konceptu a operácií, ktoré s ním žiaci majú vykonávať, sme začali používať označenie *konštrukt*. Práve to, že pri prezentovaní programovacieho jazyka Emila

aktivity, zvyčajne však nevyužívajú dostatočne ich potenciál pre rozvoj informatických cieľov.

⁶ Treba však poznamenať, že náš inovovaný Štátny vzdelávací program je v tomto kontexte výrazne novátorský, a to práve tým, že požiadavky na výkonový štandard pracujú práve s tým, čo chýba v národných kurikulumoch skoro bez výnimky až dodnes, a to je **pomenovanie úkonov či operácií**, ktoré sa majú žiaci naučiť s konceptami vykonávať. To, čo zrejme bude nasledujúca generácia nášho národného kurikula obsahovať tiež, je explicitné vyjadrenie gradácie kognitívnej náročnosti konceptov a ich operácií – založené na kvalitnej výskumnej báze a starostlivo overené.

⁴ považujeme za nevyhnutné pomenovať aj **gradáciu zvolených operácií**, považujeme za najdôležitejším výsledkom tohto príspievku.

3 PROJEKT INFORMATIKA S EMILOM

Ide o prebiehajúci výskumno-vývojový projekt, v ktorom sme doposiaľ ukončili vývoj vzdelávacieho obsahu Emil 3 a Emil 4, v súčasnosti dokončujeme ďalšie intervencie pre materské školy a zahajujeme vývoj pre 2. stupeň, ktorý plynulo nadviaže na predchádzajúce intervencie. Všetky ukončené a pripravované intervencie zapadajú do uceleného plánu na podporu systematického a atraktívneho poznávacieho procesu v školskej informatike, s dôrazom na programovanie ako nový silný nástroj žiakov na skúmanie sveta a jeho problémov a objavovanie *významných myšlienok* (aj) informatiky⁹. V tejto kapitole len stručne zhrnieme výstupy etapy Emil 3, ktoré sa zároveň stali aj východiskami pri vývoji obsahu a prostredia Emil 4. Ďalší kľúčový komponent, a to naše vývojové a pedagogické princípy, sme prezentovali v [12] a [13].

Za najdôležitejšie výstupy projektu z etapy Emil 3 považujeme:

- *komplexný vzdelávací obsah*, teda pedagogickú intervenciu, ktorá pozostáva z pracovného zošita pre žiaka, programovacieho prostredia Emil 3, detailnej metodologickej príručky pre učiteľov a ich vzdelávanie. Vzdelávací obsah tvoria tri *svety* s gradáciou úloh zorganizovaných do skupín, ktoré označujeme písmenami **A, B, C, ...** Väčšina úloh je úmyselne prezentovaná žiakom čiastočne v softvérovom prostredí a čiastočne v pracovnom zošite, teda ani softvér, ani zošit nie je sám o sebe použiteľný. Pomedzi spomínané skupiny úloh niekedy zaraďujeme v zošite aj skupiny ako **po A, po D, ...** Tieto obsahujú ďalšie úlohy na precvičenie skúmaných informatických javov, prípadne poslúžia učiteľovi¹⁰ ako účinný nástroj učiteľa na diagnostikovanie splnenia vzdelávacích cieľov. Označenie **po A** vyjadruje, že obsahuje papierové úlohy, ktoré môžu žiaci riešiť po ukončení aktivít v **B** a pod.,
- *gradácia programovacích konštruktov*, ktorá predstavuje rámec celej intervencie (niektoré z nich sme dokumentovali v [12] a [8]),
- *súbor významných myšlienok informatiky* (najmä programovania), s ktorými sa žiaci počas týchto ročníkov stretnú a ktoré budú v tomto prípade skúmať v Emilovi 3. Ide nám o to, aby sme všetkým žiakom v triede ako najvyššiu prioritu umožnili budovať si hlboké porozumenie¹¹ zodpovedajúcich konceptov a zvolených operácií, ktoré s nimi vykonávajú.

Chceme zdôrazniť, že v procese vývoja Emila 3 sme všetky tieto výstupy iteratívne a priebežne overovali prostredníctvom pozorovaní žiakov a ich učiteľov, rozhovorov so žiakmi a učiteľmi, a tiež analýzou žiackych výstupov vypracovaných pilotných verzií listov, ktoré sa po mnohých iteráciách stali východiskom pre výsledný

⁷ Rovnako ako vie inými príkazmi nastaviť farbu a veľkosť tohto tvaru.

⁸ Celé názvy intervencií sú Informatika s Emilom 3. Programovanie pre 3. a 4. ročník ZŠ a Informatika s Emilom 4. Programovanie pre 4. a 5. ročník ZŠ. Skráteno hovoríme Emil 3 a Emil 4.

⁹ Významných myšlienok v zmysle Papertových *powerful ideas* z jeho epochálnej knihy *Mindstorms*.

¹⁰ A celkom určite výskumníkom, pozri časť 5.2.

¹¹ v zmysle *... computational practices and perspectives* Brennanevej a Resnicka [7]

pracovný zošit. Tento príspevok si však zvolil užší cieľ, než analyzovať celý proces vývoja prostredia Emil 4 so všetkými jeho výstupmi, a to detailne prezentovať jeho výsledný programovací jazyk – a všetkého, čo k tomu považujeme za nevyhnutné. Nie je výskumná štúdiou, takže náš výskumný rámec predstavíme len celkom stručne.

3.1 Edukačný výskum vývojom

Vývojový proces intervencie Emil 3 sme zahájili v apríli 2017 a dokončili v decembri 2018. Dva mesiace nato sme začali s vývojom Emila 4, tento proces trval 13 mesiacov¹², teda do marca 2020. V oboch etapách sme k vývoju postúpili na báze *výskumu vývojom* či *edukačného výskumu vývojom*, pozri [14]. Aj keď tento príspevok má iné ciele než analyzovať náš vývoj z aspektu výskumného, považujeme za dôležité vysvetliť, že sme iteratívne postupovali podľa etáp výskumnej schémy (ibid.): (1) identifikácia a preskúmanie problému ➔ (2) analýza situácie a potrieb ➔ (3) návrh a vývoj intervencie ➔ (4) nasadenie a hodnotenie riešenia ➔ (5) revízia riešenia a ➔ návrat na (2) dovtedy, kým riešenie nenaplní podľa zvolených kritérií očakávania výskumníkov. Kľúčom ku kvalite takéhoto iteratívneho procesu sú práve kritériá, ktoré používame na priebežné hodnotenie riešenia, aby sme posúdili, či už môžeme ukončiť kruh iterácií alebo nie. S tým súvisí aj validita iteratívne zbieraných dát a postupov ich analýzy. V tomto príspevku sa nebudeme ďalej venovať prezentácii procesu a výstupov nášho vývoja v tomto kontexte, iba v časti 4.1 uvedieme typy dát, ktoré sme počas vývoja Emila 4 zbierali a ktoré usmerňovali naše revízie.

3.2 Programovanie v prostredí Emil 3

Ako sme spomínali skôr, Emil je zostavený z troch gradovaných svetov, každý z nich obsahuje svoju vlastnú gradáciu úloh organizovaných v skupinách **A**, **po A**, **B**,... Prvý svet ešte nemá explicitne vizualizovaný programovací jazyk, príkazy – obmedzené iba na riadenie pohybov Emila – sa implicitne a intuitívne realizujú klikaním (dotykmi prsta) priamo na Emilovu plochu s významom *sem príleť*, a to horizontálne alebo vertikálne. V druhom a treťom svete postupne budujeme elementárny jazyk, explicitne prezentovaný malou ponukou príkazov na tlačidlách. Tento jazyk pozostáva z príkazov *krok dohora* (o jedno políčko na ploche), *dolu*, *vpravo* a *vľavo* a malého počtu príkazov na zmenu stavu políčka (ako polej, pozbieraj, rozsvieť... alebo polož dielik daného tvaru – v treťom svete). Rôzne úrovne riadenia v prvých dvoch svetoch sme detailne analyzovali v [13]. Úplná analýza programovacích konceptov a zvolených operácií vo svetoch Emila 3 bude náplňou inej publikácie.

Vzhľadom na obmedzený rozsah sa teraz zameriavame výlučne na prezentáciu programovacieho jazyka v prostredí Emil 4 a potrebu úzkeho prepojenia takejto analýzy s gradáciami zvolených operácií, ktoré žiaci vo vzdelávacom obsahu Emila 4 skúmajú a spoznávajú. K tomu však potrebujeme ozrejmiť, čo v našom prístupe označujeme slovami *vlákna úloh*.

3.3 Koncepty, prekoncepty¹³ a vlákna úloh

Celý vzdelávací obsah Emila 3 a Emila 4 tvorí jedna starostlivo vyvinutá a overená gradácia úloh (s rastúcou mierou príležitosti pre skúmanie a objavovanie viacerých riešení, postupov a rozšírení).

V rámci tejto *veľkej gradácie* úloh¹⁴ ako autori obsahu vidíme rôzne *vlákna úloh*, teda rôzne pod sekvencie či výbery úloh, nie nutne nasledujúcich priamo za sebou, ktoré predstavujú prípravu (propedeutiku) smerujúcu k porozumeniu určitého konceptu. Pre ilustráciu, vzdelávacím obsahom Emil 3 sa ťahne napr. aj vlákno vedúce k používaniu a napokon aj vytváraniu zložených príkazov v Emilovi 4. Toto vlákno bude ďalej pokračovať používaním a vytváraním vlastných blokov v Scratchi a neskôr k používaniu a vytváraním funkcií v Pythone. Do nášho vzdelávacieho obsahu v Emilovi 3 a 4 sme zakomponovali viacero takýchto vlákien smerujúcich (v týchto a ďalších intervenciách) ku kľúčovým konceptom akými sú dáta, hodnoty a premenné, podmienky a podmienené vykonanie časti programu, výrazy, opakovanie s podmieneným počtom vykonaní, podprogramy (procedúry a funkcie) a pod.

Organizačný princíp každého vlákna (teda dôvod, prečo je zoradené práve tak a čo tvorí jeho elementárne kroky) vyplýva z nášho hodnotenia rastúcej kognitívnej náročnosti operácií, ktoré s konceptom alebo jeho prekonceptom žiaci vykonávajú. Výsledkom kognitívnej transformácie, ktorú má vlákno podporiť, je teda **koncept spolu s rôznymi operáciami s ním** (napr. iterácia a čo všetko s ňou robíme). Už vyššie sme tento celok označili ako *programovací konštrukt*. Ilustrujme tu jednu z gradácií operácií z Emila 3:

- Pri riešení úloh priamym riadením v Emilovi 3 žiaci v jednej etape narazia na situáciu, kedy sa obmedzený počet okienok na paneli so záznamom stáva faktorom, ktorý musia brať do úvahy a v určitom zmysle kvôli nemu optimalizovať svoje postupy. Napriek tomu zažijú situácie, kedy sa kvôli obmedzenému počtu krokov stáva úloha neriešiteľnou, pozri Obr. 1 (a), ak potrebujú urobiť ďalší krok.



Obrázok 1: Niekoľko ilustrácií panela s programom z druhého sveta Emila 3

- V ďalšej etape gradácie žiaci objavujú, že príkaz, ktorý opakovane použijú niekoľkokrát bezprostredne za sebou, sa na paneli so záznamom automaticky zhŕňa na kôpku, pozri Obr. 1 (b), čím sa môže podstatne znížiť celkový počet okienok potrebných na program s riešením.
- Ďalej žiaci riešia sériu úloh, pri ktorých získavajú skúsenosti s tým, že riešenia mnohých problémov si vyžadujú opakovať

¹² Tieto obdobia sú skutočne dlhé, najmä ak zväžeme, že každý z výsledných vzdelávacích obsahov je plánovaný asi na 16 vyučovacích hodín – vždy s prípadným presahom do nasledujúceho ročníka.

¹³ Označujeme takto prvé skúsenosti a predbežné predstavy, ktoré postupne povedú k porozumeniu konceptu (Merriam-Webster: *a rudimentary idea intermediate between an ordinary concept and a fully developed concept*).

¹⁴ A jej plánovanom pokračovaní v Scratchi a Pythone.

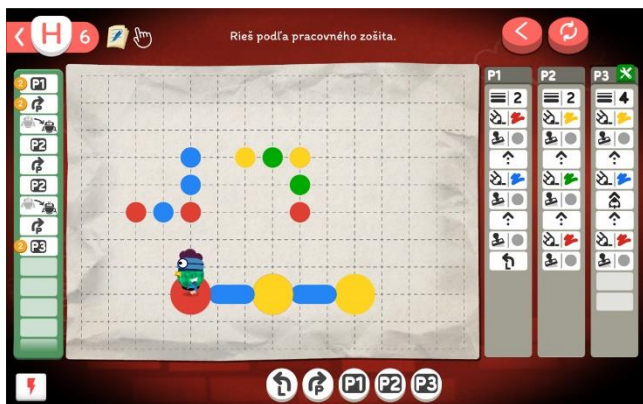
bezprostredne za sebou dva (a neskôr tri a štyri) rovnaké kroky niekoľkokrát – napr. *krok vpravo a polej*, pozri Obr. 1 (c).

- Žiaci objavia nový mechanizmus, ktorý im dovoľuje spojiť pri programovaní dva za sebou idúce príkazy – dve „kartičky“ – do jednej „dvojkartičky“. Ak takéto dvojkroky plánujú bezprostredne za sebou a spájajú ich do dvojkartičiek, v programe sa automaticky zhrňajú na kôpku, pozri Obr. 1 (d).
- Takéto záznamy krokov a programy na riešenie úloh budú žiaci vytvárať, čítať, analyzovať, vysvetľovať, rozvíjať a redukovať, čím si v zmysle Papertovho *program ako objekt, ktorým premýšľam a o ktorom premýšľam* rozvíjajú hlboké porozumenie viacerých kľúčových programovacích konštruktov, pozri Obr. 1(e).

Naším cieľom je, aby uvedené vlákno systematicky pripravovalo hlboké porozumenie konceptu opakovania a v Emilovi 4 naň potom nadväzujeme ďalšími krokmi¹⁵. Priradené, mnohé úlohy môžeme vnímať ako súčasť viacerých vlákien, pretože žiaci pripravujú na rôzne koncepty (napr. dvoj- a trojkartičky sú zároveň s opakováním aj prípravou na abstrakciu sekvencie príkazov pomocou vytvorenia zloženého príkazu alebo podprogramu). Niektoré vlákna sa teda v úlohách akoby križovali, rozdeľovali alebo i spájali dokopy, aby spoločne prispeli k porozumeniu niektorého konceptu a operácií s ním.

4 EMIL 4

Z hľadiska všeobecných vzdelávacích cieľov sme s Emilom 4 nadviazali na vzdelávací obsah a metódu Emila 3, v snahe pokračovať v rozvoji informatickej myslenia a poznania, a to opäť prostredníctvom programovania ako *nástroja na objavovanie, skúmanie, riešenie problémov a tvorbu*. Rozhodli sme sa využiť overenú paradigmu korytnačej geometrie¹⁶ s jej okamžitou vizuálnou spätnou väzbou v podobe čiarových kresieb, pečiatkovaných kompozícií a (vyfarbených) polygónov, pozri napr. Obr. 2. Po bohatých skúsenostiach so staršími implementáciami Loga sme sa však rozhodli pre vývinovo primeranú redukciju jazyka a konštruktov, o Emilovi 4 teda môžeme uvažovať ako o *pred logovskom programovaní*.



Obrázok 2: Ilustrácia prostredia Emil 4, úloha H6

Významné myšlienky informatiky, ktoré rozvíjame v Emilovi 3, sme detailnejšie prezentovali a analyzovali v [12] a [13]. V Emilovi 4

sme sa posunuli k budovaniu silnejšieho programovacieho jazyka a zamerali sa predovšetkým na tieto kľúčové javy v programovaní:

- Transformáciu z *absolútnej na relatívnu navigáciu* v zmysle korytnačej geometrie, čiže na **riadenie robota jeho otáčaním vľavo a vpravo**¹⁷ – spôsob orientácie, ktorý je už viac ako 50 rokov predmetom záujmu výskumníkov napr. v kontexte jazyka Logo, pozri [15] alebo [16], vždy považovaný za dôležitý ale problematický fenomén¹⁸.

Podľa našich skúseností je priame riadenie fyzického objektu, napr. programovateľnej hračky Bee-Bot alebo Blue-Bot svojím spôsobom relatívnej navigácie celkom prirodzené už pre žiakov nižších ročníkov primárneho vzdelávania, a to vďaka princípu **stotožnenia sa** (angl. *body syntonicity*) s reálnym zariadením, pozri napr. [17]. Žiaci sa v prvých etapách práce zároveň s otočením robota presúvajú po miestnosti tak, aby vždy stáli **za ním**, jednoducho, **otáčajú sa zároveň s ním**. Na druhej strane, väčšina programovacích prostredí na počítači (včítane Emila 3) pre tú istú vekovú skupinu dáva prednosť¹⁹ *absolútnej navigácii*, kedy sa krok vpred realizuje prostredníctvom jedného za štyroch príkazov *krok hore, vpravo, dolu a vľavo*. Domnievali sme sa však, že v Emilovi 4 je už vhodná chvíľa prejsť na riadenie a programovanie v systéme s relatívnou orientáciou. Pri vývoji a overovaní Emila 4 sme ale kládli mimoriadnu pozornosť (i) samotnej vizualizácii príkazov *vľavo a vpravo* (a viackrát ju aj modifikovali), a rovnako aj (ii) vzdelávaciemu obsahu a pedagogickým postupom na podporu rozvoja tohto konceptu. Na záver celého procesu vyslovujeme presvedčenie²⁰, že sa nám podarilo implementovať spomínanú transformáciu prekvapujúco hladko.

- *Príkazy na nastavenie a príkazy so vstupom* (na úrovni propedeutiky, teda prvých krokov). Kým v Emilovi 3 sú všetky príkazy vždy bez vstupu a ich vykonanie má na pracovnej ploche jednoznačnú a viditeľnú odozvu, v Emilovi 4 postupne zavádzame aj príkazy na nastavenie (konkrétne: farby pera a pečiatky a hrúbky čiary a veľkosti pečiatky), a tiež dva **príkazy s vizuálnym výberom hodnoty pre vstup**: *opečiatkuj tvar a vyfarbi farbou*. Všetky štyri spomínané príkazy však prešli v procese vývoja početnými modifikáciami, najviac z nich zrejme príkaz *vyfarbi farbou* (pozri záverečnú diskusiu).

- *Zložené príkazy*. Už v Emilovi 3 sme iniciovali vlákno s cieľom, aby žiaci urobili prvé kroky v smere **abstrakcie**²¹ v zmysle zastúpenia niekoľkých príkazov jediným symbolom prostredníctvom určitej vizuálnej skratky. V Emilovi 4 postúpime v tomto vlákne o niekoľko krokov ďalej tým, že od písmena **G** už majú žiaci k dispozícii jeden, dva alebo tri zložené príkazy *P1, P2 a P3*. Ďalej v tejto kapitole ich popíšeme detailnejšie.

Podobne ako v Emilovi 3, aj teraz celý vzdelávací obsah prezentujeme ako gradáciu približne 100 úloh rozdelených do 18 skupín, ktoré označujeme písmenami **A, po A, B, po B, ...**. Táto ústredná gradácia predstavuje sekvenciu malých krokov v objavovaní či rozvíjaní programovacích konštruktov. V gradácii úloh môžeme zároveň identifikovať viacero vlákien (ako sme vysvetlili skôr), ktoré buď nadväzujú na niektoré vlákno z predchádzajúceho ročníka,

¹⁵ Ktoré v nasledujúcom vzdelávacom obsahu v Scratchi povedú k explicitnému používaniu blokov *opakuj, opakuj stále a opakuj až kým*.

¹⁶ Pozri napr. ikonickú monografiu [15].

¹⁷ Zatiaľ vždy do 90 stupňov v zmysle vpravo v bok a vľavo v bok.

¹⁸ Vo vývinovej psychológii a v pedagogike známy ako pravo-ľavá orientácia.

¹⁹ Podľa nášho názoru oprávnené a správne.

²⁰ Ktoré treba, samozrejme, následne validovať ďalšími výskumnými postupmi.

²¹ Ktoré považujeme za súvisiace s konceptom opakovania, najmä vtedy, ak majú žiaci možnosť opakujúcu sa skupinu príkazov dáko označiť, napr. používaním symbolu (v treťom svete Emila 3 symbolom pamäťovej karty – zelenej, červenej alebo fialovej).

alebo prispievajú k novým objavom, alebo tvoria propedeutiku konštruktov, ktoré budú žiaci objavovať neskôr.

Každá úloha gradácie poskytuje žiakovi iba obmedzený výber z úplnej množiny príkazov. Tie sú v úlohe prístupné v tvare okrúhlych tlačidiel pod Emilovou plochou, pozri Obr. 2. V bežnom režime práce²² sa každé použitie (kliknutie alebo dotyk) tlačidla okamžite vykoná a zapíše do nasledujúceho riadka panela so záznamom vykonaných príkazov. Úplnú množinu príkazov jazyka Emil 4 má žiak k dispozícii iba v úlohe X14 rozširujúcej sady úloh X.

Tento princíp meniacej sa sady dostupných nástrojov vyplýva z didaktickej podstaty našej metódy: žiak v každej chvíli disponuje (viac-menej iba) tými programovacími prostriedkami, ktoré mu pomôžu pri nasledujúcom objave alebo pri automatizácii predchádzajúcich poznatkov či vlastnej tvorbe²³. Tento prístup môžeme názorne ilustrovať tak, že si všimneme aktuálne dostupné výbery príkazov v kľúčových úlohách a zoradíme ich pod seba. Výsledok vidíme v Tabuľke 2, ale podrobnejšie ju komentujeme až ďalej. Už teraz si však môžeme v nej všimnúť *pulzujúci počet príkazov*, ktorý v každom významnom bode, ktorý si vyžaduje objav žiakmi, klesne úmyselne na minimum.

4.1 Proces vývoja

Pri vývoji Emila 4 sme postupovali rovnakou stratégiou výskumu vývojom ako v Emilovi 3, starostlivejšie sme ho však dokumentovali a na záver vývoja a overovania analyzovali. Bližší opis procesu a výsledky jeho analýzy nie sú súčasťou tohto príspevku, tu len spomenieme, že sme v celom procese identifikovali päť rôznych, aj keď prepletených aspektov, a to vývoj: (i) programovacieho prostredia, (ii) programovacieho jazyka, (iii) vzdelávacieho obsahu²⁴, (iv) pracovného zošita, a napokon (v) stratégie a obsahu vzdelávania pedagógov. Na „ortogonálnej osi“ sme identifikovali päť rôznych zdrojov dát, ktoré sme priebežne zbierali a využívali pri rozhodovaní o nasledujúcej iterácii vývoja: (a) pozorovania žiakov, (b) rozhovory s učiteľmi, (c) pilotné pracovné listy, (d) postrehy a reakcie z pilotných vzdelávaní učiteľov a (e) vypracované zošity Emila 3 žiakov z rôznych škôl. V procese vývoja mali výsledky analýzy dát z týchto zdrojov určujúci vplyv na ďalší vývoj v každom zo spomínaných aspektov. Napr. po analýze vyplnených pilotných pracovných listov²⁵ od žiakov sme často usúdili, že musíme pridať, vynechať alebo prepracovať rôzne detaily v prostredí, jazyku i pripravovanom pracovnom zošite.

4.2 Programovací jazyk

Príkazy jazyka sú žiakovi prístupné ako tlačidlá pod pracovnou plochou, kliknutím myšou alebo dotykom prstu sa príkaz okamžite vykoná a na zelenom paneli vľavo od plochy o tom vznikne záznam – buď do nasledujúceho voľného okienka (riadka) alebo zvýšením počtu opakovaní v naposledy zapísanom okienku, ak ide o rovnaký príkaz. Ak Emil príkaz vykonať nemôže, napr. narazí na okraj mriežky, dostáva sa do chybového stavu. Ten žiaci odstránia tak, že zrušia načerveno svietiaci krok na paneli so záznamom. Emil sa vtedy vráti do aktuálneho stavu pred týmto krokom. Takýto *krok späť* funguje kedykoľvek a dá sa aplikovať aj opakovane na zrušenie niekoľkých posledných príkazov.

Príkazy jazyka delíme na **základné** (je ich deväť) a **zložené** (sú tri),

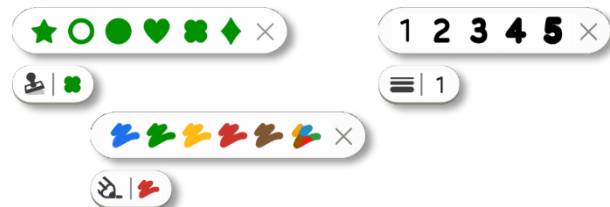
pozri Obr. 3. Podľa ďalších syntaktických a sémantických kritérií ich môžeme detailnejšie charakterizovať v takýchto skupinách:

- Pohyb vpred v myslenej štvorcovej mriežke bodov²⁶, vždy horizontálne alebo vertikálne, v smere, ktorým je Emil práve otočený, vždy do susedného myslenej mriežkového bodu – ak je to možné. Rozhodli sme sa nezaviesť príkaz(y) na zapínanie a vypínanie pera, a to z dôvodu dĺžky programov. Preto majú žiaci k dispozícii dva nezávislé príkazy *vpred* (bez kreslenia) a *kresli*. Príkazy nemajú vstup. Nakreslená čiara, presnejšie úsečka do susedného mriežkového bodu je aktuálne nastavenej farby a hrúbky Emilovho pera – v úvodných úlohách sú tieto nastavenia fixne určené autormi obsahu (v rôznych úlohách rôzne).



Obrázok 3: Úplný programovací jazyk v prostredí Emil 4

- Otáčanie *vľavo* a *vpravo* o 90 stupňov. Ani tieto príkazy nemajú vstup.
- Pečiatkovanie jedného zo šiestich tvarov do mriežkového bodu, v ktorom sa Emil práve nachádza, aktuálnou farbou a veľkosťou. V prvých skupinách úloh je farba aj veľkosť pečiatky fixne nastavené autormi, neskôr každé kliknutie na príkaz najprv otvorí ponuku so všetkými tvarmi (pozri Obr. 4) a potom, keď si žiak jeden z nich kliknutím zvolí, sa ponuka zavrie a príkaz vykoná. Aktuálne nastavenú farbu a naposledy použitý (alebo autormi prednastavený) tvar vidíme aj v pravej časti tlačidla s príkazom *opečiatkuj*. Je to prvý príkaz, kedy žiaci – aj keď bez zdôrazňovania zo strany pedagóga – pracujú s príkazom so vstupom.
- Príkazy na nastavenie farby a hrúbky/veľkosti. Ponuka farieb obsahuje päť fixných farieb, od úlohy G6 a po G1 už obsahuje aj šiestu, ktorá predstavuje *náhodnú farbu*, pozri Obr. 4. Nastavenie farby platí až do momentu ďalšieho nastavenia farby²⁷, aktuálnu farbu vidíme v pravej časti tlačidla²⁸. Záznam o voľbe farby sa zobrazí na zelenom paneli vľavo od Emilovej plochy. Ponuka hrúbok/veľkostí ponúka fixne päť možností, toto nastavenie určuje hrúbku čiary a veľkosť pečiatky. V skupinách úloh A, B a C hrúbku čiary a veľkosť pečiatky nastavili vopred autori.



Obrázok 4: Príkaz *opečiatkuj* so vstupom a ponukou možností. Príkazy na nastavenie farby a hrúbky/veľkosti

- Príkaz na vyplnenie ohraničenej oblasti. Ak majú žiaci v danej úlohe tento príkaz k dispozícii, zobrazuje sa v jednom z dvoch stavov: ako prítomný v ponuke príkazov ale momentálne nepoužiteľný, alebo ako aktívny a pripravený na použitie, pozri Obr. 5. Správanie a implementácia tohto príkazu prešla v procese vývoja Emila 4 zrejme najväčšími a najpočetnejšími zmenami zo všetkých príkazov. Finálny stav tohto nástroja implementuje zásadu,

²² Teda v režime priameho riadenia.

²³ Tento prístup sa vo výskumnej literatúre niekedy označuje ako *immersion learning* alebo *immersion in inquiry-based learning*, pozri napr. [18].

²⁴ Ktorý v sebe odráža aj všetky pedagogické a vývojové princípy.

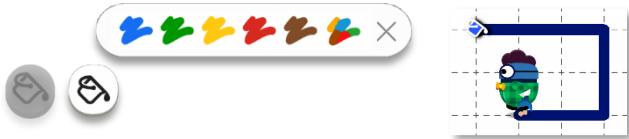
²⁵ Bola to pilotná podoba výsledného pracovného zošita.

²⁶ Niekedy zobrazenej v pracovnej ploche, inokedy nie.

²⁷ Alebo prechodu na inú úlohu.

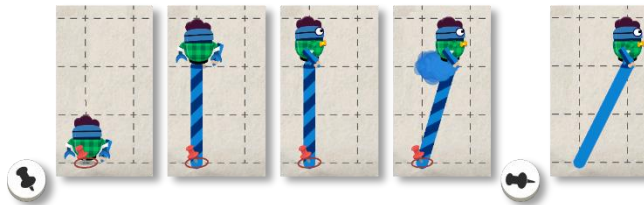
²⁸ A zároveň aj na pastelke, ktorú Emil drží v pravej ruke.

že oblasť²⁹ sa dá vyplniť iba vo chvíli (nie skôr a nie neskôr), kedy dokončíme kreslenie jej obrysu postupnosťou navzájom úsečiek rovnakej farby a hrúbky a s Emilom sme práve na mieste, kde sme obrys začali kresliť – iba vtedy sa aktivuje tlačidlo *vyfarbi*, jeho použitím sa zobrazí identická ponuka farieb ako pri nastavovaní farby. Keď si žiak zvolí farbu, oblasť sa vyfarbí a tlačidlo sa deaktivuje.



Obrázok 5: Vizuálne pomôcky na podporu vyfarbovania ohraničenej oblasti

Aby sme žiakom sprostredkovali túto operáciu čo najprimeranejším spôsobom, na základe početných pozorovaní sme napokon pridali ešte jednu pomôcku: v bode pracovnej plochy, kde Emil začne kresliť čiaru (teda možno obrys budúcej oblasti) sa objaví animovaný symbol vedierka (pozri Obr. 5 vpravo) ako pripomienka: sem sa musíme vrátiť kreslením čiary, aby sme mohli potom celú oblasť vyfarbiť.



Obrázok 6. Kreslenie šikmých čiar

- Príkaz *špendlík* na kreslenie šikmých čiar (napriek tomu, že Emil sa pohybuje iba horizontálne a vertikálne). Ak majú žiaci tento príkaz k dispozícii, aj ten sa zobrazuje v jednom z dvoch stavov: v stave *zapichni ma*, ak chceš začať kresliť šikmú čiaru, a potom v stave *keď si so svojou šikmou čiarou hotový, vyťahni ma* (pozri Obr. 6). Šikmou čiarou môže Emil spojiť dva ľubovoľné body mriežky: na jej „začiatku“ zapichnete špendlík, potom Emila riadime v mriežke bežným spôsobom príkazmi *kresli*, *vľavo* a *vpravo* – Emil sa pohybuje ako doposiaľ, ale čiara, ktorú za sebou ťahá, sa správa ako *gumka*, ktorá stále spája zapichnutý špendlík a jeho aktuálnu polohu. Gumka sa zmení na bežnú nakreslenú čiaru v ploche až vo chvíli, kedy opäť použijeme príkaz

Tabuľka 1: Príklad otázok do záverečných diskusií so žiakmi v druhom svete Emil 3 a v Emilovi 4

	(1)	(2)	(3)	(4)
A				
B				
C				

V diskusiách, ktoré sú rozvíjajúce, hodnotiace alebo záverečné, sa pýtame žiakov napr.:

- Čo je prvý krok, ktorý Emil urobí, keď sa riadi programom A (keď vykonáva takýto program)?
- Čo vykoná ako druhé? Ako piate? Ako deviate? Ako desiate? A jedenáste?
- Čo urobí Emil ako štvrté, keď vykonáva program B? Ako piate? A ako ôsme?
- ... atď.

Štyri situácie (1) až (4) predstavujú žiacke riešenia niektorých úloh záverečnej tretiny vzdelávacieho obsahu a ilustrujú najvyššie stupne kognitívnej obťažnosti pri práci so zloženými príkazmi v Emilovi 4 – medzi (1), (2), (3) a (4) vidíme jasnú gradáciu náročnosti, ktorú v sebe zahŕňajú. Učiteľom odporúčame, aby sa žiakov pýtali napr.:

- [v situácii typu (1)] Keď Emilovi zadáme príkaz *P1*, ktorý základný príkaz urobí ako prvý (čo urobí prvé)? (nastaví hnedú farbu) Čo urobí ako druhé, tretie, piate... siedme?
- [v situácii typu (2)] Čo Emil urobí ako prvé, keď mu zadáme *P1* raz? Čo ako druhé? Čo urobí ako štvrté, keď mu *P1* zadáme dvakrát? Ak zadáme *P1* za sebou štyrikrát, čo urobí ako siedme a ôsme?
- [v situácii typu (3)] Keď Emilovi zadáme *P2*, ktorý základný príkaz vykoná ako prvý? Čo urobí ako tretie? (príkaz *kresli*) Ako štvrté? A šieste? ...
- [v situácii typu (4), náročnej] Keď Emilovi zadáme *P3*, ktorý základný príkaz vykoná ako prvý? (nastaví hrúbku na 5) Ako tretí? Ako pätnásty? Dvadsiaty druhý? (vpred)

²⁹ Ľubovoľný mnohoúhelník, pravidelný alebo nepravidelný, konvexný alebo konkávný, vymedzený úsečkami, ktoré sme nakreslili postupným spájaním dvojíc mriežkových bodov (vodorovných, zvislých alebo šikmých – pozri ďalej).

špendlík – akoby sme z plochy „vytiahli“ špendlík, ktorý doposiaľ pridržal začiatok gumičky.

- Zložené príkazy *P1*, *P2* a *P3*. Z didaktického hľadiska ide o ďalší krok smerom ku konceptu podprogramu či funkcie. Žiaci môžu v Emilovi 4 zložené príkazy používať, neskôr aj opravovať a upravovať, a napokon aj sami vytvárať. Celkovo tú k dispozícii iba tri a definícia každého z nich má obmedzený počet okienok (riadkov), maximálny počet je 13. Zložený príkaz môže v svojom tele používať ktorýkoľvek základný príkaz, a tiež iný zložený príkaz – ale s obmedzením, že toto volanie nevedie k priamej alebo nepriamej rekurzii. Na Obr. 2 na začiatku tejto kapitoly vidíme, ako v ponuke aktívnych príkazov na tlačidlách pod Emilovou plochou vizuálne sprostredkujeme žiakom zložené príkazy (presnejšie ich volanie), ako na paneli so záznamom vykonaných príkazov zapisujeme volanie zloženého príkazu, a tiež ako zobrazíme definície zložených príkazov na paneloch vpravo.

Príkaz na volanie (vykonanie) zloženého príkazu v našej metóde z didaktických dôvodov nepovažujeme za základný príkaz. To nám dovoľuje vekovo primeraným spôsobom budovať systematické porozumenie konštruktu funkcie či zloženého (rozširujúceho) príkazu, teda sprostredkovať tento koncept žiakom ako gradáciu konkrétnych činností, bez akejkoľvek potreby výkladu, iba riešením problémov v dvojici, a následnými dômyselne riadenými diskusiami celej skupiny. Podobne ako sme v Emilovi 3 nehovorili o dvojkartičke³⁰ ako o bežnej kartičke či základnom príkaze, aj teraz rozlišujeme základné príkazy, ako napr. *kresli*, *vľavo* atď. a zložené príkazy *P1*, *P2* a *P3* – a budeme tým mať na mysli aj ich volanie. Prítom vlastná definícia každého zloženého príkazu je na obrazovke alebo v pracovnom zošite stále viditeľná. Pojmy *volanie* a *definícia zloženého príkazu* so žiakmi v tomto ročníku a ich učiteľmi ešte nepoužívame. Namiesto *volať* hovoríme *použiť*. Keď chceme upriamiť ich pozornosť na *definíciu*, pýtame sa *Čo presne robí P1? Aké kroky, teda aké základné príkazy?* Tento prístup nám dovoľuje postupne budovať ich kvalitné a trvácne porozumenie konštruktu zloženého príkazu už v tomto veku, a to aj vďaka dôležitým záverečným otázkam, ktoré ilustruje Tabuľka 1 v ľavej časti k Emilovi 3³¹ a v pravej k Emilovi 4.

Predchádzajúci text spolu s Tabuľkou 1 je zrejme dobrou ilustráciou skutočností, že v školskej informatike pri tvorbe, overovaní a následnej didaktickej analýze pedagogickej intervencie³² je nedostatočné prezentovať iba príkazy a konštrukcie programovacieho jazyka, prípadne sumarizovať koncepty, ktorých porozumenie tým budujeme. Takáto analýza by bola neúplná bez uvažovania o vzdelávacom obsahu, o gradácii *operácií*, ktoré s nimi žiaci vykonávajú, a o metodike, ako sa s nimi postupne oboznamujú a vytvárajú si tak hlboké a systematické porozumenie zodpovedajúcich *programovacích konštruktov*. Napr. anglické národné kurikulum pre druhý cyklus (ročníky 3 až 6) stanovuje, že žiak sa má naučiť... *use sequence, selection, and repetition in programs; work with variables and various forms of input and output*. Pre tretí cyklus (ročníky 7 až 9) potom vyžaduje, aby sa žiak naučil... *use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structu-*

res [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions... Slová, ktoré sa tu používajú (ako *use, work with, make appropriate use, ...*), však nešpecifikujú s dostatočnou detailnosťou, čo presne má žiak 3. až 6. ročníka vedieť robiť s postupnosťou príkazov, podmieneným vykonaním či opakovaním (ako sme ho prezentovali v časti 3.3).

Pozrime sa preto najprv na gradáciu toho, ako sa vo vzdelávacom obsahu mení³³ ponuka príkazov, ktoré majú žiaci k dispozícii v jednotlivých úlohách. Po ukončení procesu vývoja Emila 4 a jeho priebežného overovania sme v rámci spomínanej záverečnej analýzy zoradili pod seba tieto skupiny a redukovaním opakujúcich sa alebo veľmi podobných riadkov tak vznikla vizualizácia, ktorú prezentujeme v Tabuľke 2. Domnievame sa, že názorne ilustruje jeden z našich didaktických princípov: každý nový – a zrejme kognitívne náročnejší – jav, ktorý idú žiaci skúmať a poznávať, sa snažíme prezentovať tak, aby sa naň žiaci mohli plne sústrediť, teda v čo najmenšej a v tej chvíli podstatnej ponuke príkazov. Optické „vlny“, ktoré sa takto zviditeľnili, identifikujú práve tieto momenty. Vzniklo tak deväť dobre rozpoznatelných úsekov gradácie vzdelávacieho obsahu (1) až (9), oddelených myslenskými medznicami v očakávanom poznávacom procese žiakov. Kým v Tabuľke 2 textom stručne charakterizujeme, čomu sa v danom úseku venujeme, pomenujme teraz samotné medzníky a čo predstavujú:

- 1 intuitívny vstup do základného pohybu Emila, pečiatkovanie tu používame preto, aby žiakom vznikala na Emilovej ploche určitá stopa či zmena,
- 2 v úlohe **A3** sme vytvorili určité napätie u žiakov – cítia a diskutujú tu o potrebe otáčať Emila. Už nasledujúca úloha im dovoľí otáčať ho vpravo, ďalšie úlohy už aj vľavo,
- 3 pribúda druhý typ pohybu vpred, a to s kreslením čiary príkazom *kresli* a zvýraznenie tohto javu tým, že v úlohe **B4** majú k dispozícii (a aj potrebujú) iba príkazy *vpred* a *kresli*,
- 4 žiaci objavujú, ako sa dá Emil (nie príkazmi jazyka) ťahať myšou alebo prstom (tzv. *meta ťahanie*), a tiež, ako môžu vyfarbiť zvolenou farbou oblasť, ktorú práve ohraničili čiarou,
- 5 kreslenie šikmých čiar v rámci mriežky bodov pomocou príkazu špendlík („zapichnutý“ na začiatku čiary a „vytiahnutý“ na jej konci),
- 6 – 9 odteraz až do konca vzdelávacieho obsahu sú centrálnym konceptom vzdelávacieho procesu zložené príkazy. V gradácii sa od tohto medzníka cez všetky ďalšie úlohy vinie vlákno zamerané na jednotlivé operácie so zloženými príkazmi, ktoré sme identifikovali a naprojektovali ako vývinovo primerané pre žiakov pri postupnom oboznamovaní sa s týmto kľúčovým mechanizmom abstrakcie v programovaní.

Ako vidíme, vlastný programovací jazyk Emila 4 je jednoduchý a značne obmedzený. Tým viac – špeciálne v kontexte elementárneho školského programovania – sa nám javí ako **málo informatívne, ak by sme nešpecifikovali spolu s programovacím jazykom aj náš didaktický zámer**. Ten v našom prípade vnímame ako už spomínanú sústavu *vlákieň úloh úplnej gradácie* zameraných na budovanie systematického a hlbokého porozumenia jednotlivých konštruktov. Pre obmedzený rozsah príspevku tu však nemôžeme

zhrnúť na jednu „kôpku“. To prirodzene platí aj pre tri dvojkartičky atď.

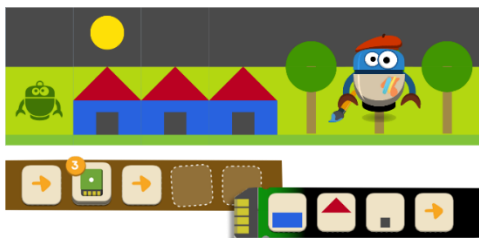
³² Akú tu so zameraním na programovací jazyk Emila 4 práve robíme.

³³ Úmyselne sa tu vyhýbame slovu rozširuje, ako ozrejmuje Tabuľka 2.

³⁰ Videli sme ich na Obr. 1.

³¹ Na záver skupiny úloh **G** v druhom svete, teda po viacnásobných konkrétnych skúsenostiach, pri ktorých žiaci objavili tento koncept a mechanizmus a už viackrát využili skutočnosť, že ak sa v programe dajú spájať vytvoriť dve rovnaké dvojkartičky (z dvoch a dvoch kartičiek) idúce hneď za sebou, automaticky sa

špecifikovať všetky vlákna Emila 4, preto sa obmedzíme na jedno, a to vlákno zamerané práve na zložené príkazy – aj preto, že sa vnuje kognitívne najnáročnejšiemu konštruktú intervencie Emil 4.

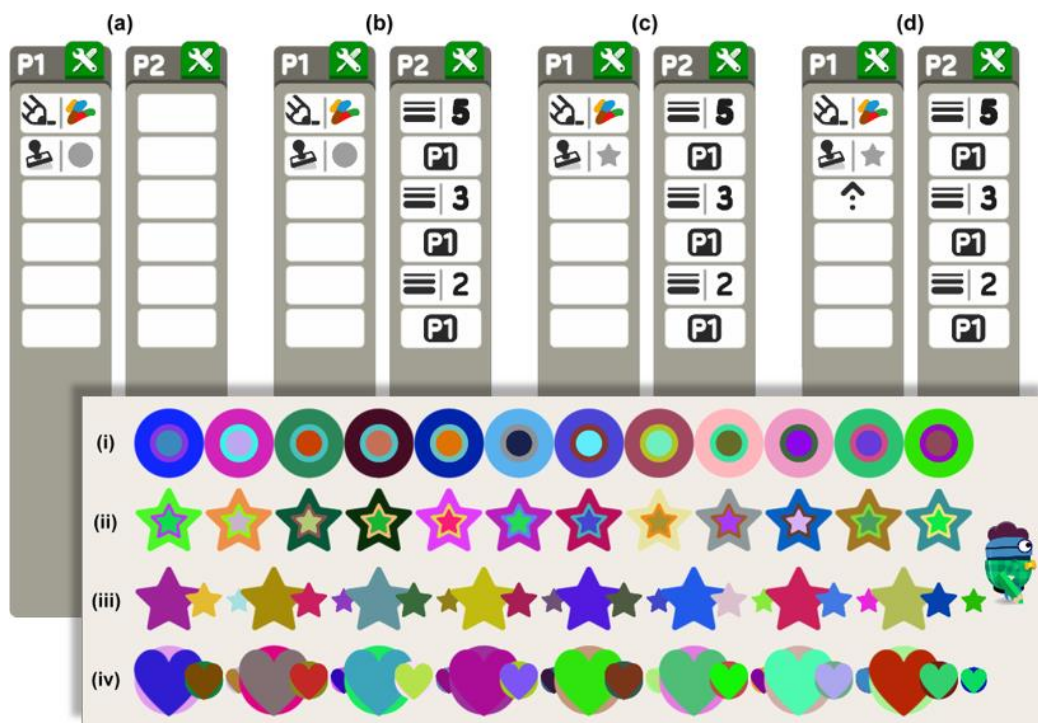


Obrázok 7. Emil 3 v tretom svete so zelenou pamäťovou kartou

Toto vlákno sa v skutočnosti svojou propedeutickou časťou začína už v Emilovi 3, kde v tretom svete žiaci používajú, analyzujú a vytvárajú tri „pamäťové karty“ (s dvoma, tromi, štyrmi či piatimi Emilovými príkazmi), pozri ilustráciu na Obr. 7. Každá karta sa potom dá pri programovaní úloh použiť ako volanie (zloženého) príkazu. Ak túto kartu žiak použije viackrát za sebou, v programe sa kartičky s jej obrázkovým symbolom automaticky zhŕňajú na kôpku (Tabuľka 1 ilustruje obdobnú vizualizáciu opakujúcich sa príkazov v Emilovom druhom svete). Už tu teda začíname postupne budovať prekoncept zloženého príkazu. Následne v Emilovi 4 pokračujeme v rámci tohto vlákna úloh ďalšími krokmi, ktoré zvyšujú mieru abstrakcie v plánovaní riešenia, pretože umožňujú

žiakom vytvárať a využívať zložené príkazy, ako sme prezentovali v úsekoch 6 – 9 v texte vyššie. V úlohách týchto úsekov žiaci postupne:

- pri priamom riadení Emila používajú zložený príkaz *P1*, vopred vytvorený autormi obsahu³⁴. K dispozícii majú iba tento príkaz a žiadny iný. Definícia *P1* obsahuje päť základných príkazov bez nastavení a vstupov,
- ako v predchádzajúcom kroku, ale k dispozícii už majú aj *P1*, aj niekoľko základných príkazov,
- pri priamom riadení Emila používajú vopred pripravené zložené príkazy *P1* a *P2*, vopred vytvorené autormi obsahu. Tieto zložené príkazy vo svojich definíciách už obsahujú aj volania základných príkazov so vstupom a sú navzájom nezávislé – ani jeden nevolá druhý z nich,
- v rozvíjaní svojho porozumenia konštruktú zloženého príkazu žiaci používajú volanie (vhodne zvoleného) príkazu *P1* raz, dvakrát, trikrát a štyrikrát a priebežne si kreslia obrázok, ktorý takto vzniká. V tejto skupine úloh križuje práve prezentované vlákno zároveň aj vlákno, ktoré už od Emila 3 rozvíja konštruktú opakovania,
- pri priamom riadení Emila používajú vopred pripravené zložené príkazy *P1*, *P2* a *P3* spolu s ďalšími základnými príkazmi,
- upravujú posledný riadok definície pripraveného *P1* (konkrétne, vyfarbenie oblasti inou farbou), aby mohli vyriešiť zadanie úlohy),



Obrázok 8. Niektoré etapy v riešení úlohy J1

³⁴ V tomto príspevku nevenujeme našim pedagogickým a vývojovým princípom, ktoré aplikujeme pri vývoji obsahu, metodické príručky a vzdelávania pedagógov. V tomto prípade však chceme zdôrazniť, že žiadny z javov či postupov, ktoré chceme u žiakov rozvíjať, učiteľ nevysvetľuje a „neprezrádza“. Úlohy sú na-

vrhnuté a gradované tak, aby všetky pribúdajúce možnosti (novinky) žiaci objavili pri riešení úloh v dvojiciach a pri priebežných diskusiách s celou skupinou či triedou. Výsledkom vzdelávania je práve pripravenosť učiteľa riadiť tieto diskusie a pýtať sa vhodných otázok.

- upravujú iný než posledný krok v definícii pripraveného *P2 analógiu s podobným* pripraveným zloženým príkazom *P1* tak, aby mohli vyriešiť zadanie úlohy,
- programujú svoj vlastný príkaz *P2 analógiu s podobným*, už vopred pripraveným *P1* tak, aby daný program, ktorý volá aj *P1* aj *P2* vytvoril dané obrázky,
- programujú svoj vlastný príkaz *P2* tak, že v jeho definícii opakovane volajú *P1*, neskôr aj spolu s ďalšími základnými príkazmi,
- programujú svoj vlastný príkaz *P2* tak, že opakovane volajú *P1*, ale pred každým volaním nastaví farbu alebo hrúbku/veľkosť. Tieto nastavenia vždy zmenia, čo presne vznikne použitím *P1*. Spománanú etapu vlákna ilustruje Obr. 8 na úlohe **J1**: (a) vyjadruje úvodný stav definícií *P1* a *P2*. Malý zelený symbol náradia v hlavičke definície signalizuje, že do nej žiaci vedia „vojsť“ a opraviť ju alebo ju celú vytvoriť. Keď si žiaci vytvoria *P2*, budú ho v priamom režime používať striedavo s príkazom *vpred*, čím vznikne retiazka (i); stav (b) zobrazuje riešenie prvej etapy úlohy. V druhej etape žiakom navrhujeme, aby *P2* už nemenili, ale modifikovali *P1* tak, aby rovnakým programom (*P2 vpred P2 vpred...*) vznikli retiazky ako (ii). Ďalšia etapa navrhuje preskúmať retiazku (iii) a doplniť *P1* tak, aby programom *P2 P2 P2...* vznikol napr. takýto výsledok, riešenia vidíme na (d). Retiazka (iv) zodpovedá ďalšej etape, ktorá vyžaduje ďalšiu úpravu *P1*,
- v poslednom kroku gradácie Emila 4 na vlákne zložených príkazov budú žiaci programovať svoje vlastné *P1* podľa obrázkového alebo textového zadania, potom *P2*, ktoré používa *P1*, a tiež *P3*, ktoré používa *P2* – analyzujú, plánujú a riešia takto rôzne komplexnejšie obrázkové problémy.

5 DISKUSIA

5.1 Vzdelávacie ciele, vzdelávací obsah a programovací jazyk

V predchádzajúcej kapitole sme charakterizovali programovací jazyk, ktorý postupne žiaci odkrývajú v gradácii úloh vzdelávacieho obsahu Emila 4. Argumentovali sme, že v takejto situácii – špeciálne vo vzdelávacom prostredí, a možno ešte viac, ak ide o primárne vzdelávanie – prezentácia vlastného programovacieho jazyka ponúka len čiastočný pohľad na podstatu vyvinutej intervencie, teda na jej vzdelávacie ciele, vzdelávací obsah a *významné myšlienky a konštrukty programovania*, ktoré chceme rozvíjať. Preto sme stručne predstavili aj náš prístup k tomu, ako v gradácii úloh budujeme tzv. vlákna zamerané na postupné gradovanie príležitostí na objavovanie a hlboké porozumenie kľúčovým konštruktom programovania, teda konceptom a operáciám, ktoré s nimi vykonávame. Naším základným cieľom v projekte je metódami výskumu vývojom (i) identifikovať tie koncepty a tie operácie s nimi, ktoré sa potvrdia ako vývinovo primerané pre úvod do programovania v primárnom vzdelávaní, a tiež (ii) iteratívne navrhnuť, implementovať a overiť primeranú sekvenciu aktivít, ktorá dovolí žiakom konštruovať si kvalitné a trvalé porozumenie zvolených konceptov a operácií v krokoch s vhodným rastom kognitívnej náročnosti. Tento prístup sme detailnejšie prezentovali na vlákne smerujúcom ku konštruktu podprogramu.

³⁵ Ktorý v sebe zahŕňa aj všetky pedagogické a vývojové princípy.

³⁶ Toto presvedčenie vyslovujeme aj vzhľadom na počty škôl, ktoré

V časti 4.1 sme zmienili, že v procese vývoja výslednej intervencie v našom projekte môžeme sledovať päť rôznych, ale prepletených aspektov: vývoj programovacieho prostredia, programovacieho jazyka, vzdelávacieho obsahu³⁵, pracovného zošita, a napokon stratégie a obsahu vzdelávania učiteliek a učiteľov: v procese vývoja každý z týchto aspektov vplýval na všetky zvyšné. Vo svetle tejto skúsenosti sme si vedomí toho, že náš príspevok prezentuje výsledok vývoja Emila 4 iba čiastočne. Úmyselne sme sa však zamerali prevažne na jeden z aspektov a pokúsili sa ho prezentovať detailne a komplexne.

5.2 Výsledná intervencia ako výskumný inštrument

Za ďalšiu dôležitú skutočnosť, ktorú tu chceme zmieniť, je naše presvedčenie, že obe doposiaľ vytvorené pedagogické intervencie, teda Emila 3 a Emila 4, nepovažujeme iba za odpoveď na naše výskumné otázky v procese vývoja, ale zároveň aj *za nástroj, ktorý je k dispozícii na ďalšie výskumné projekty*. Každý z nich podľa nášho názoru ponúka širokú paletu príležitostí na realizáciu následných kvalitatívnych a kvantitatívnych³⁶ výskumných projektov, zameraných napr. na:

- transformáciu učiteľov (z rôznych pohľadov), ich postojov pred a po úvodnom vzdelávaní a počas nasledujúceho obdobia, kedy začnú nový vzdelávací obsah používať vo svojej triede, zmeny v ich nazeraní na informatiku a jej kľúčovú súčasť – programovanie, ako sa vyvíja ich vlastné porozumenie jednotlivých konceptov a operácií, spôsoby, ako realizujú jednotlivé hodiny a ako prijímajú zatiaľ u nás nie celkom bežné metodické postupy, ktoré im navrhujeme a pod.,
- transformáciu žiakov (opäť z mnohých rôznych pohľadov), ich poznávacieho procesu v jednotlivých detailoch, napr. proces vznikajúceho porozumenia jednotlivých skúmaných konceptov a jednotlivých operácií, ktoré s nimi vykonávajú, trvácnosť a hĺbku vznikajúceho porozumenia, transformáciu ich postojov a postupov pri riešení úloh, interakcie v rámci dvojice počas vyučovacej hodiny, ale aj v rámci celoskupinových diskusií, spôsoby ich vlastných zápisov postupov a riešení do pracovného zošita a pod.

Nasledujúce roky praxe a ďalšie výskumy ukážu, či a ako prispievajú naše intervencie k zmene vo vyučovaní informatiky na prvom stupni. V tomto období pokračujeme vo výskume vývojom ďalšieho vzdelávacieho obsahu pre 2. stupeň ZŠ, ktorý na rovnakých pedagogických a didaktických princípoch nadviaže na Emila 4 v 5. ročníku. Za programovací jazyk sme si celkom prirodzene zvolili Scratch a domnievame sa, že progres žiakov, ktorí už predtým pracovali s oboma intervenciami Emila, bude strmsší, než sme zvyknutí pri piatakoch ako začiatočníkoch v programovaní.

6 POĎAKOVANIE

Ďakujeme agentúre VEGA, ktorá prostredníctvom nášho projektu 1/0602/20 prispela k tomu, že výsledky vývoja a ich analýzu môžeme publikovať na konferencii Didinfo 2021. Jeden z autorov tiež ďakuje svojim kolegom C. Hoyle, R. Noss, L. Benton a P. Saunders z projektu ScratchMaths (2014 – 2017), pretože mnoho vývojových a didaktických princípov v našej súčasnej práci má svoje východiská práve v tejto spolupráci.

už začali používať Emila 3 a 4 na vyučovaní informatiky, zvyčajne v 3. a 4. ročníku, niekedy čiastočne aj v ročníku 2. a/alebo 5., na Slovensku, v Čechách a ďalších krajinách.

BIBLIOGRAFICKÉ ODKAZY

- [1] SPIELER, B. et al. The learning value of game design activities: association between computational thinking and cognitive skills. *Proc of WiPSCE '20*, October 2020, doi.org/10.1145/3421590.3421607.
- [2] GRIZIOTI, M., KYNIGOS, Ch. Programming approaches to computational thinking: Integrating turtle geometry, dynamic manipulation and 3D space. In: *Proc of the Constructionism*, 2018, Vilnius, doi.org/10.15388/infedu.2018.17.
- [3] MEERBAUM-SALANT, O., ARMONI, M., BEN-ARI, M. Learning computer science concepts with Scratch. *Computer Science Education*, Vol 23 (3), 2013, pp. 239-264.
- [4] OUAHBI, I. et al. Learning Basic Programming Concepts By Creating Games With Scratch Programming Environment. *Procedia – Social and Behavioral Sciences* 191, 2015, pp. 1479 – 1482.
- [5] SENTENCE, S.: Moving to mainstream: developing computing for all. *Proc of WiPSCE '19*, doi.org/10.1145/3361721.3362117.
- [6] BLACKWELL, A.F. What is Programming? In *14th Workshop of the Psychology of Programming Interest Group*, 2002, pp. 204-218.
- [7] BRENNAN, K., RESNICK, M. New frameworks for studying and assessing the development of computational thinking. *Proc. of the 2012 Annual Meeting of the American Edu Research Association*, Vancouver, 2012.
- [8] KALAS, I., BENTON, L. Defining Procedures in Early Computing Education. In: Tatnall, A., Webb, M. (eds.) *Tomorrow's Learning: Involving Everyone. Learning with and about Technologies and Computing*. WCCE 2017. IFIP Advances in Information and Communication Technology, vol 515. Springer, Cham, 2017, pp. 567-578.
- [9] PERLMAN, R. *Using Computer Technology To Provide A Creative Learning Environment For Preschool Children*. AI Memo 360, MIT 1976, 32 p.
- [10] MORGADO, L., CRUZ, M., KAHN, K. *Radia Perlman – A pioneer of young children computer programming*. Current Developments in Technology-Assisted Education. Formatex 2006, pp.1903-1908.
- [11] BENTON, L., HOYLES, C., KALAS, I., NOSS, R. Bridging Primary Programming and Mathematics: Some Findings of Design Research in England. *Digit. Exp. in Math. Educ* (2017) Vol 3: 2017, Springer, pp. 115-138.
- [12] KALAS, I. Programming in lower primary years: design principles and powerful ideas. *Proc of Constructionism. Computational Thinking and Educational Innovation*, Vilnius 2018, pp.7 1-80.
- [13] KALAS, I., BLAHO, A., MORAVCIK, M. Exploring Control in Early Computing Education. In: Sergei N. Pozdniakov and Valentina Dagienė (eds.) *Informatics in Schools. Fundamentals of Computer Science and Software Engineering*. ISSEP 2018. LNCS, vol 11169, Springer 2018, pp. 3-16.
- [14] PLOMP, T., NIEVEEN, N. (Eds.) *Educational Design Research: An Introduction*. SLO Enschede 2013, 203 p.
- [15] ABELSON, H., diSESSA, A.A. *Turtle Geometry. The Computer as a Medium for Exploring Mathematics*. Cambridge, Mass: MIT Press 1980, 477 p.
- [16] CHIOCCARIELLO, A., LECCIOLI, N.C., ORESTE, C. Four Steps to the Right. In: diSESSA, A.A HOYLES, C., NOSS, R., EDWARDS, L.D. (Eds.) *Computers and Exploratory Learning*, Nato ASI Subseries F 146, Sringer 1995, 482 p.
- [17] WATT, S., Syntonicity and the psychology of programming. *Proc of 10th Annual Workshop of Psychology of Programming Interest Group*, Milton Keynes, UK 1998, pp. 75-86.
- [18] VOET, M., De Wever, B. Effects of immersion in inquiry-based learning on student teachers' educational beliefs. *Instr Sci* 46, 2018, pp. 383-403.

Mobilná aplikácia na interaktívne vychádzky

Mobile application for interactive trips

Miroslav Blšák

Katedra informatiky FPV UMB

Tajovského 40

974 01 Banská Bystrica

Slovensko

miroslav.blsak@student.umb.sk

Dana Horváthová

Katedra informatiky FPV UMB

Tajovského 40

974 01 Banská Bystrica

Slovensko

dana.horvathova@umb.sk

ABSTRACT

The article deals with the use of mobile information and communication technologies in the out-of-school environment. We want students not to spend a lot of time by using mobile phones, tablets or computers, but we also want from them to be digitally literate. It is in this article that we want to show how we can combine these two differences into one entity. In an effort to solve this paradox, we have created a mobile application for the popular encryption game called Night Ciphers, which takes place in Banská Bystrica every year. Subsequently, it was tested by high school students in the form of a competition. We verified the adequacy and complexity of the used tasks, as well as the intuitiveness of the control or suitability of the graphic side of the application in the form of a questionnaire, which is located in the application itself. There was also an interview after the competition with a certain group of students.

Keywords

Mobile technology. Block programming. Didactic trip. Out-of-school environment.

ABSTRAKT

Článok sa zaoberá využitím mobilných informačných a komunikačných technológií v mimoškolskom prostredí. Na jednej strane chceme, aby žiaci netrúvali veľa času používaním mobilov, tabletov, počítačov, ale zároveň chceme, aby boli digitálne gramotní. Práve v tomto článku chceme ukázať, ako dokážeme tieto dve odlišnosti spojiť do jedného celku. V snahe riešiť tento paradox, sme pre obľúbenú šifrovaciu hru s názvom *Nočné šifry*, ktorá sa koná v Banskej Bystrici každý rok, vytvorili mobilnú aplikáciu. Následne ju otestovali žiaci stredných škôl formou súťaže. Overenie primeranosti a náročnosti použitých príkladov, ako aj intuitivnosti ovládania, či vhodnosti grafickej stránky aplikácie, sme realizovali formou dotazníka, ktorý sa nachádza v samotnej aplikácii. Nechýbal ani rozhovor po ukončení súťaže s určitou skupinou žiakov.

Kľúčové slová

Mobilné technológie. Blokové programovanie. Didaktická vychádzka. Mimoškolská činnosť.

1 ÚVOD

Pre súčasných aj budúcich učiteľov akademických predmetov, bol Ján Amos Komenský jednou z dôležitých postáv pedagogickej histórie. Nielenže chcel vzdelanie pre všetkých bez rozdielu, presadzoval zásadu názornosti, ale aj zastával myšlienku „škola hrou“. Dve posledné skutočnosti nás inšpirovali k tomu, aby sme vytvorili mobilnú aplikáciu, ktorá kopíruje priebeh Nočných šifier, avšak v digitálnej podobe. Aplikácia môže byť výborným nástrojom, pretože didaktická zásada názornosti v nej má dominantné miesto. Aplikáciu sme následne otestovali žiakmi

stredných škôl počas súťaže, ktorú sme zorganizovali v spolupráci s katedrou informatiky a katedrou matematiky FPV UMB.

2 ŠIFRY A ŠIFROVANIE

Šifrovanie bolo kedysi iba výsadou armády, tajných služieb a iných vládnych agentúr. V súčasnosti, kedy sú digitálne technológie prirodzenou súčasťou nášho každodenného života a my sa čím ďalej tým viac otvárame svetu, je nevyhnutné, aby sme sa zamýšľali nad tým, ako naše dáta ochrániť. Potreba ochrany nášho súkromia a majetku narastá hlavne v súvislosti s používaním internetových a mobilných technológií na komunikáciu napr. sociálnych sieťach.

Kryptológia je veda, ktorá sa zaoberá šifrovaním a dešifrovaním a skladá sa z dvoch častí – kryptografie a kryptoanalýzy. Prvá menovaná časť študuje, ako navrhnúť šifrovaciu metódu a z akých teoretických aspektov pri tom vychádzať. Druhá je zameraná na opak prvej, teda metódy lúštenia šifier [1].

Šifrovanie je proces zmeny obsahu otvoreného textu za účelom ukrytia prenášanej informácie a dešifrovanie (inverzná operácia) je proces spätného získania správy vo forme otvoreného textu z jej zašifrovanej podoby (zašifrovaného textu). Tento proces teda transformuje zašifrovaný text na otvorený text. Kryptografický algoritmus, nazývaný tiež šifra, je jednoducho postupnosť určitých procesov, ktoré zahŕňajú šifrovanie a prislúchajúce dešifrovanie. Šifra je matematická operácia špeciálne navrhnutá tak, aby zakryla, resp. znemožnila odhaliť obsah dát [2].

Je dobré všetky tieto pojmy poznať a rozumieť im. Preto je nevyhnutné, aby sa problematika kryptografie zmysluplne začlenila do výučby predmetu informatika. Nasvedčuje tomu aj fakt, že podľa inovovaného štátneho vzdelávacieho programu predmetu Informatika „*Poslaním výučby informatiky je viesť žiakov k pochopeniu základných pojmov, postupov a techník používaných pri práci s údajmi a toku informácií v počítačových systémoch*“ [3]. Vyučovanie šifrovania tento cieľ veľmi pekne spĺňa. Každá ďalšia aktivita, ktorá zábavnou formou rozširuje rozhľad detí a študentov v tejto oblasti, je samozrejme tiež vítaná.

3 MOBILNÉ TECHNOLOGIE VO VZDELÁVANÍ

Hlavná výhoda využitia mobilných technológií vo vzdelávaní je fakt, že môžu prispieť k skvalitneniu a zefektívneniu vzdelávacieho procesu. Ich výhody využitia vo vzdelávaní podľa Verešovej sú nasledovné [4]:

- zvyšuje atraktivnosť, motiváciu a celkovú chuť pracovať,
- vhodný pre žiakov so znevýhodnením (napr. sluchovým postihnutím) práve prostredníctvom názornosti a osobnej skúsenosti,

- umožňuje motivovať podpriemerných aj nadaných žiakov k práci.

Prínos, ktorý nám mobilné technológie ponúkajú práve vo vzdelávaní zhrnula Britská agentúra pre vzdelávanie a komunikáciu [5]:

- spolupráca medzi žiakmi sa zvyšuje,
- podporuje medzi nimi komunikáciu a vymieňanie si skúseností, názorov a vedomostí,
- umožňuje to, aby sa žiak zamerával na interpretáciu odpovedí a na stratégiu.

Samozrejme, existuje veľa spôsobov implementácie informačno-komunikačných technológií do vyučovania a rôzni autori ako Mišut [6], Gregaňová [7], Šedivý, Fulier, Michalička, Čerťková [8] a iní opisujú detailnejšie ako ich vhodným spôsobom použiť.

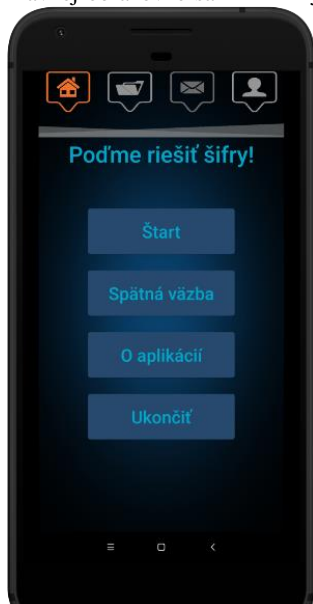
Zhrnuté poznatky o šifrách a šifrovaní, ako aj o výhodách využitia mobilných technológií v procese vzdelávania boli pre nás východiskom pre vytvorenie mobilnej aplikácie, ktorú sme použili pri realizácii matematicko-logickej súťaže pre žiakov stredných škôl. Okrem toho medzi východiská patrila aj regionálna výchova, keďže sa súťaž konala v centre mesta Banská Bystrica. Regionálna výchova ako jedna z prierezových tém vo vyučovacom procese prispieva k zvýšeniu vedomostnej úrovne a motivácii žiakov spoznať viac svoje mesto a okolie.

4 MOBILNÁ APLIKÁCIA

Najskôr vysvetlíme, čo sú to *Nočné šifry* a ako nami vytvorená aplikácia funguje.

Nočné šifry sú šifrovacou hrou, v ktorej súperia medzi sebou trojčlenné tímy. Tie majú za úlohu počas hry na stanoviskách vylúštiť jednotlivé šifry, ktorých riešenie ich posunie na ďalšie stanovisko. Vyhrá tím, ktorý získa najviac bodov [9].

Naša aplikácia funguje na podobnom princípe. Podmienkou správneho fungovania je mobilné zariadenie pripojené k internetu s aktívnym globálnym lokalizačným systémom (zapnutým GPS). Zamerajme sa najskôr na funkcie tlačidiel v hlavnom menu. Na hlavnej obrazovke sa nachádzajú štyri tlačidlá a to: *Štart*, *Spätná väzba*



Obrázok 1 Hlavné menu aplikácie. Zdroj: [Autor].

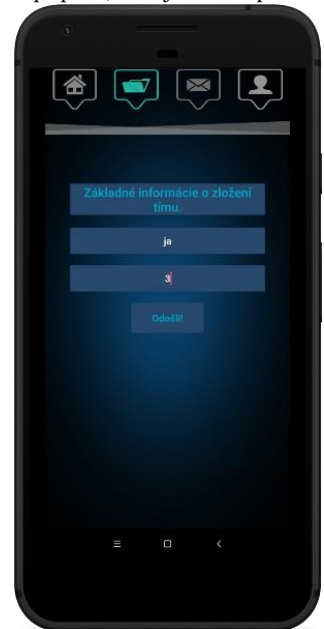


Obrázok 2 Scéna „O aplikácii“. Zdroj:[Autor].

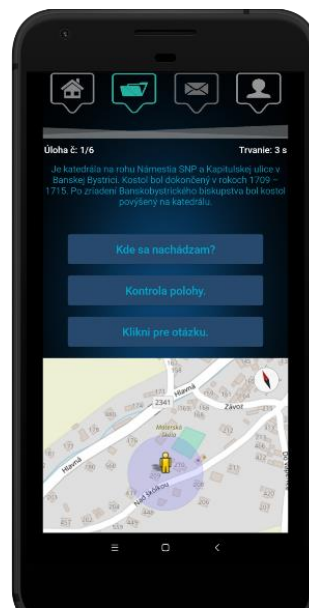
väzba, *O aplikácii* a *Ukončiť* (obr. 1). V karte *O aplikácii* nachádzame základné informácie o našej aplikácii (obr. 2). Tlačidlo *Spätná väzba* zobrazí scénu s dotazníkom, ktorý obsahuje otázky smerované k samotnej aplikácii a k príkladom použitým v súťaži (obr. 3). Nasleduje tlačidlo *Štart*. Pred samotným súťažením si aplikácia z databázy stiahne konkrétne miesta, ich GPS súradnice s indíciami a príkladmi. Potom musí používateľ zadať údaje o tíme (obr. 4). Následne sa v hornej časti zobrazí pomôcka, navádzajúca nás, kam sa máme vydať (obr. 5). Indície nás zavedú k historickým pamiatkam, ale aj rôznym iným kultúrno-spoločenským miestam mesta. Keď sa v okolí daného miesta nachádzame, overíme polohu a v prípade, že je naša poloha



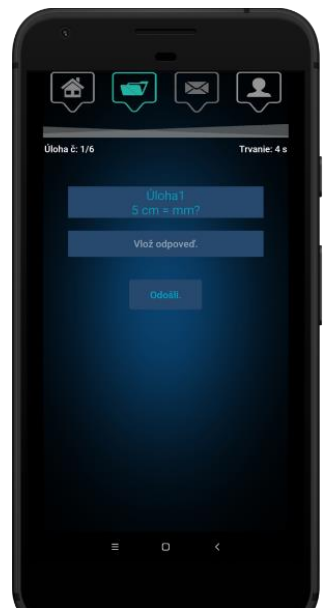
Obrázok 3 Karta „Spätná väzba“. Zdroj:[Autor].



Obrázok 4 Scéna „Štart“. Zdroj:[Autor].



Obrázok 5 Indícia so zobrazením našej polohou. Zdroj:[Autor].



Obrázok 6 Súťažná otázka (ilustračná). Zdroj:[Autor].

správna, dostaneme príklad, ktorý máme vyriešiť (obr. 6). V prípade nesprávnej polohy budeme upozornení na nesprávnu polohu chybovým hlásením. Po zadaní odpovede dostaneme ďalšiu indíciu, kam máme ísť a tak pokračujeme cyklicky, kým nevyriešime všetky úlohy.

Pre správne fungovanie aplikácie je nutné mať zapnutú lokalizáciu zariadenia a povoliť aplikácii prístup k polohe zariadenia. Samozrejmosťou sú taktiež aktívne mobilné dáta. Je to z dôvodu, aby si aplikácia mohla stiahnuť údaje z databázy do pamäte, ale aj pre odoslanie vyplneného dotazníka.

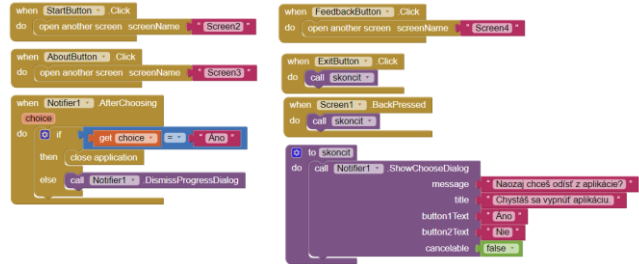
Ako sme spomínali, všetko je realizované prostredníctvom internetu a práce s databázami. Prikladáme ilustračný obrázok, ako vyzerá naša databáza, z ktorej aplikácia čerpá všetky údaje a informácie.

	A	B	C	D	E	F	G	H
1	Otázka	Odpov	Latitude	Longitude	polomer	Kde majú ísť?		Najvzdialene Najvziales
2	Kolko je 2 + 2?	4	48.7303279	19.136110	0.0014016	Je moderným a príťažlivým miestom, kt	48.731092	19.13731
3	90° + 90° = °?	180	48.7351290	19.149900	0.0003815	Jeden z najvýznamnejších pamätníkov	48.735240	19.15027
4	5*5?	10	48.7355300	19.146240	0.0001991	Je katedrála na rohu Námestia SNP a	48.735643	19.14607
5	2*2?	4	48.7314730	19.146500	0.0002235	Je železničná zastávka v Banskej Bystr	48.731496	19.14675
6	5 cm = mm?	50	48.7331060	19.146220	0.0002358	Patrí medzi významné slovenské divad	48.733021	19.14600
7	Ktoré číslo nasi	2	48.7368710	19.137850	0.0011934	Vznikol v roku 1896, kedy plán konečn	48.736100	19.13698

Obrázok 7 Databáza s údajmi. Zdroj:[Autor].

4.1 Zdrojové kódy

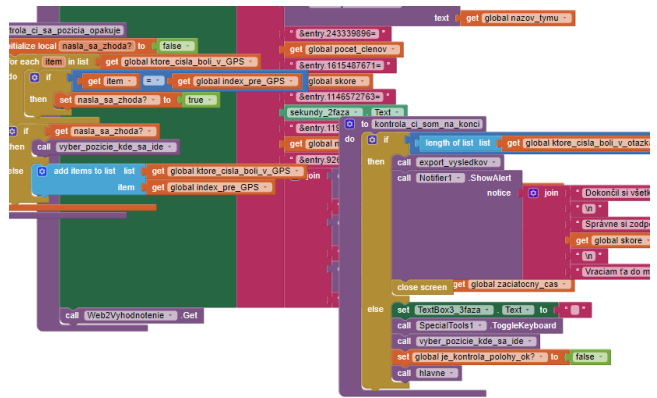
Na vývoj aplikácie sme zvolili grafický editor App Inventor, ktorý využíva blokové programovanie. Funkcie aj nástroje, ktoré nám spomínaný editor ponúka, sú dostatočné pre vytvorenie plnohodnotnej aplikácie. Aplikácia je dostupná len na operačnom systéme Android. Z dôvodu, aby sme zachovali čitateľnosť obrázkov a zmestili sa do rozsahu strán práce, neukážeme každý jeden blok v každej scéne, ale len vybrané bloky.



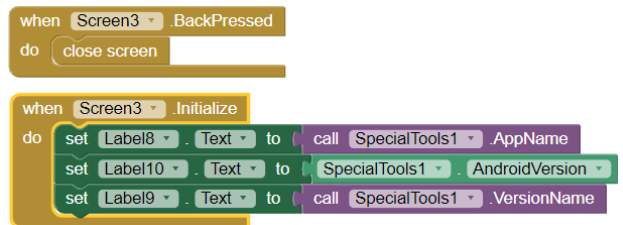
Obrázok 8 Bloky hlavného menu. Zdroj:[Autor].



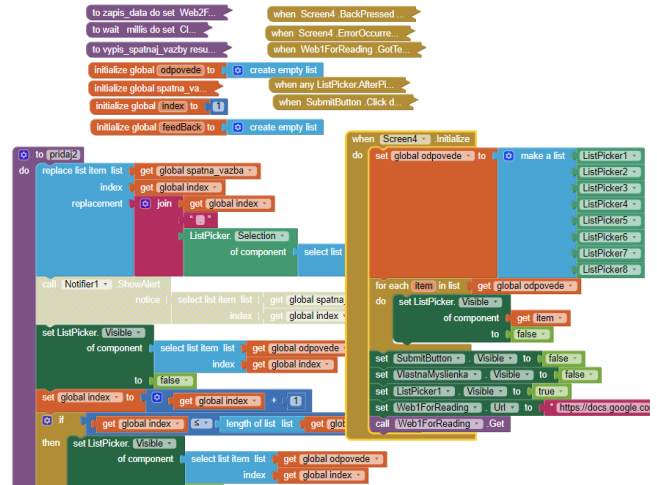
Obrázok 9 Skryté bloky v scéne „Štart“. Zdroj:[Autor].



Obrázok 10 Ukážka pár blokov v scéne „Štart“. Zdroj:[Autor].



Obrázok 11 Bloky v scéne „O aplikácii“. Zdroj:[Autor].



Obrázok 12 Bloky v scéne s dotazníkom. Zdroj:[Autor].

5 SÚŤAŽ

Súťaž bola určená pre žiakov všetkých stredných škôl. Zapojiť sa mohli dvoj až trojčlenné tímy. Naším cieľom bolo spojiť informatiku, matematiku a geografiu do jedného celku. Chceli sme využiť negatívny fenomén, v ktorom žiaci trávajú veľké množstvo času za obrazovkami rôznych zariadení. Aplikáciou s prvkami súťaže ako súčasťou didaktickej vychádzky sme poukázali na dominantné vlastnosti a výhody mobilných technológií vo vzdelávaní. Vzhľadom na možnosť ukladania dát, potrebných v súťaži do databázy, ako aj ďalších informácií čerpaných z internetu, nič nebráni použitiu aplikácie v iných predmetoch, prípadne jej jednoduchšej adaptácii pre ďalšie podobné účely.

5.1 Priebeh

Oslovili sme učiteľov na vybraných stredných školách s prosbou o pomoc pri propagácii akcie. Dokonca sme aj osobne rozdávali

pred školou informačné letáky, v ktorých sa nachádzali základné informácie o súťaži. Následne sme začali dostávať otázky od žiakov aj učiteľov, čo naznačovalo záujem o pripravovanú akciu a utvrdilo nás to v našom zámere.

Súťaž sme zorganizovali na Námestí SNP v Banskej Bystrici a aj napriek zložitej epidemiologickej situácii a nepriaznivej predpovedi počasia sme sa ju rozhodli uskutočniť dňa 29.9.2020. Nakoniec nám počasie doprialo a uskutočnenie akcie nebolo ohrozené žiadnymi inými faktormi. Samotnej súťaži sa zúčastnilo 15 žiakov stredných škôl. Mrzelo nás, že dva dni pred termínom uzavreli na jednom banskobystrickom gymnáziu niekoľko tried do karantény, odkiaľ mali prísť viacerí súťažiaci. Ďalší faktor, ktorý ovplyvnil počet súťažiacich, bola spomínaná situácia ohľadom vírusu covid-19, nakoľko sa žiaci obávali o svoje zdravie a súťaže sa na koniec rozhodli nezáúčastniť.

Žiaci sa v úvode nahlásili (meno a priezvisko) do prezenčnej listiny. Boli im rozdávané čísla, ktoré slúžili ako lístok do tomboly.



Obrázok 13 Úvodná registrácia. Zdroj:[Organizačný tím].

Potom dostali organizačné pokyny k celej súťaži, mobilnej aplikácii aj vyhodnoteniu.



Obrázok 14 Organizačné pokyny. Zdroj:[Organizačný tím].

Následovalo samotné jadro súťaže. Tímy si nainštalovali aplikáciu z Obchodu Play https://play.google.com/store/apps/details?id=ap.inventor.ai_mirob333.Sifry&hl=sk, zadali vstupné údaje a začali riešiť šifry, ktoré ich zaviedli k pamiatkam v blízkosti Námestia SNP. Pri každom takomto spoločenskom/kultúrnom mieste dostali matematickú úlohu.

Na všetkých stanovištiach sa nachádzala dvojica organizátorov, ktorá zapisovala odpovede tímov na danú úlohu, ich čas odchodu a v prípade problémov pomohla súťažiacim.



Obrázok 15 Súťažný tím na stanovisku v hradnom areáli. Zdroj:[Organizačný tím].



Obrázok 16 Iný súťažný tím na tom istom mieste. Zdroj:[Organizačný tím].



Obrázok 17 Súťažný tím nachádzajúci sa pri štátnej opere. Zdroj:[Organizačný tím].

Po vyriešení všetkých piatich úloh všetkými súťažiacimi prebiehalo vyhodnotenie pred budovou Banskobystrického samosprávneho kraja.

5.2 Vyhodnotenie

Za každú správne vyriešenú úlohu získal tím 1 bod. Maximálny počet bodov bol 5. V prípade nesprávnej odpovede sa na konto tímu nepočítali žiadne body (ani plusové, ani mínusové body). Ak sa niektoré tímy zhodovali v počte bodov, vyhral ten, ktorý skončil skôr, t. j. tím, ktorý vyplnil všetky odpovede v kratšom časovom intervale.

Vďaka viacerým inštitúciám sme mali pripravené výhry pre prvé tri tímy a ďalšie vecné ceny pre všetkých súťažiacich z tomboly, napríklad:

- logické a spoločenské hry,
- knihy,
- tričká s potlačou,
- vitamínové balíčky,
- bezdrôtové slúchadlá,
- kupóny (pizza, zážitok vo virtuálnej realite),
- propagačné materiály (perá, logo vytlačené 3D tlačiarňou a iné) .



Obrázok 18 Stôl s cenami pre prvé tri tímy a ceny v tombole.
Zdroj:[Organizačný tím].

Z dôvodu nižšieho počtu účastníkov a pomerne vysokej pravdepodobnosti výhry v tombole získal každý jeden účastník nejakú cenu – sme za to veľmi radi.

5.3 Spätná väzba

Spätná väzba spočívala hlavne v pozorovaní, keďže ide o prirodzené správanie človeka (ktoré sa skoro nedá oklamať), ktoré považujeme za najdôveryhodnejšie. Môžeme skonštatovať, že sa to žiakom páčilo, príklady ich bavili, dokonca sme videli, ako jeden tím uteká na stanovisko, aby bol prvý.

Od žiakov sme si pýtali spätnú väzbu prostredníctvom dotazníka nachádzajúceho sa v aplikácii. Väčšina žiakov uviedla, že sa im páčil takýto koncept súťaže. Aplikácia im vyhovovala po obsahovej aj po grafickej stránke, nenašli v nej žiadne chyby, príklady boli formulované zrozumiteľne.

Členovia organizačného tímu tiež podali spätnú väzbu. Páčila sa im myšlienka a určite by do podobného projektu išli znovu. Vyskytli sa len drobné chyby, ktoré súviseli s našimi skúsenosťami s organizovaním takejto akcie. Súťaž sme organizovali prvýkrát. Je nutné podotknúť, že nie sme organizátori udalosti *Nočné šifry*, ktoré sa konajú každý rok v Banskej Bystrici, len sme sa inšpirovali myšlienkou podobnú aktivitu uskutočniť za pomoci mobilnej aplikácie zasadené do mimoškolského vzdelávania.

ZÁVER

Sme radi, že sa nám podarilo súťaž zorganizovať aj napriek nepriaznivej epidemiologickej situácii. Táto súťaž za pomoci mobilnej aplikácie mala na FPV UMB premiéru a zdá sa, že sa nám táto zaujímavá forma využitia moderných technológií vo vzdelávaní prostredníctvom hry osvedčila. V budúcnosti máme motiváciu vyhnúť sa drobným chybám, aplikáciu ešte vylepšiť a súťaž znovu zopakovať. Bola to nezabudnuteľná skúsenosť a určite sme sa poučili z pár chýb, ktoré sme vykonali.

Veríme, že nabudúce sa nám podarí zorganizovať súťaž ešte lepšie a snáď už bez obmedzujúcich opatrení.

POĎAKOVANIE

Ďakujeme katedre matematiky Fakulty prírodných vied UMB, katedre informatiky Fakulty prírodných vied UMB, samotnej fakulte prírodných vied UMB aj Univerzite Mateja Bela v Banskej Bystrici za podporu pri organizovaní súťaže a za poskytnutie vecných cien pre prvé tri miesta aj cien do tombole.

Osobitné ďakujem patrí Banskobystrickému samosprávnemu kraju za finančnú podporu pri organizovaní tejto súťaže.

Veľké ďakujem patrí organizačnému tímu - bez nich by sme to nezvládli.

Príspevok je podporený grantovým projektom Implementácia nových trendov v informatike do výučby algoritmického myslenia a programovania v predmete informatika v sekundárnom vzdelávaní, projekt KEGA 018UMB-4/2020.



Obrázok 19 Členovia organizačného tímu.
Zdroj:[Organizačný tím].

BIBLIOGRAFICKÉ ODKAZY

- [1] HURAJ, L. 2002. *Nebojme sa šifrovania*. Bratislava : Metodicko-pedagogické centrum, 2002. 50 s. ISBN 80-8052-160-3
- [2] SORIANO, M. 2017. *Kryptografia, kyberkriminalita*. Praha : TechPedia, 2017, 39s., ISBN 978-80-01-06205-0. Dostupné na: [https://techpedia.fel.cvut.cz > download](https://techpedia.fel.cvut.cz/download) [cit. 16.12.2020].
- [3] ŠTÁTNY PEDAGOGICKÝ ÚSTAV, 2015. *Štátny vzdelávací program pre gymnáziá* [online]. Dostupné na: <https://www.statpedu.sk/sk/svp/inovovany-statny-vzdelavaci-program> [cit. 3.12.2020].
- [4] VEREŠOVÁ, J. 2016. *Využívanie projekčnej techniky a interaktívnej technológie vo výchovno-vzdelávacích zariadeniach*. Bratislava : Metodicko-pedagogické centrum, 2019. 19 s.
- [5] BAYA'A, N., DAHER, W. 2013. *Mathematics Teachers' Readiness to Integrate ICT in the Classroom: The Case of Elementary and Middle School Arab Teachers in Israel*. In International Journal of Emerging Technologies in Learning. 2013. 70 p. Vol. 8, No. 1. ISSN 1863-0383.
- [6] MIŠÚT, M. 2013. *IKT vo vzdelávaní*. Trnava : Pedagogická fakulta Trnavskej univerzity v Trnave, 2013. ISBN 978-80-8082-695-6.
- [7] GREGÁNOVÁ, R. 2005. *Informačné technológie vo výučbe matematiky*. Zborník : Univerzitné vzdelávanie po vstupe do Európskej únie. Nitra : Fakulta ekonomiky a manažmentu Slovenskej poľnohospodárskej univerzity v Nitre, 2005.
- [8] ŠEDIVÝ, O., FULIER, J., MICHALIČKA, P., ČERETKOVÁ, S. 2005. *Informačné a komunikačné technológie vo vyučovaní matematiky*. Nitra : Katedra matematiky Fakulty prírodných vied Univerzity Konštantína Filozofa v Nitre, 2007. 212 s. ISBN 80-8050-925-5.
- [9] *Nočné šifry*. <http://www.outdoor-institute.sk/51/n0-c-n-sh1-free>

Porovnanie abstraktných úloh v stredoškolských kategóriách súťaže iBobor

Comparison of abstract tasks in high school categories in the Bebras competition

Lucia Budinská

KDMFI, Fakulta matematiky, fyziky a informatiky
Univerzita Komenského v Bratislave

Mlynská Dolina
842 48 Bratislava
Slovensko

Lucia.Budinska@fmph.uniba.sk

Michal Winczer

KDMFI, Fakulta matematiky, fyziky a informatiky
Univerzita Komenského v Bratislave

Mlynská Dolina
842 48 Bratislava
Slovensko

winczer@fmph.uniba.sk

ABSTRACT

In this paper, we analyse three graph tasks from Slovak Bebras competition included in year 2020/21. These three tasks are focused on abstraction use, more precisely how to describe real situation with a graph. Tasks were in both Junior and Senior categories in competition, which means they were solved by high school students. Next, the success rate of students in these tasks was analysed. We were interested in distribution of answers, and differences between boys and girls and differences in between individual school years.

Keywords

Bebras competition. Abstraction. Graph. High school.

ABSTRAKT

V tomto článku opíšeme analýzu troch grafových úloh zo súťaže iBobor, ktoré boli zaradené v školskom roku 2020/21. Všetky tri úlohy sa zameriavajú na použitie abstrakcie, konkrétne ako opis reálnej situácie pomocou grafu. Tieto úlohy boli zaradené do kategórií Juniori a Seniori, čiže ich riešili všetci stredoškoláci. Následne sme skúmali úspešnosť žiakov v týchto úlohách. Zaujímalo nás rozdelenie jednotlivých odpovedí, ale tiež aj porovnanie úspešnosti chlapcov a dievčat a porovnanie jednotlivých ročníkov medzi sebou.

Kľúčové slová

Súťaž iBobor. Abstrakcia. Graf. Stredná škola.

1 ÚVOD

Súťaž v riešení informatických úloh iBobor sa uskutočnila na Slovensku v šk. roku 2020/21 už 14. raz. Zúčastňuje sa na nej každoročne desaťtisíce žiakov základných a stredných škôl. V tomto školskom roku to bolo 55 064 súťažiacich z 804 škôl¹, čo je menej ako po minulé roky. Pokles v súťažiacich bol spôsobený tým, že v čase súťaže bola pre žiakov od druhého stupňa zavedená dištančná výučba, preto súťažili z domu. Aj napriek nižšej účasti nám riešenia súťažiacich poskytujú cenný zdroj dát na ďalšie skúmanie.

V predchádzajúcich článkoch [1, 2] sme sa sústreďovali na analýzu grafových úloh a porovnanie výsledkov chlapcov a dievčat v týchto úlohách. Kým v úlohách určených pre základnú školu je jasný rozdiel medzi úlohami, v ktorých sú úspešnejší chlapci (abstraktné znázornenie štruktúry, potreba prevedenia

algoritmu na štruktúre, vytváranie stratégie, optimalizácia) a v ktorých sú úspešnejšie dievčatá (zložitejšie štruktúry, jednoduchá manipulácia so štruktúrou, prevádzanie medzi reprezentáciami), v úlohách zo stredoškolských kategórií sa takéto delenie nepotvrdilo. Väčšinu grafových úloh riešili s vyššou úspešnosťou chlapci, jednoduchšie úlohy, respektíve úlohy spĺňajúce kritériá pre dievčenské úlohy zo ZŠ, riešili chlapci aj dievčatá na SŠ rovnako.

Grafové úlohy môžu rozvíjať rôzne časti informatického myslenia a informatických zručností. Môžu byť použité ako modely, čiže abstraktné či zjednodušené znázornenie reality, alebo môžu byť použité pri rôznych algoritmoch [3]. Práve tvorba modelov rozvíja abstraktné myslenie žiakov, a to na rôznych úrovniach. Na základe našich skúseností so žiakmi ZŠ a SŠ, ale aj študentami VŠ, sa nám ukazuje, že koncepty zovšeobecnenia a abstrakcie, tak dôležité pri programovaní ako aj v informatike [4], robia žiakom pomerne veľké ťažkosti.

Preto sa v tomto článku zameriame na analýzu niekoľkých úloh, ktoré od riešiteľa vyžadovali nejakú mieru abstrakcie. Porovnali sme ako boli pri ich riešení úspešné dievčatá, chlapci a tiež žiaci rôzneho veku.

2 METODOLÓGIA

Graf predstavuje abstrakciu vzťahov medzi nejakými konkrétnymi objektmi. Z toho dôvodu bolo prirodzeným záujmom nájsť práve také úlohy, ktoré mali v zadaní použitý graf a úloha bola o porovnaní konkrétnej situácie s jej abstraktnou reprezentáciou grafom. Vynechali sme úlohy, v ktorých bolo nutné použiť nejakú stratégiu či algoritmus.

Z viacerých úloh, ktoré sme takto vybrali sme sa zamerali len na tie, ktoré boli zároveň vo viacerých kategóriách, konkrétne v kategórii Juniori (1. a 2. ročník SŠ) a Seniori (3. až 5. ročník SŠ). Umožnilo nám to porovnanie úspešnosti ich riešenia v závislosti od veku riešiteľov. Takéto úlohy boli tri, všetky s výberom jednej zo štyroch možností. V našich výsledkoch uvádzame poradie odpovedí pred ich náhodným pomeškaním (riešitelia ich mali v súťaži poprehadzované). Počty žiakov a ich štruktúra je v Tabuľke 1. Žiakov osemročných gymnázií sme sčítali s alternatívou štvorročnej školy z dôvodu, že ich počet bol malý. Rovnako sme pripočítali žiakov 5. ročníka, čiže maturantov bilingválnych gymnázií, keďže žiakov nižších ročníkov týchto škôl nevieme odlišiť a pripočítavame ich k rovnakému ročníku štvorročných stredných škôl.

¹ <http://ibobor.sk/vyhodnotenie2020.php>

Tabuľka 1 - Rozdelenie žiakov po ročníkoch

	1. ročník + kvinta	2. ročník + sexta	3. ročník + septima	4. a 5. ročník + oktáva	SPOLU
žena	1652	1299	793	543	4287
muž	1946	1337	1076	817	5176
SPOLU	3598	2636	1869	1360	9463

Na analýzu dát sme použili χ^2 a z-skóre [5] a porovnali sme dosiahnuté výsledky pre jednotlivé úlohy podľa ročníka riešiteľov a tiež podľa pohlavia riešiteľov. Konkrétne sme vytvárali kontingenčné tabuľky, ktoré v riadku obsahovali jednotlivé odpovede a možnosť neodpovedania, a v stĺpcoch boli jednotlivé ročníky súťažiacich resp. chlapci a dievčatá. Bunky v tabuľke tak obsahovali súhrnný počet odpovedí v danej skupine. Z-skóre nám umožnilo vidieť, kde v tabuľke je porušená rovnosť, a tiež na akej hladine. Pre štatistickú významnosť sme zvolili $\alpha = 0.05$, čo predstavovalo z-skóre ≥ 1.96 .

Tie isté úlohy sme dali riešiť aj asi dvadsiatim študentom našej fakulty, išlo o skupinu magisterských študentov aplikovanej informatiky a skupinu študentov bakalárskeho stupňa učiteľstva informatiky. Ich výsledky neporovnávame s výsledkami stredoškôlkov, dôležitá bola pre nás viac slovná spätná väzba k jednotlivým úlohám.

3 Analyzované úlohy

V nasledujúcich kapitolách opisujeme trojicu úloh, ktoré sme vybrali na analýzu. Ide o úlohy **Mestá a diaľnice** (v oboch kategóriách zaradená ako stredne ťažká úloha), **Tajná správa** (v oboch kategóriách zaradená ako stredne ťažká úloha) a **Víťaz šachového turnaja** (v oboch kategóriách zaradená ako ťažká úloha). Zaradovanie úloh podľa náročnosti sa robí pred súťažou, rozhodujú o nej autori úloh a organizátori, ako je vidieť aj z výsledkov prezentovaných v nasledujúcich častiach, nie vždy sa zhoduje s reálnou náročnosťou úlohy pre súťažiacich.

3.1 Mestá a diaľnice

Úloha **Mestá a diaľnice**, pozri Obrázok 1, je slovenská, do medzinárodnej databázy bola poslaná pôvodne vo verzii, kde miesto diaľnic boli rieky. Na základe recenzií sa motivácia úlohy upravila na diaľnice, a na medzinárodnom workshope sa úloha iba kozmeticky upravovala. Zaujímavé je, že pri hodnotiacom procese nám prišla aj spätná väzba, ktorej autor predpokladal, že úloha sa nehodí pre stredoškolské kategórie a mala by byť zaradená do kategórií na konci základnej školy. Keďže sme mali skúsenosti s podobnými úlohami aj z minulých rokov, v slovenskej súťaži sme ju ponechali pre starších súťažiacich, a ako je ďalej z výsledkov viditeľné, ani v týchto kategóriách nepatrla medzi najľahšie.

Súťažiaci majú v úlohe prevádzať medzi dvoma reprezentáciami, pričom je potrebné, aby si dôkladne prečítali zadanie. Predpokladáme, že práve nepozorne prečítané zadanie bola najčastejšia chyba, a preto vo výsledkoch vidieť výraznú prevahu odpovede A nad správnu odpoveďou C. Kompletne výsledky sú v Grafe 1 Je z nich vidieť, že starší žiaci úlohu riešili úspešnejšie ako mladší, že odpoveď A bol úspešne zvolený distraktor

a odpovede B a D zvolila približne rovnako veľká časť riešiteľov nezávisle od ich veku.

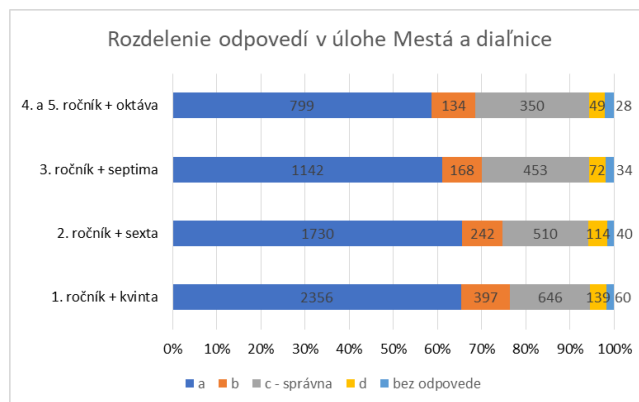
Na mape je 5 miest (čieme body) a 4 diaľnice (farebné čiary).

Mapu chceme reprezentovať diagramom, v ktorom

- mestá reprezentujú kruhy,
- dve mestá sú spojené čiarou práve vtedy, keď cez ne prechádza tá istá diaľnica.

Ktorý diagram reprezentuje správne mapu?

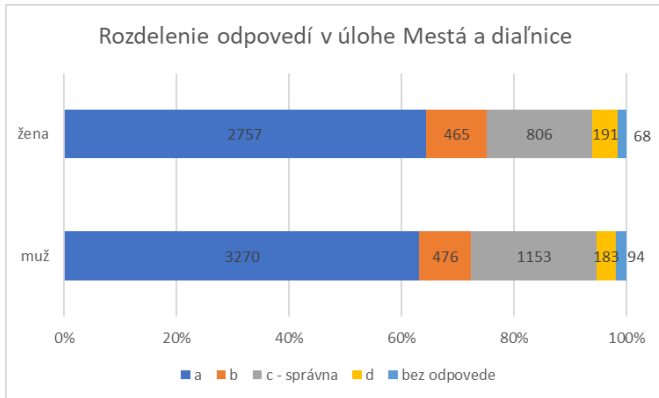
Obrázok 1 - Zadanie úlohy Mestá a diaľnice



Graf 1 - Odpovede v úlohe Mestá a diaľnice podľa ročníka

Pri porovnaní odpovedí chlapcov a dievčat sa ako štatisticky významné ukázali rozdiely pri voľbe nesprávnych odpovedí B a D (dievčatá ich volili častejšie) a pri správnej odpovedi C (chlapci ju vyberali častejšie). Rozdiel pri výbere odpovede A, či pri neodpovedaní, bol štatisticky nevýznamný. Kompletne výsledky vidíme v Grafe 2.

Z pohľadu kategorizácie [1] je potrebné v úlohe prevádzať medzi dvoma reprezentáciami a to sa pre súťažiacich v nižších kategóriách ukazovalo ako tzv. „dievčenská úloha“, a tie v nasledujúcom výskume na stredoškolských kategóriách mali neutrálne výsledky [2], tu sa to nepotvrďuje. Je však nutné poznamenať, že pri voľbe najčastejšej odpovede, čiže distraktora A, nebol štatisticky významný rozdiel.



Graf 2 - Odpovede v úlohe Mestá a diaľnice podľa pohlavia

3.2 Tajná správa

V úlohe Tajná správa, pozri Obrázok 2, riešiteľ dostal časti grafu, ktoré reprezentovali vzťahy medzi agentmi a ich prístupu ku správam. Informácie potrebné na zistenie, čo aká správa znamená, boli zadané v texte. Riešiteľ mal na základe týchto informácií zistiť, ktorý agent má prístup ku všetkým správam. Z formálneho hľadiska šlo o orientované grafy a prepájanie informácií.

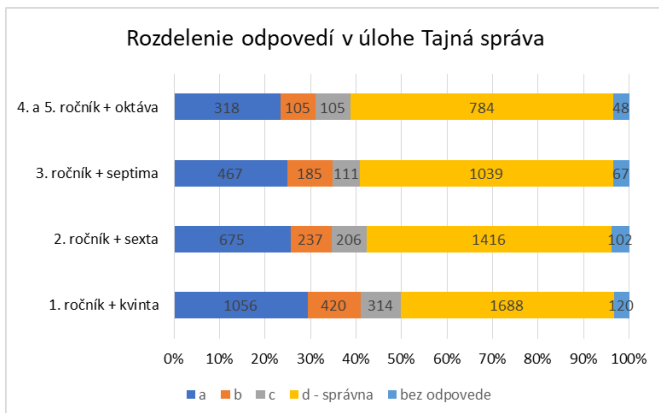
Výsledky úlohy v jednotlivých kategóriách sú v Grafe 3.

Veľký šéf rozdelí tajnú správu medzi svojich šiestich agentov, ktorým ale neverí rovnako. Preto ju rozdelí na štyri časti (1 až 4) a každej z nich priradí úroveň utajenia. Správu s úrovňou „-“ smie čítať len agent, ktorý ju dostal (na obrázku ide bodkovaná šípka k agentovi, ktorý správu dostal). Správu s úrovňou „#“ smie čítať len agent, ktorý ju dostal a jeho podriadení (na obrázku ide sivá šípka od podriadeného k nadriadenému). Správu s úrovňou „+“ smie čítať každý agent (preto od nej nevedie žiadna bodkovaná šípka). Nasledujúci obrázok zobrazuje agentov a ktoré správy dostali.

Ktorý agent si môže prečítať všetky časti správy?

A) B) C) D)

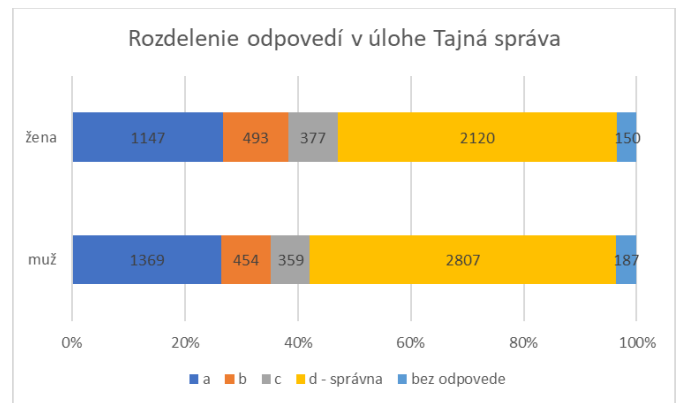
Obrázok 2 - Zadanie úlohy Tajná správa



Graf 3 - Odpovede v úlohe Tajná správa podľa ročníka

Z grafu úspešnosti riešenia je vidieť, že úloha patrila v oboch kategóriách k stredne ťažkým, úspešne ju vyriešilo v každej kategórii približne 50 až 55 % riešiteľov, pričom úspešnosť riešenia rástla s vekom riešiteľov. Z chybných odpovedí sa dá usúdiť, že odpoveď A bol najlepší distraktor, kde si súťažiaci asi neprečítali dobre zadanie a postupovali iba podľa šípok v grafe. Keďže smer šípok je od podriadeného k nadriadenému a zároveň niektoré správy agenta vedia čítať jeho podriadení, práca s časťou grafu môže byť kontraintuitívna.

Rozdiely medzi chlapcami a dievčatami boli štatisticky významné pri odpovediach B, C a D. Správnu odpoveď D vyberali častejšie chlapci, distraktory C a B dievčatá. V najčastejšej chybnéj odpovedi A a v neodpovedaní na úlohu nie je medzi chlapcami a dievčatami štatisticky významný rozdiel. Počty odpovedí dievčat a chlapcov sú uvedené v Grafe 4. Opäť podľa [1, 2] by mala byť úloha dievčenská alebo neutrálna, keďže obsahuje zložitejšiu štruktúru a pracuje sa iba s ňou a informáciami v texte.



Graf 4 - Odpovede v úlohe Tajná správa podľa pohlavia

3.3 Víťaz šachového turnaja

V Bobrove prebieha šachový turnaj. Niektoré šachové partie sa už odohrali. Na obrázku je šípka od víťaza k porazenému. Vidíme, že napríklad Bruno vyhral nad Alicou a Alica vyhrala nad Dávidom. Víťaz turnaja je ten, kto podľa tohto pravidla vyhral nad všetkými.

Pretože nemajú dost času, aby mohol hrať každý s každým, prijali na určenie víťaza nové pravidlo: Ak Bruno vyhral nad Alicou a Alica vyhrala nad Dávidom znamená to, že Bruno vyhral nad Dávidom. Víťaz turnaja je ten, kto podľa tohto pravidla vyhral nad všetkými.

Žiaľ, výsledky, ktoré už mali, nestačili na určenie víťaza turnaja. Rozhodli sa preto odohrať ešte ďalšie štyri partie, ktorých výsledky sú:

Kto je víťazom turnaja?

A) Gréta
B) Igor
C) Cyril
D) Ema

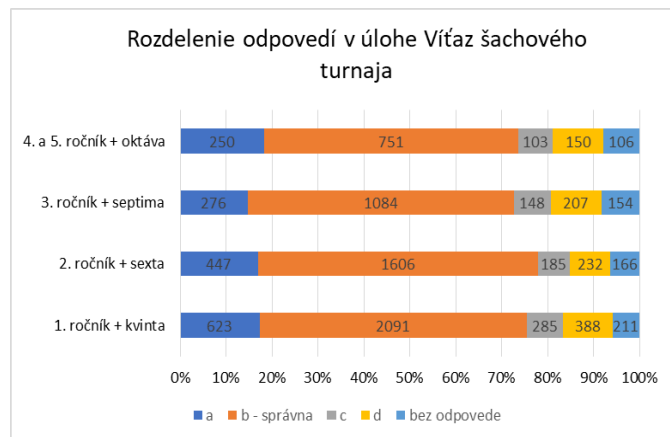
Obrázok 3 - Zadanie úlohy Víťaz šachového turnaja

Úloha Víťaz šachového turnaja, pozri Obrázok 3, má v zadaní orientovaný graf, resp. niekoľko orientovaných grafov, ktoré zachytávajú výsledky šachových partí. Riešiteľ mal doplniť

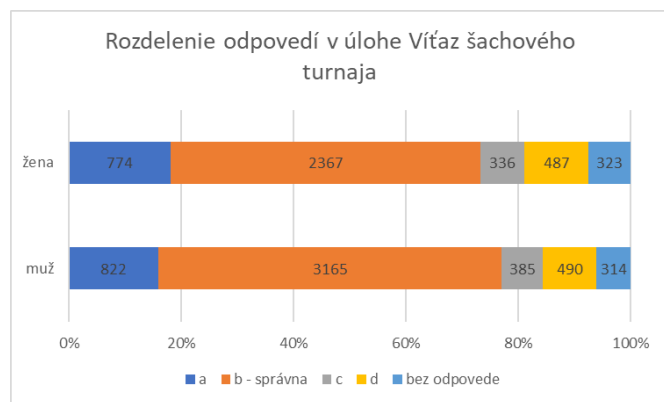
ďalšie údaje do grafu a na ich základe určiť celkového víťaza turnaja.

Úspešnosť riešenia tejto úlohy je vidieť v Grafe 5. Úloha podľa počtu správnych riešení (55 až 60%) patrila skôr k tým ľahším v rámci stredoškolských kategórií, hoci bola zaradená medzi ťažké. Je zaujímavé, že úspešnosť jej riešenia bola najvyššia u druhákov, prváci a tretiaci dosiahli rovnakú úspešnosť a najhoršiu úspešnosť riešenia mali štvrtáci.

Počty odpovedí dievčat a chlapcov sú viditeľné v Grafe 6. Aj túto úlohu riešili chlapci štatisticky významne úspešnejšie ako dievčatá, tie výrazne viac vyberali odpovede A alebo D. V tejto úlohe nie je viditeľný veľmi významný distraktor, čo súvisí aj s vyššou úspešnosťou úlohy.



Graf 5 - Odpovede v úlohe Víťaz šachového turnaja podľa ročníka



Graf 6 - Odpovede v úlohe Víťaz šachového turnaja podľa pohlavia

4 ZÁVER

Všetky tri analyzované úlohy majú spoločné to, že súťažiaci v nich musia porozumieť abstrakcii reálnej situácie pomocou grafu. Na správne riešenie úloh stačilo pozorne prečítať text zadania a porovnať ho s grafom. Samozrejme, počas súťaže sú súťažiaci pod časovým stresom a na riešenie úlohy majú v priemere 3 minúty. Preto sa často stáva, že text úlohy iba rýchlo preletia očami, vyberú si kľúčové slová, či dokonca odhadujú, čo by mohlo byť v zadaní. To sa potvrdilo aj vo výsledkoch, keď

napríklad v úlohe Mestá a diaľnice bol najčastejšie zvolený distraktor a nie správna odpoveď.

Úlohy z kategórie Senior sme zadali po súťaži aj dvom skupinám študentov na FMFI, budúcim učiteľom informatiky a budúcim informatikom. V oboch skupinách sa tiež vyskytli podobné chyby ako v súťaži a pri následnej reflexii študenti uvádzali, že chyby spravili práve z nepozornosti, respektíve predpokladania, čo je úlohou, či čo znamenajú smery šípok v grafe. V oboch prípadoch ide o študentov, ktorí sa už stretli s teóriou grafov, a to, že aj oni robili podobné chyby ako súťažiaci nás môže privádzať k záveru, že v úlohách neboli zvolené grafy najintuitívnejšou či najjednoduchšou reprezentáciou daných situácií (napr. smer šípok v úlohe Tajná správa). V ďalšom výskume sa chceme preto venovať zisťovaniu či a ako ovplyvňuje graf použitý v úlohe celkové výsledky.

Na druhej strane si myslíme, že je dôležité naučiť žiakov to, že iba grafická reprezentácia nestačí, je nevyhnutné ju dodatočne opísať, alebo vysvetliť, pretože málokto grafické znázornenie je jednoznačné či samovysvetľujúce. Z toho dôvodu považujeme za vhodné používať v súťaži aj úlohy, ktoré od súťažiacich vyžadujú prácu s informáciami v grafe, a nedajú sa riešiť iba metódou „pozriem-vidím“.

Vo všetkých analyzovaných úlohách boli chlapci štatisticky významne úspešnejší ako dievčatá. To môže byť ovplyvnené rôznymi faktormi – najčastejšie sa u stredoškôľakov stretávame s poklesom záujmu o informatiku (a s tým spojeným záujmom o súťaž iBobar), tiež sú aj z hodín informatiky viditeľné iné prístupy k riešeniu úloh. Aké konkrétne postupy použili žiaci v týchto úlohách, chceme skúmať v ďalšej fáze výskumu spolu s overením teórie o tom, že žiaci z dôvodu časového stresu menej čítali zadania. Pri skúmaní využijeme aj metódy kvalitatívneho výskumu na menšej skupine stredoškôľakov. Na túto časť výskumu však potrebujeme, aby už boli stredoškôľáci v školách a bolo možné sledovať ich prácu.

POĎAKOVANIE

Chceli by sme sa poďakovať projektu VEGA 1/0602/20, vďaka ktorému mohli byť výsledky nášho výskumu publikované.

BIBLIOGRAFICKÉ ODKAZY

- [1] BUDINSKÁ, L., MAYEROVÁ, K.. Graph Tasks in Bebras Contest: What does it have to do with gender? In *Proceedings of the 6th Computer Science Education Research Conference. ACM*, 2017. s. 83–90.
- [2] BUDINSKÁ, L., MAYEROVÁ, K., WINCZER, M. Gender Differences in Graph Tasks – Do They Exist in High School Bebras Categories Too? In *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*. Springer, 2018. s. 295–306
- [3] ERDŐSNÉ NÉMETH A. Teaching Graphs for Contestants in Lower-Secondary-School-Age. In *Olympiads in Informatics*, 2017. Vol. 11, s. 41–53.
- [4] WING J. M. (2017). Computational thinking's influence on research and education for all. In *Italian Journal of Educational Technology*. 25.2, s. 7–14.
- [5] WIMMER, Gejza. *Štatistické metódy v pedagogike*. Hradec Králové: GAUDEAMUS, 1993. 154 s. ISBN 80-7041-864-8.

Rozvoj informatického myslenia pri hraní videohier

Development of computational thinking in playing video games

Mária Čujdiková

KDMFI, Fakulta matematiky, fyziky a informatiky UK
Mlynská dolina I
842 48 Bratislava
Slovensko
cujdikova1@uniba.sk

ABSTRACT

In our dissertation research, we focus on the development of mathematical thinking in playing video games. We realized that video games also provide opportunities for the development of computational thinking. We explore these opportunities in our paper. We took a closer look at A Plague Tale and Hollow Knight, as we see them as good examples to illustrate this phenomenon. We chose content analysis of video games as the main research strategy, relying on our own gaming experience as well as observations and interviews with two other players. We have found that video games provide rich opportunities for algorithmic thinking, decomposition, generalization, evaluation, and abstraction.

Keywords

Computational thinking, video games, informal learning

ABSTRAKT

V našom dizertačnom výskume sa zaoberáme rozvojom matematického myslenia pri hraní videohier. Pri tom sme si uvedomili, že videohry poskytujú tiež príležitosti aj na rozvoj informatického myslenia. Tieto príležitosti skúmame v našom príspevku. Bližšie sme sa zamerali na preskúmanie hier A Plague Tale a Hollow Knight, pretože ich považujeme za dobré príklady na ilustráciu tohto fenoménu. Ako hlavnú výskumnú stratégiu sme zvolili obsahovú analýzu videohier, pričom sme sa opierali o vlastné herné skúsenosti ako aj o pozorovania a rozhovory s ďalšími dvoma hráčmi. Zistili sme, že videohry poskytujú bohaté príležitosti pre algoritmicke myslenie, dekompozíciu, generalizáciu, hodnotenie a abstrakciu.

Kľúčové slová

Informatické myslenie, videohry, neformálne učenie sa

1 ÚVOD

Videohry neodmysliteľne patria k dnešnej digitálnej dobe. Sú obľúbenou formou zábavy mnohých, najmä mladých ľudí. Viacero výskumov však ukazuje, že nie sú iba zdrojom zábavy, ale že predstavujú aj dobrý prostriedok na rozvoj rôznych kompetencií potrebných pre život v 21. storočí [7, 8, 9]. Hráčov, ktorí ich hrajú, často zapájajú do ťažkých výziev, pri ktorých potrebujú komplexne uvažovať a neustále sa učiť nové veci aktívnym a kritickým spôsobom. Javia sa teda aj ako zaujímavý prostriedok na učenie sa, ktoré je v súlade s modernými teóriami o vzdelávaní. Papert a Gee v [3, 4, 5, 6] poukazujú na to, že princípy učenia sa sú vo videohrách

dokonca tak dobre zvládnuté, že sa pri nich môžeme o učení sa veľa dozvedieť.

V našom dizertačnom projekte z didaktiky matematiky sa zaoberáme skúmaním, aké príležitosti poskytujú videohry v súvislosti s rozvojom matematického myslenia. Zaujímá nás tiež aké aspekty tohto média si v súvislosti s rozvojom matematického myslenia všímajú žiaci strednej školy a ako vnímajú matematiku vo videohrách v porovnaní so školskou matematikou. Videohry predstavujú jednu z možností, ako sa s matematikou a potrebou matematicky myslieť stretnúť v situáciách, kedy to naozaj potrebujeme. Poskytujú teda zaujímavý priestor na objavovanie a osvojovanie si matematických zákonitostí vlastnou skúsenosťou.

Pri skúmaní príležitostí, ktoré majú potenciál rozvíjať matematické myslenie pri hraní videohier sme si čoraz častejšie všímali, že videohry poskytujú tiež množstvo príležitostí aj vzhľadom na rozvoj informatického myslenia. Prišlo nám preto zaujímavé venovať sa aj týmto príležitostiam.

Pojem informatické myslenie (computational thinking) zaviedla Wing v [15], kde o ňom hovorí ako o spôsobe myslenia, aký používajú informatici. Tvrdí však, že tento spôsob myslenia je dôležitý pre všetkých, nielen pre informatikov, a poukazuje, že je tiež súčasťou rôznych ďalších disciplín, či činností v každodennom živote. Podľa jej slov: „*Informatické myslenie spočíva v riešení problémov, navrhovaní systémov a porozumení ľudskému správaniu na základe koncepcií základných pre informatiku.*“ A ďalej zdôrazňuje: „*Výpočtové myslenie je spôsob, ako riešia problémy ľudia. Nesnaží sa prinútiť ľudí, aby premýšľali ako počítače.*“ Hovorí, že počítače sú sami osebe nudné, ale „*My ľudia robíme počítače vzrušujúcimi. Vďaka digitálnym zariadeniam využívame našu chytrosť na riešenie problémov, na ktoré by sme si netrúfli pred digitálnou dobou, a budujeme systémy s funkčnosťou obmedzenou iba našimi predstavami.*“ O tejto novej vzrušujúcej ceste myslenia a premýšľania o myslení, ktorú otvárajú počítače hovoril aj Papert už vo svojom diele *Mindstorms: Children, Computers, and Powerful Ideas* [14].

Jednou z možností, ktoré počítače a iné digitálne zariadenia ponúkajú ako priestor na myslenie sú aj videohry. V našom príspevku, chceme preskúmať, aké myšlienkové procesy, ktoré sú v súlade s informatickým myslením, sú súčasťou ich hrania. Chceme tým tiež prispieť do diskusie o užitočnosti informatického myslenia pri činnostiach, ktorým sa ľudia venujú vo svojom bežnom živote.

Existuje viacero vymedzení, ktoré sa snažia opísať, čo je informatické myslenie a aké komponenty v ňom môžeme identifikovať. My budeme vychádzať z definícií [1] a [2]. Cansu & Cansu v [1] analyzovali, ako informatické myslenie vnímajú vo svojich vedeckých publikáciách rôzni autori a na základe toho dospeli k názoru, že medzi základné komponenty informatického myslenia patrí abstrakcia, dekompozícia, algoritmické myslenie, automatizácia a generalizácia, pričom uvedené pojmy charakterizovali nasledujúcim spôsobom:

- **Abstrakcia** – proces, vďaka ktorému určité detaily ustupujú do pozadia – vždy, keď od nich môžeme “odstúpiť”, čím sa problém stáva jednoduchším.
- **Dekompozícia** – rozdelenie problému na menšie, zrozumiteľnejšie podproblémy.
- **Algoritmické myslenie** – proces vytvárania postupnosti usporiadaných krokov, ktoré vedú k vyriešeniu daného problému.
- **Automatizácia** – konfigurácia vytvorených algoritmov cez počítače a technologické zdroje, aby ich bolo možné efektívne použiť pri iných problémoch.
- **Generalizácia** – identifikovanie vzorov, podobností a súvislostí a ich použitie v inom kontexte.

Selby & Woollard v [2] zahŕňajú do informatického myslenia podobné komponenty, ale vynechávajú automatizáciu a pridávajú **hodnotenie** (evaluation), ktoré vnímajú ako schopnosť hodnotiť procesy z hľadiska efektívnosti a využívania zdrojov a schopnosť rozpoznávať a hodnotiť výsledky. Aj keď Cansu & Cansu nezaraďujú hodnotenie medzi komponenty informatického myslenia, uvádzajú reflexiu ako jednu z techník používaných pri informatickom myslení. Chápu ju v zmysle zručnosti robiť primerané, čestné a zmysluplné hodnotenia v zložitých situáciách.

V našom príspevku sa budeme opierať o definíciu Cansu & Cansu, rozšírenú o hodnotenie v zmysle, aký uvádzajú Selby & Woollard.

2 VÝSKUMNÝ DIZAJN

Na preskúmanie príležitostí na rozvoj informatického myslenia pri hraní videohier sme použili predovšetkým obsahovú analýzu vybraných hier. Tieto hry sme sami hrali, ale tiež sme vychádzali aj zo skúseností dvoch iných hráčov. Obsahová analýza je podľa Krippendorffa [10] výskumná stratégia, pri ktorej čítame obsah vybraného produktu spôsobom, ktorý nás vedie k odpovediam na naše výskumné otázky. Podľa Schmierbacha a Mallieta [11, 12] je obsahová analýza vhodným nástrojom aj na skúmanie videohier. Avšak ako zdôrazňujú, pri analýze videohier treba myslieť na špecifickosť tohto média. Na rozdiel od iných produktov, ktoré sú predmetom obsahovej analýzy (napr. písaný text, film, video,...), to, ako sa vyvíja daná videohra, často závisí od konkrétneho hráča. Výskumník, ktorý hrá videohru, je teda zároveň aj spoluvorcom analyzovaného obsahu.

Pre analýzu sme zvolili dve hry, a to A Plague Tale: Innocence a Hollow Knight. Išlo o zámerný výber, pretože tieto hry sa javili ako bohatý zdroj príležitostí pre postupy, chápané ako súčasť informatického myslenia, a teda poskytovali podnetné prostredie pre skúmaný fenomén.

A Plague Tale: Innocence (Obr. 1) je akčná adventúra. Jej príbeh sa odohráva vo Francúzsku počas storočnej vojny. Mladá Amica spolu s jej malým bratom Hugom sa snažia utiecť pred inkvizíciou.

Všade vládne mor a krajina je zaplavená nebezpečnými zmutovanými krysmi.

V hre Hollow Knight (Obr. 2) sa zasa hráč ocitne v roli osamelého bojovníka, ktorý preskúmava tajuplnú podzemnú ríšu. Táto hra je metroidvania, čo znamená, že už od začiatku má hráč prístupnú celú mapu, ale najskôr má len veľmi obmedzené schopnosti a mnohé miesta, z ktorých niektoré síce aj vidí, sú preňho nedostupné. Postupne, ako získava nové schopnosti, sa môže vrátiť na miesta, kam sa predtým dostať nevedel.



Obrázok 1: Situácia z A Plague Tale. Amica sa skrýva za ohradou. Pred ňou sú krysy, nepriateľ a lampa (zakružkované). Hráč si potrebuje situáciu rozdeliť na niekoľko menších častí (podproblémov) a vymyslieť postupnosť krokov, ktorá mu pomôže dostať sa ďalej.



Obrázok 2: Hollow Knight – súboj s bossom. Boss má niekoľko taktík boja, ktoré používa a ktoré si hráč potrebuje všimnúť a vymyslieť postup, ako na ne správne a v čas zareagovať.

3 METÓDA

Zvolili sme kvalitatívny prístup, keďže sme nechceli overovať vopred stanovené hypotézy, ale chceli sme skúmaný fenomén čo najviac pochopiť a následne podľa toho popísať.

Pri skúmaní hier sme vychádzali z vlastného hrania a z hrania dvoch participantov, Martina a Adama. Týchto participantov sme zvolili zámerným výberom, pretože sa javili ako vhodní z hľadiska nášho skúmaného fenoménu a zároveň boli ochotní sa do výskumu zapojiť. Martin má 32 rokov a pracuje ako grafik. Videohry hrá už od detstva a má množstvo skúseností s hrami rôznych žánrov. Martin hral obe vybrané hry. Pri hraní častí týchto hier sme ho pozorovali a rozprávali sa s nim o jeho postupe. Ďalšie rozhovory

o uvažovaní pri týchto hrách sme spolu viedli aj po ich dohraní. Adam má 26 rokov a je doktorand venujúci sa didaktike matematiky. Aj jeho skúsenosti s hraním videohier sú dlhoročné. Adam hral z vybraných hier len Hollow Knight. Potom, ako mal dohranú celú hru, sme s ním viedli rozhovor na tému premýšľania a postupov pri jej hraní. Rozhovory, ktoré sme s účastníkmi uskutočnili, boli pološtruktúrované.

Pri analýze situácii z hier a dát získaných z pozorovaní a rozhovorov sme použili kvalitatívny prístup. Zamerali sme sa na skúmanie jednotlivých výziev vo vybraných hrách a toho, ako v nich hráč využíva postupy, ktoré sú chápané ako súčasť informatického myslenia. Náš prístup mal deduktívny charakter [13]. Kategórie, na preskúmanie ktorých sme sa zamerali, vychádzali z vymedzení komponentov informatického myslenia podľa [1, 2], ako uvádzame v úvode.

4 VÝSLEDKY

V tejto časti zhrnieme, aké príležitosti na rozvoj informatického myslenia poskytujú videohry. Hoci sa pri našej analýze zaoberáme samostatne jednotlivými komponentmi informatického myslenia, tieto sa v skutočnosti často prelínajú a nie je možné ich pri hraní od seba odlúčiť.

4.1 Algoritmické myslenie

Vo videohrách sa často nachádzajú náročné výzvy, pri ktorých je potrebné naplánovať postupnosť krokov, ktoré vedú k ich zdolaniu. Presne tak je to aj pri oboch uvedených hrách. Mnohé problémy sa hráčovi nepodarí vyriešiť na prvýkrát. Často sa stane, že má myšlienku, vyskúša ju, ale nevyjde mu to. Skúsi teda vyhodnotiť, prečo sa to stalo a svoj postup dodadí. Niektoré situácie si vyžadujú aj niekoľko kôl takéhoto ladenia. Zväčša neexistuje iba jedna správna postupnosť krokov, ktorou sa dá danú situáciu vyriešiť, ale viacero rôznych postupov môže viesť k rovnako úspešnému riešeniu.

Dôležitou súčasťou hier, ktorá poskytuje priestor na rozvoj algoritmického myslenia je správanie nepriateľov a iných postáv, ktoré neovláda hráč. V oboch uvedených hrách existujú viaceré typy nepriateľov. Hráč potrebuje na úspešný boj s nimi pochopiť na základe akého algoritmu fungujú - čo a kedy robia, ako sa pohybujú, ako a kedy útočia. Podľa toho môže vymyslieť svoj postup, ako s nimi bojovať. Potrebuje prísť na to, v ktorej časti útoku sa má uhnúť a ako a v ktorej časti môže útočiť a kam má útok cieľiť. Tiež potrebuje prísť na to, aké pravidlá platia pre skrývanie sa pred daným typom nepriateľov a ako môže bezpečne utiecť, lebo útek je niekedy najlepšia obrana. Ešte náročnejší na pochopenie ich správania a teda aj na vymyslenie postupov pri boji s nimi sú bossovia, pretože ich správanie je komplikovanejšie ako pri bežných nepriateľoch.

Premyslieť si ako bojovať s danými nepriateľmi alebo bossmi je výhodné takmer pri každej hre, ale pri niektorých stačí namiesto toho aj len byť dostatočne rýchly a ostražitý. Hry Hollow Knight a A Plague Tale sú príkladom hier, v ktorých je premyslenie si postupu boja nevyhnutné.

Naopak ako pri nepriateľoch, pri postavách, ktoré sú priateľské hráč zisťuje, akým spôsobom mu vedľa tieto postavy pomôcť. Príkladom takýchto postav sú obchodníci. Nákup u obchodníkov môže byť tiež jedným z krokov stratégie riešenia určitej situácie.

Ďalšou príležitosťou na rozvoj algoritmického myslenia vo videohrách je prostredie. Hráč vlastným preskúmaním spoznáva pravidlá, ktoré v ňom platia - čo v prostredí mu pomáha,

čo škodí a akým spôsobom. Počas hrania hry potrebuje neustále vyhodnocovať dané situácie a vopred si premyslieť postupnosť krokov, vďaka ktorej dokáže prežiť a postúpiť v hre ďalej.

Hráč môže do svojho postupu tiež zapojiť rôzne schopnosti a predmety, ktoré už vlastní alebo dokáže získať.

4.2 Dekompozícia

Videohry obsahujú množstvo problémových situácií, ktoré je potrebné postupne riešiť. V niektorých hrách je vopred určené, ako tieto problémy nasledujú za sebou, (napr. A Plague Tale), iné (napr. Hollow Knight) umožňujú v danej chvíli riešiť viacero problémov a to, ako si ich hráč rozloží a rozplánuje, je na ňom. Pri Hollow Knight sa hráč snaží plniť úlohy tak, aby postupne získaval nové schopnosti, predmety a medailóny s vylepšeniami, ktoré mu umožňujú následne plniť ďalšie úlohy, ktoré by bez nich zdolať nedokázal.

Dekompozícia je často potrebná aj pri jednotlivých situáciách. Mnohé situácie, ktoré vyvstanú v hrách, sú silne komplexné a na ich úspešné zvládnutie si ich hráč potrebuje rozdeliť na viacero menších častí - podproblémov. Každý podproblém následne ešte ďalej delí a plánuje postupnosť krokov, ktoré vedú k jeho riešeniu. Dôležité je tiež, aby vymyslel, v akom poradí bude dané podproblémy riešiť, aby sa v konečnom dôsledku prepracoval k vyriešeniu celej situácie.

Typickým problémom vo videohrách, ktorý tiež vyžaduje dekompozíciu je, že na mnohých miestach sa nachádza viacero nepriateľov, pri ktorých sa oplatí nájsť spôsob, ako ich rozdeliť a bojovať s nimi postupne, buď po jednom alebo po menších skupinách. Takéto situácie často nachádzame aj v oboch uvedených hrách.

4.3 Hodnotenie

Pri hraní videohier je potrebné neustále vyhodnocovať rôzne aspekty. Celkovo, ako sme už spomenuli v časti 4.1, hráč neustále skúša rôzne stratégie a hodnotí, či sú úspešné. Vždy, keď vidí, že jeho myšlienka nefunguje, snaží sa analyzovať prečo a vymyslieť, čo by mohol urobiť inak, aby danú situáciu vyriešil.

Hráč tiež často vyhodnocuje, či sa do riešenia určitého problému v danom okamihu vôbec púšťať. Napr. pri Hollow Knight je preňho užitočné zvážiť, či má na zvládnutie určitého problému dostatočné zdravie, či už má schopnosti, ktoré sú k riešeniu tohto problému nevyhnutné a pod. Tiež, keď získa nejakú novú schopnosť, zvažuje, kam sa má vrátiť, aby ju mohol použiť.

Ďalej hodnotí, aké predmety alebo vylepšenia má použiť v danej situácii. Napr. pri A Plague Tale potrebuje vyhodnotiť, aký predmet sa mu najviac oplatí vyrobiť zo surovín, ktoré má k dispozícii. Musí to premyslieť, pretože surovín je vždy obmedzený počet, a ak nejakú použije na výrobu určitého predmetu, môže mu chýbať pri inom, ktorý by sa mu mohol v danej chvíli hodiť viac. Je preňho teda výhodné zvoliť optimálne využitie týchto surovín. Podobne v Hollow Knight si hráč vyberá medailóny, ktoré mu dávajú určitú výhodu, pričom ich môže mať súčasne pripnutých a teda aktívnych iba určitý počet. Je preto tiež užitočné tento výber optimalizovať a zvoliť tie medailóny, ktoré sú pre danú situáciu najvýhodnejšie.

Pri Hollow Knight, rovnako ako pri iných hrách, kde je možné liečiť sa, hráč tiež neustále hodnotí, či má dostatočné zdravie, alebo ho potrebuje doplniť a ak vyhodnotí, že ho potrebuje doplniť, zvažuje, kedy a kde to môže spraviť bezpečne.

4.4 Generalizácia

Keď sa hráč v hre raz naučí nový postup alebo objaví novú súvislosť, zväčša to znova použije aj v ďalších situáciách. Ak napr.

pochopí, aké pravidlo platí pre boj s určitým typom nepriateľa, použije ho aj inde, keď daného nepriateľa stretne. Rovnako to platí pri podobných nepriateľoch. Keď niečo fungovalo pri jednom, dá sa z toho vychádzať pri hľadaní stratégie pri ďalšom, ktorý má s predošlým v niečom podobné správanie. Podobne je to s pochopením fungovania prostredia. Hráč je často vystavený situácii, kedy má zmysel použiť už naučené pravidlo v nových kontextoch. Neustále teda dochádza k zovšeobecňovaniu a aplikácii predchádzajúcich skúseností.

4.5 Abstrakcia

Pri riešení jednotlivých situácií je pre hráča výhodné zamerať sa iba na to, čo je podstatné práve pre ich riešenie. Z celého komplexného prostredia hry si môže všimnúť len tie prvky, ktoré mu práve pomáhajú, alebo naopak môžu uškodiť a ostatné nemusí brať v danej chvíli do úvahy.

5 ZÁVER

V našom dizertačnom projekte sa venujeme skúmaniu, aké príležitosti na rozvoj matematického myslenia ponúkajú videohry. V priebehu výskumu sme zaregistrovali, že vo videohrách nájdeme tiež množstvo príležitostí na rozvoj informatického myslenia. Rozhodli sme sa preto preskúmať aj tieto príležitosti.

Na základe analýzy vybraných hier a postupov, ktoré pri nich volia hráči, sme dospeli k záveru, že videohry poskytujú príležitosti pre algoritmické myslenie, dekompozíciu, generalizáciu, hodnotenie aj abstrakciu. Tieto komponenty informatického myslenia sú pri hraní často prepojené a nie je ich možné od seba oddeliť. V hrách sa nachádza množstvo zložitých situácií, pri ktorých je vhodné naplánovať si postupnosť krokov, ktoré vedú k ich riešeniu. Zväčša je pri tom dosť užitočné rozdeliť si celý problém na niekoľko menších podproblémov. Je tiež potrebné vedieť rozlíšiť, čo je pre danú situáciu dôležité a naopak, čo je možné v danej chvíli zanedbať. Pri každej situácii je potrebné zhodnotiť množstvo aspektov, ktoré s ňou súvisia a ktoré môžu ovplyvniť priebeh riešenia.

Za najväčšiu limitáciu tohto výskumu považujeme malý počet zapojených hráčov. Keďže rôzni hráči hrajú hry rôznym spôsobom, pri väčšom počte participantov, by sa boli mohli vynoriť aj iné prístupy a tým pádom by sa možno boli objavili aj ďalšie zistenia. Určite by bolo tiež zaujímavé zapojiť žiakov strednej školy a preskúmať, aké postupy používajú oni. Kvôli pandemickej situácii sme však prístup do školy nemali.

V našej analýze sme sa sústredili predovšetkým na dve hry – Hollow Knight a A Plague Tale: Innocence, avšak nemyslíme si, že tieto hry sú z hľadiska opísaných príležitostí pre rozvoj informatického myslenia jedinečné, hoci ich považujeme za dobré príklady. Podobný potenciál vo väčšej či menšej miere nájdeme aj v mnohých iných videohrách rôznych herných žánrov. Ako zhodnotil aj Martin napr. rozpoznávanie ako fungujú nepriatelia a hľadanie postupu, ako s nimi bojovať je súčasťou množstva hier. Podobne aj ostatné princípy nájdeme tiež v iných hrách. Väčšina hier vyžaduje v určitej miere vhodne používať obmedzené zdroje a vedie hráčov k zovšeobecneniu už naučeného. Rovnako mnoho hier ponúka komplexné problémy, ktoré je vhodné rozdeliť na menšie časti a tiež vyžaduje určitý stupeň abstrakcie pri riešení problémov. Samozrejme podobne je to aj pri iných hrách, nielen pri videohrách, a tiež pri mnohých iných činnostiach.

Určite tiež v iných typoch videohier by sme mohli nájsť aj ďalšie zaujímavé princípy, ktoré sú v súlade s informatickým myslením. Ako veľmi zaujímavé z hľadiska informatického myslenia sa javia

napr. hry, ktoré umožňujú vopred naplánovať postupnosť krokov, a potom hru vyzvať, aby tieto kroky vykonala. Dá sa povedať, že tak hráč určitým spôsobom programuje budúce správanie postavy. Takáto možnosť je napr. pri hre Star Wars: Knights of the Old Republic. Mnohé videohry sú tiež zaujímavé z hľadiska optimalizácie. Aj keď aj pri A Plague Tale a Hollow Knight je pre hráča vhodné premýšľať nad optimálnymi voľbami, sú hry, kde je to potrebné vo väčšej miere. Napr. pri mnohých RPG (hrách na hrdinov), v ktorých si hráč vytvára postavu s tým, že má obmedzené množstvo bodov, ktoré môže investovať do rôznych vlastností a tiež obmedzené miesto v inventári. Určite aj tieto hry a informatické aspekty v nich by mohli stáť za hlbšie preskúmanie. Myslíme si, že videohry v spojení s informatickým myslením poskytujú ešte veľký priestor pre ďalšie bádanie a zaslúžili by si ďalší výskum.

POĎAKOVANIE

Chcela by som sa poďakovať profesorovi Ivanovi Kalašovi za jeho cenné rady, nápady a pripomienky. Ďalej by som sa chcela poďakovať Martinovi Bolhovi a Adamovi Jakubičkovovi za ich ochotu zdieľať svoje myšlienky, postupy a názory. Ďakujeme tiež agentúre VEGA, ktorá prostredníctvom projektu 1/0602/20 prispela k tomu, že výsledky nášho výskumu môžeme publikovať na konferencii Didinfo 2021.

BIBLIOGRAFICKÉ ODKAZY

- [1] CANSU, F. K., CANSU, S. K. An Overview of Computational, Thinking. In *International Journal of Computer Science Education in Schools*, 2019, vol. 3, no. 1, p. 17-30. DOI=<https://doi.org/10.21585/ijcses.v3i1.53>
- [2] SELBY, C. C., WOOLLARD, J. Computational thinking: The developing definition. In *Presented at the 18th annual conference on innovation and Technology in Computer Science Education*, Canterbury. 2013
- [3] GEE, J. P. *What videogames have to teach us about learning and literacy*. New York: Palgrave Macmillan. 2003. 232 s. ISBN 1-4039-6538-2
- [4] GEE, J. P. *Good videogames and good learning*. Madison: University of Wisconsin, 2007. Dostupné tiež https://academiccolab.org/resources/documents/Good_Learning.pdf
- [5] PAPERT, S. The Parent Trap. In *Time Magazine*. 1995. Dostupné tiež http://papert.org/articles/parent_trap.html,
- [6] PAPERT, S. Does Easy Do It? Children, Games, and Learning. In *Game Developer magazine*. 1998. Dostupné tiež <http://www.papert.org/articles/Doeseasydoit.html>
- [7] BEDIU, B. a kol. Meta-Analysis of Action Video Game Impact on Perceptual, Attentional, and Cognitive Skills. In *Psychological Bulletin*. 2017, vol. 144, no. 1, p. 77-110. DOI=<http://dx.doi.org/10.1037/bul0000130>
- [8] WHITLOCK, L. A., MCLAUGHLIN, A. C., ALLAIRE, J. C. Individual differences in response to cognitive training: Using a multi-modal, attentionally demanding game-based intervention for older adults. In *Computers in Human Behavior*. 2012, vol. 28, no. 4, p. 1091-1096. ISSN 0747-5632. DOI=<https://doi.org/10.1016/j.chb.2012.01.012>

- [9] MENGEL, F. Computer Games and Prosocial Behaviour. In *PLoS ONE*. 2014, vol. 9, no. 4. DOI=<https://doi.org/10.1371/journal.pone.0094099>
- [10] KRIPPENDORFF, K. *Content analysis: An introduction to its methodology*. 2nd ed. Thousand Oaks, CA: Sage Publications. 2004. 413 p. ISBN 0761915451, 9780761915454
- [11] SCHMIERBACH, M. Content Analysis of Video Games: Challenges and Potential Solutions, In *Communication Methods and Measures*. 2009, vol. 3, no. 3, p. 147-172. DOI=10.1080/19312450802458950
- [12] MALLIET, S. Adapting the principles of ludology to the method of video game content analysis. In *Game Studies*. 2007, vol. 7, no. 1, p. 1-18
- [13] MAYRING, P. *Qualitative content analysis: theoretical foundation, basic procedures and software solution*. Klagenfurt, Austria. 2014
- [14] PAPERT, S. *Mindstorms: Children, Computers, and Powerful Ideas*. New York: Basic Books, Inc. 1980. p. 242. ISBN: 0-465-04627-4
- [15] WING, JEANNETTE M. Computational Thinking. In: *Communications of ACM*. 2006. vol. 49, no. 3, p. 33–35. DOI=<http://doi.acm.org/10.1145/1118178.1118215>

Vztah učitelů mateřských škol k využívání digitálních technologií při přímé pedagogické činnosti

Attitudes of kindergarten teachers about use of digital technologies in direct pedagogical activities

Václav Dobiáš
Katedra informatiky PF JČU
Branišovská 1645/31a
370 05 České Budějovice
Česká republika
dobias@pf.jcu.cz

ABSTRACT

The paper tries to answer the research question: What are attitudes of kindergarten teachers about use of digital technologies in direct pedagogical activities? And what factors influence these attitudes? The research was carried out by a qualitative analysis of the seminar works of future kindergarten teachers. In the seminar papers, based on interviews, students sought the opinions of teachers on the use of digital technologies in direct pedagogical activities.

Kindergarten teachers observe the specific negative consequences of the overuse of digital technologies by children in the family environment. This leads some teachers to reject the use of digital technologies in direct pedagogical work. However, most kindergarten teachers understand digital technology in direct pedagogical activities only as a diversification or complement. The identified factors influencing teachers' attitudes to the use of digital technologies are primarily the age of teachers which are often associated with low digital literacy, which doesn't allow teachers to see the possibilities of using digital technologies in teaching.

Keywords

- Kindergarten
- Digital technology
- Kindergarten teachers

ABSTRAKT

Příspěvek se snaží odpovědět na otázky: Jaké názory zastávají učitelé mateřských škol na využití digitálních technologií při přímé pedagogické činnosti? A jaké faktory tyto názory ovlivňují? Výzkum byl realizován kvalitativní analýzou seminárních prací studentů kombinovaného studia učitelství v mateřských školách. V seminárních pracích studenti na základě rozhovorů zjišťovali názory učitelů na využívání digitálních technologií při přímé pedagogické činnosti.

Učitelé mateřských škol pozorují konkrétní negativní důsledky nadužívání digitálních technologií dětmi v rodinném prostředí. Což vede část učitelů k odmítavému postoji na využívání digitálních technologií při přímé pedagogické práci. Většina učitelů mateřských škol chápe digitální technologie při přímé pedagogické činnosti pouze jako zpestření či doplněk. Identifikovanými faktory ovlivňujícími vztah učitelů k využívání digitálních technologií je především věk učitelů a s ním často spojená jejich nízká digitální gramotnost, která mimo jiné neumožňuje učitelům nahlédnout možnosti využití digitálních technologií při výuce.

Klíčová slova

- Mateřská škola
- Digitální technologie
- Učitelé mateřských škol.

1 ÚVOD

Tento text volně navazuje na článek publikovaný v minulém ročníku konference didinfo [1]. Podle tohoto příspěvku o vybavenosti a způsobu využití digitálních technologií, rozhoduje především přesvědčení učitelů a ředitelů mateřských škol o smysluplnosti využití digitálních technologií. Dalším identifikovaným faktorem ovlivňujícím způsob využití digitálních zařízení v mateřských školách byla úroveň digitální gramotnosti učitelů mateřských škol. Zaměřili jsme se proto na názory učitelů a ředitelů mateřských škol na využití digitálních technologií v mateřských školách při přímé pedagogické činnosti. K tomuto účelu jsme zformulovali výzkumné otázky:

- Jaké názory zastávají učitelé mateřských škol na využití digitálních technologií při přímé pedagogické činnosti?
- Jaké faktory ovlivňují vztah učitelů k využití digitálních technologií v mateřských školách?

Jak bylo zmíněno výše, tento příspěvek navazuje na rok starý příspěvek. Na základě porovnávání dat z minulého a tohoto ročníku, jsme se rozhodli proto dále identifikovat změny či trendy v oblasti využívání digitálních technologií v mateřských školách při přímé pedagogické činnosti.

2 POSTUP VÝZKUMU

Výzkum byl realizován analýzou seminárních prací studentů kombinovaného studia učitelství v mateřských školách. V seminární práci si studenti měli vybrat konkrétní mateřskou školu, charakterizovat ji, popsat vybavenost a způsob práce s digitálními technologiemi. Studenti ale především zjišťovali názory učitelů v konkrétní mateřské škole. Měli se zaměřit na následující oblasti:

- Jaké pedagogické přínosy očekávají učitelé MŠ od digitálních technologií.
- Spatřovaná rizika ohledně využívání digitálních technologií v MŠ.
- Co by učitelům pomohlo ve využívání digitálních technologií v MŠ.
- Schopnosti práce učitelů s digitálními technologiemi.
- Přístup vedení MŠ k využití digitálních technologií v MŠ.

- Charakterizovat věk učitelů.

Výrazná většina studentů realizovala rozhovory k seminární práci v mateřských školách, ve kterých pracují nebo vykonávají dlouhodobou pedagogickou praxi. Velmi dobře tak znali popisované prostředí a osobně se znali s učiteli, se kterými realizovali rozhovory. Způsob a forma realizace rozhovorů byly ponechány na studentech. Rozhovory realizovali studenti v rámci předmětu „Digitální technologie v předškolním vzdělávání“. Celkem bylo analyzováno 40 seminárních prací popisujících 42 různých mateřských škol. Dohromady tak tato práce mapuje názory odhadem 120 učitelů mateřských škol. Jako problém takového způsobu sběru dat se ukázala jistá nejednotnost v popisu reality mateřské školy mezi jednotlivými seminárními pracemi. V některých seminárních pracích byl například uveden věk učitelů mateřských škol ve věkovém rozmezí. Z tohoto důvodu nejsme schopni uvádět přesné kvantitativní informace v závislosti na věku učitelů.

Většina mateřských škol zúčastněných na výzkumu byla z Jihočeského kraje. Ve výzkumu byly ale zastoupeny i čtyři mateřské školy z Prahy, tři středočeské mateřské školy, dvě západočeské mateřské školy a jedna jihomoravská mateřská škola. Seminární práce byly zpracovávány od října 2020 do prosince 2020.

Samotné seminární práce byly analyzovány kvalitativně-kvantitativní formou. Kvalitativní analýza byla realizována v programu Atlas.ti [2]. Pro získání přehledu nad daty byla data kódována otevřeným kódováním podle Strauss a Corbinové [3] a Švařička a Šedřové [4]. Jednotlivé kódy pak byly mezi sebou porovnávány a seskupovány z různých úhlů pohledu, jako je například: věk učitelů, klasická x alternativní mateřská škola, velikost obce, pod kterou spadá mateřská škola apod. Při samotném psaní byly jednotlivé texty spadající pod konkrétní kód znovu kompletně přečteny, byly hledány souvislosti, naopak rozdíly, či extrémní situace v rámci jednotlivých kódů. Data kvantitativního charakteru byla zaznamenávána do tabulkového procesoru. Následně byly z těchto dat tvořeny kontingenční tabulky. Získaná data byla dále porovnávána s rok starými daty z článku [1]. Studenti byli zahrnuti do procesu evaluace textu, kdy jim byl zaslán hotový text s výzvou k připomínkám. Žádné připomínky ale nebyly zaznamenány.

3 VÝSLEDKY

3.1 Změny ve vybavenosti digitálních technologií v mateřských školách

Na základě porovnání dat z minulého ročníku [1] a současných dat, uvádíme změny ve využívání digitálních technologií v mateřských školách při přímé pedagogické činnosti, čímž se snažíme identifikovat trendy v této oblasti.

Oproti konci roku 2019 se pomalu zvyšuje penetrace digitální techniky v mateřských školách. Tabulka 1 zachycuje procentní podíl mateřských škol, které v daném roce disponovaly alespoň jedním kusem této digitální techniky.

Tabulka 1: Penetrace digitální techniky v mateřských školách (v procentech)

	2019	2020
Interaktivní tabule	54,8	59,5
Fotoaparát	28,6	28,6
Smartphone	21,4	26,2
Tablet	21,4	26,2
Bee-bot	9,5	11,9
Albi tužka	4,8	7,1

3.1.1 Interaktivní tabule

V roce 2019 celkem třetina interaktivních tabulí instalovaných v mateřských školách byla využívána pouze jako projektor, anebo nebyla využívána vůbec. Oproti tomu v roce 2020 jsme zaznamenali mnohem menší počty špatně nebo vůbec nepoužívaných interaktivních tabulí. Můžeme zde pozorovat trend postupného zvykání učitelů mateřských škol na interaktivní tabule při přímé pedagogické činnosti. Díky tomuto trendu ale někteří učitelé začínají pociťovat nedostatečné možnosti přístupu k interaktivní tabuli jak například: „Na jejich MŠ se nachází pouze jedna interaktivní tabule a to je na 8 třídíní školku opravu málo. On sám by ji během výuky používal více, kdyby k ní měl snazší přístup. Učitelé tak preferují variantu instalace interaktivní tabule do své třídy. Zároveň ale většina učitelů, kteří mají přístup k interaktivní tabuli, s ní pracují maximálně jedenkrát týdně.“

3.1.2 Fotoaparát

Většina učitelů využívá fotoaparátů pouze k dokumentaci aktivit v mateřské škole. Nejčastěji to bývá za účelem prezentace fotografií rodičům, případně si v některých situacích pomocí fotografií připomenou proběhlé aktivity. Mnohdy jsou jednoúčelové fotoaparáty nahrazovány fotoaparáty v mobilních telefonech, případně tabletech. V některých případech učitelé využívají soukromých zařízení k pořizování fotografií. Jako například: „Učitelky využívají příležitostně vlastní mobily pro fotodokumentaci aktivit dětí, fotografie pak stahují do pracovního PC. Ve školce je k dispozici také digitální fotoaparát, ale protože se opět jedná o starší typ, je v poslední době využíván minimálně.“ V datech jsme nezaznamenali žádné dětské fotoaparáty.

3.1.3 Mobilní telefon

S nástupem chytrých telefonů čím dál více učitelů mateřských škol začíná využívat mobilní telefon nejen ke komunikaci s rodiči, ale i k dalším činnostem. Nejčastěji k fotodokumentaci aktivit v mateřských školách. Pouze několik učitelů ale uvedlo, že mobilní telefony využívají při přímé pedagogické činnosti. Pozitivním příkladem může být: „Například když jdeme ven, vidíme neznámou rostlinu obsypanou květy, netrhám ji, ale vyfotím a užiji aplikaci, která mi určí druh. Poté přijdeme do školky a již známou květinu si vyhledáme v knize a přečteme si o ní něco poučného.“

3.1.4 Tablety

Ač se to z tabulky 1 nezdá, domníváme se, že v datech je patrný trend cíleného nákupu více tabletů (cca od 8 do 20 ks) do mateřských škol a jejich následného využití při přímé pedagogické činnosti. Nejčastěji zde učitelé využívají různé výukové programy. Někteří učitelé vnímají tablety, jako možnou náhradu za interaktivní tabule například: „Tablet by mohl částečně nahrazovat interaktivní tabuli. Děti na tabletu mohou reagovat přímým dotykem ruky, nalézat tak řešení pokus-omyl online a zároveň poskytuje názornost.“

3.1.5 Robotické hračky

V roce 2019 jsme nezaznamenali žádnou mateřskou školu, ve které by docházelo k záměrnému rozvoji inforatického myšlení anebo algoritmizace pomocí robotických včelek tak, jak to popisují například Maněnová a Pekárková [6]. V tomto roce 2020 už většina mateřských škol vybavených robotickými včelkami (3 z 5) je už používala k danému účelu. Nemluvíli ovšem o rozvoji inforatického myšlení. Nejkonkrétnější popis pedagogických cílů poskytla paní učitelka: „Včelka učí děti plánovat, uvažovat, počítat a soustředit se. Děti musí používat logické myšlení a postupovat strategicky, učí se z chyb a omylů.“ Ostatní nezmiňovali nebo popisovali sledované výukové cíle pomocí robotických hraček velmi obecně.

3.2 Názory učitelů na využívání digitálních technologií v mateřských

Učitelé mateřských škol se znepokojením sledují negativní vlivy nadměrného využívání digitálních technologií u dětí v rodinném prostředí. Toto znepokojení dokumentují například citace: „Není žádnou novinkou, že většina dětí čas trávený v rodině, doopravdy tráví spíše v televizi, tabletu či na počítači.“ „V dnešní době se děti naučí dříve používat tablet nežli mluvit.“ nebo případně „Často od nich slyšíme nelichotivá slova, jejich význam děti ani neznají. Děti o těchto výrazech ani nepřemýšlejí a s nadšením vyprávějí, že sledují na tabletu „youtubery“ a ti takto hovoří.“

Velké množství času stráveného u digitálních technologií má podle učitelů mateřských škol za následek omezení sociálních kontaktů dětí, které má následně vliv na jejich komunikační schopnosti a socializaci. Zároveň množství času stráveného u digitálních zařízení omezuje čas dětí strávený pohybovými aktivitami. Četní učitelé zdůrazňují, že jde především o pasivní aktivitu, která nedokáže děti rozvinout tak, jako aktivní hra.

Mnozí učitelé mají pocit, že současné děti mají menší znalosti o reálném světě, což přičítají nadměrnému využívání digitálních technologií v rodinném prostředí. Například: „Už ani děti, které bydlí na vesnici, nepoznají kaštan, žalud, borovou šišku atd., ale hry na tabletech znají všechny“ Pomocí digitálních technologií děti velmi často konzumují nevhodný obsah, který u nich může posilovat agresivní chování, či způsobovat zkrácené představy o realitě. Tento obsah je koncipován tak, aby děti upoutal či zaujal, což může u dětí vést k nezájmu o reálný svět, roztěkanosti. Mnozí učitelé mluví o závislosti. Důsledkem může být: „neochota dětí pracovat jinak, než s používáním digitálních pomůcek či hraček. Například „dítě „čte“ rádo pouze s ALBI tužkou, ale o jiné knihy zájem nemá“, nebo: „při pracovních, výtvarných činnostech či rozvoji rozumových vlastností spolupracuje pouze, když je u toho využívaná interaktivní tabule a jinak činnosti odmítá.“

Mnozí učitelé, převážně starší učitelé, s vědomím negativních následků přílišného využívání digitálních technologií v rodinném prostředí odmítají digitální technologie při přímé pedagogické činnosti využívat.

Převažují ale učitelé, kteří si realisticky uvědomují, že děti se od brzkého mládí setkávají s digitálními technologiemi a toto setkání má mnoho rizik (např. závislost). Proto je třeba cíleně budovat vztah dětí k nim, například: „Budeme-li pravidelně a řízeně zapojovat digitální technologie do výuky, můžeme korigovat to, jaký vztah si k nim děti vyvinou – zda pro ně budou dobrým sluhou, ne špatným pánem. Pokud děti necháváme se s digitálním světem seznamovat samovolně, vystavujeme je mnoha rizikům.“ Učitelé se

mnohdy snaží ukázat „že digitální technologie nejsou jen pro zábavu, ale jsou především pracovním, nebo kreativním nástrojem.

Tito učitelé na digitálních technologiích při výuce oceňují motivační prvek technologií. Tento motivační prvek je tak silný, že někteří učitelé zmiňují, že je složité zorganizovat výuku s digitálními technologiemi, protože všechny děti si chtějí naráz vyzkoušet danou technologii. Čekání než se na děti dostane řada, tak může u dětí trénovat trpělivost a učení se dodržování pravidel. Někteří učitelé si všimají, že děti jsou skrze technologie ochotny lépe spolupracovat. Což ale mnohdy přičítají častému využívání digitálních technologií v domácím prostředí.

Učitelé velmi často oceňují digitální technologie pro jejich názornost a jako zdroj informací. Vyžívají jich k usnadnění výkladu a demonstraci probíraných témat. Jako například: „vývoj motýla se slovy popisuje těžce“ nebo „Děti si vymyslely aktivitu skládání velice těžkého origami parníku. Skládat umím, ale bohužel tak velkému počtu předškolních dětí to nelze jinak ukázat, než po sekvencích na velké obrazovce pustit a postupně krok po kroku složit.“ případně děti „mohou natočené video shlédnout vícekrát a povídat si o něm.“

Zároveň učitelé upozorňují, že učení pomocí digitálních technologií je zprostředkované učení. Děti si nemohou na zkoumaný předmět sáhnout, čichnout k němu, potéžkat jej či hmatem cítit strukturu povrchu. „dítě by mělo poznávat požitkem. DT tento požitek ochuzují.“

Mnoho učitelů mateřských škol se v souvislosti s digitálními technologiemi obává pasivity nejen dětí, ale i učitelů. „je pohodlnější zapnout dětem televizi“ Takováto pasivní aktivita pak může být náporom na zrak dětí, může omezovat jejich pohyb, minimálně rozvíjet motorické schopnosti a omezovat sociální kontakty, a tím socializaci dětí v průběhu digitálních aktivit. Díky chápání digitálních technologií, jako zdroje pasivní zábavy, mohou rodiče nesouhlasit s využíváním digitálních technologií při přímé pedagogické činnosti.

Učitelé zmiňují digitální technologie, jako zdroj inspirace (nově často zmiňovaný Pinterest), případně jako zdroj didaktických materiálů.

Další učitelé oceňovaný aspekt využití digitálních technologií byla možnost lepší, jednodušší a efektivnější komunikace s rodiči. Například: „IT se mohou stát dobrým odrazovým můstkem k zahájení pravidelné a účelné spolupráce mateřské školy s rodiči, v každé třídě je notebook a učitelka pravidelně komunikuje s rodiči, posílá fotky a učební materiály.“ Aspekt komunikace s rodiči učitelé nově mnohem více zdůrazňují. Domníváme se, že je to důsledek především současné epidemie covid 19.

Většina učitelů mateřských škol mluví o digitálních technologiích při výuce, jako o zajímavém zpestření či doplňku. (Čemuž také odpovídá uváděná četnost jejich využívání digitálních technologií při přímé pedagogické činnosti například: Zároveň většina učitelů, kteří mají přístup k interaktivní tabuli, s ní pracují maximálně jedenkrát týdně). Pouze jeden učitel uvedl jako klad využití digitálních technologií rozvoj myšlení, schopnost plánovat (oba dva jedenkrát). Celkem tři učitelé v souvislosti s robotickou včelkou bee-boot uvedli jako klad rozvoj prostorové orientace. Pojmy jako algoritmizace, abstrakce, generalizace související s rozvojem inforatického myšlení [5, 6] nebyly zmíněny ani jednou.

3.3 Faktory ovlivňující vztah učitelů mateřských škol k využití digitálních technologií při přímé pedagogické činnosti

Názor učitelů na digitální technologie a jejich využívání v přímé pedagogické činnosti přímo ovlivňují četnost a způsob využití digitálních technologií ve třídě. Jako například „Jakékoli technologie odsuzuje. V její třídě najdeme interaktivní tabuli, ale tu vůbec nevyužívá. (32 let)“ Opakem je: „Technologie při práci s dětmi miluju! Já i děti se vždy dokážeme vyřadit v rozmanitostech, které různé aplikace nabízejí. Zkušenost mám se vším, co mi přijde pod ruku (tablet, notebook, fotoaparát, interaktivní tabule, mobil, videonahrávky,.. Nebojím se objevovat a experimentovat. (27 let)“ Je proto důležité lépe pochopit pohnutky a názory učitelů mateřských škol na využívání digitálních technologií. V této kapitole budeme zkoumat především pohnutky vedoucí k odmítnutí práce s digitálními technologiemi.

Mnozí učitelé pozorují na dětech negativní aspekty nadužívání digitálních technologií v domácím prostředí. Tyto zkušenosti mnohé z pedagogů mateřských škol vedou k rezervovanému, či odmítavému postoji k využívání digitálních technologií při přímé pedagogické činnosti. Jako například: „Většina dětí má digitální technologie doma, kde často není čas strávený užíváním těchto médií regulován (nebo jen velmi málo). Pro tyto děti může tedy být pobyt v MŠ jediným časem stráveným bez digitálních technologií.“

Z kontextu vyjádření učitelů se domníváme, že velká část učitelů mateřských škol příliš nezná jiný způsob využití digitálních technologií, než je domácí konzumace digitálního obsahu. Např: DT nekladou takový nárok na učitelky v MŠ, které se stávají pasivnějšími (je pohodlnější zapnout dětem televizi)“ A protože učitelé spatřují na dětech mnoho negativních důsledků takového využívání digitálních technologií, logicky odmítají využívání digitálních technologií v mateřských školách. Mnoho učitelů proto jako negativum využití digitálních technologií v mateřských školách zmiňuje pasivitu dětí.

Dalším důležitým faktorem ovlivňujícím názor učitelů na digitální technologie je věk učitelů. Využití digitálních technologií k přímé pedagogické práci odmítají především starší ročníky učitelů, kteří často zároveň mají podle svých tvrzení nižší úroveň digitální gramotnosti. Domníváme se, že díky své nízké digitální gramotnosti si neuvědomují možnosti, které jim využití digitálních technologií přináší při přímé pedagogické činnosti. Zároveň mají již nastavený funkční systém výuky bez digitálních technologií. Jako například: „Paní jsou učitelky již starší a samy mají problém i s ovládním počítače a přijmutím nových směrů a jiné práce s dětmi. Tím, že ony tyto technologie moc neovládají a ani neví, co by vlastně mohly chtít, jak a které technologie využít“ Což může mít za následek, že i když jsou digitální technologie k dispozici, nejsou využívány. „Děti využívají tablety při práci ovšem velmi málo, protože v těchto třídách vyučují starší kolegyně, které nemají interaktivní tabuli příliš v oblíbě a raději si pracují podle svých „zajetých“ metod.“ Ovšem existují i výjimky starších učitelů, kteří dokázali výrazně pozvednout svou digitální gramotnost: „Učitelky se v počáteční fázi s technikou sžívaly metodou pokusu omylu. Svě zkušenosti s používáním IT si mezi sebou sdílely a zanedlouho byly poradním hlasem dětí, které se techniky nebojí a činnosti na IT i s tablety si užívají.“ Věkové rozmezí těchto učitelek se pohybuje mezi 40 a 60 lety. Všechny tedy jsou „digitálními imigranty“.

Dalšími důvody odmítnutí digitálních technologií pro přímou pedagogickou činnost jsou řekněme „filozofického“ charakteru, kdy učitelé typicky názorově souzní s nějakým „alternativním“

pedagogickým směrem. Filozofické předpoklady tohoto směru jsou pak v konfliktu s využíváním digitálních technologií při přímé pedagogické činnosti. Velmi nás ale překvapilo, že i přes odmítnutí digitálních technologií pro přímou pedagogickou činnost, byla v jedné mateřské škole využívající prvky Montessori pedagogiky zastoupena robotická včelka bee-bot. Včelka nemá žádný displej, domníváme se proto, že mnohými učiteli není chápána jako klasické digitální zařízení.

Zmíněným důvodem k odmítnutí využití digitálních technologií při výuce byla také osobní negativní zkušenost. „Mám doma dvě dopívající děti. Technologie je doslova pohltila. Vůbec nevím, kdy se zrodila tahle závislost. Nikdy jsme je s manželem k tak nadměrnému užívání nevedli. Teď už se můžu jen modlit a doufat, že jsou vychovaní a ví, co se sluší a hlavně kde jsou určité hranice. Proto se využívání technologií v mateřské škole momentálně snažím vyhábat.“

3.4 Vztah vedení mateřských škol k využívání digitálních technologií

Ředitelé mateřských škol rozhodují o jednotlivých nákupech digitálních technologií, díky čemuž zásadně ovlivňují penetraci digitální techniky v mateřských školách. Příkladem může být: „Vedení úplně nesouhlasí s využíváním digitálních technologií u nás v mateřské škole, a to hlavně z důvodu, že tyto technologie nepodporuje. To znamená, že u nás v mateřské škole se nenachází žádné zařízení digitální technologie, takže veškerá digitální zařízení, která chce pedagog využít při práci s dětmi, si musí sám přinést.“ V opačném případě může nastat situace, kdy ředitelé zakoupí digitální zařízení (typicky interaktivní tabuli) a toto zařízení, se ale následně dostane k paní učitelce, která jej nepoužívá.

Ředitelé na základě svého názoru ovlivňují názory svých podřízených učitelů. Ředitelé mohou diskutovat, přesvědčovat, či vyžadovat využití digitálních technologií při výuce. Jako například: „Ze strany vedení jsem byla k aktivitám přímo vybízena.“ Díky vysoké digitální gramotnosti mnohých učitelů mateřských škol, může být důležitá technická podpora či různé školení, týkající se využití digitálních technologií v mateřských školách. Z pohledu učitelů „je především důležitá podpora, pochopení a ochota pomoci ze strany vedení.“

Z našeho porovnání dat vychází, že ředitelé mateřských škol mají o něco pozitivnější pohled na využívání digitálních technologií při přímé pedagogické činnosti, oproti běžným učitelům mateřských škol. Jako například: „Už jsem s tímto plánem kolegyni seznámila ☺, jen nevím, jestli jsem ji natolik přesvědčila, že to bude skvělé, nebo za to může můj post ředitelky. Už si dokážu představit místo, kde bude interaktivní tabule umístěna.“ Většina ředitelů mateřských škol se tak snaží shánět finanční prostředky a postupně vybavovat učební prostory digitálními technologiemi.

Několik ředitelů mateřských škol zastávalo negativní postoj k využívání digitálních technologií při přímé pedagogické činnosti. Tito ředitelé výrazně brzdí využití digitálních technologií ve svých mateřských školách. V těchto mateřských školách se digitální technologie používají naprosto minimálně. Učitelé, kteří je chtějí používat, tak musí používat své soukromé technologie.

4 DISKUSE VÝSLEDKŮ

Učitelé často zmiňované nadužívání digitálních technologií dětmi v rodinném prostředí a jeho negativní důsledky ve formě omezení sociálních kontaktů dětí, komunikačních schopností a pohybových aktivit dětí, chápeme jako možný ukazatel alarmujícího problému.

Tento problém se týká již velmi malých dětí. Z našeho soukromého života, například víme o 18ti měsíčním dítěti, které již zvládá základy ovládání smartphonu. Problém předčasného využívání digitální techniky je třeba dále pečlivě monitorovat. V dalších výzkumech je vhodné zaměřit se na důsledky tohoto problému.

Ve shodě s většinou učitelů se domníváme, že děti se od brzkého mládí v rodinném prostředí setkávají s digitálními technologiemi. Proto je třeba cíleně budovat vztah dětí k nim tak, aby děti využívaly digitální technologie nejen pro konzumaci obsahu, ale i jako kreativní, nebo pracovní nástroje.

Ve výsledcích výzkumu se ukazuje, že většina učitelů mateřských škol chápe digitální technologie při výuce jako zajímavé zpestření či doplněk. Z našeho pohledu jde o nedocení možnosti využití digitálních technologií při přímé pedagogické činnosti. Zároveň jsme jak v tomto výzkumu, tak v našem starším výzkumu [1] zaznamenali mnohé případy nákupu velmi drahé digitální techniky (typicky interaktivní tabule) a jejího následného minimálního používání.

Jako řešení těchto problémů spatřujeme v cíleném rozvoji digitální gramotnosti a prezentaci možností smysluplného využití digitálních technologií v pregraduální přípravě budoucích učitelů mateřských škol.

5 ZÁVĚR

Učitelé mateřských škol pozorují konkrétní negativní důsledky nadužívání digitálních technologií dětmi v rodinném prostředí. Což vede menší část učitelů k odmítavému postoji k využívání digitálních technologií při přímé pedagogické práci. Někteří učitelé také odmítají digitální technologie z „filozofických“ postojů, týkajících se většinou názorové sounáležitosti s nějakou z alternativních pedagogických metod. Většina učitelů ale přistupuje k digitálním technologiím realisticky, uvědomuje si, že digitální technologie jsou běžnou součástí našeho světa a díky jejich integraci do výuky tak mohou utvářet vztah dětí k digitálním technologiím. Tito učitelé se většinou snaží představovat digitální technologie jako pracovní či kreativní nástroj. Na digitálních technologiích oceňují názornost. Využívají jich k usnadnění výkladu, demonstraci probíraných témat či jako zdroj informací. Velmi oceňují motivační aspekt digitálních technologií a možnost snadnější a efektivnější komunikace s rodiči, u které digitální technologie otvírají nové formy komunikace, například sdílení fotografií či videozáznamů. Naopak jako negativní aspekt využívání digitálních technologií spatřují omezený kontakt dítěte se zkoumaným objektem a omezené možnosti sociálních kontaktů, a tím socializace dětí v průběhu digitálních aktivit. Většina učitelů mateřských škol chápe ale digitální technologie při přímé pedagogické činnosti pouze jako zpestření či doplněk.

Ředitelé mateřských škol rozhodují o nákupu jednotlivých digitálních zařízení a na základě svého názoru mohou ovlivňovat názory svých podřízených učitelů. Ředitelé mateřských škol mají o něco pozitivnější pohled na využívání digitálních technologií při přímé pedagogické činnosti, oproti běžným učitelům mateřských škol. Hlavním identifikovaným faktorem ovlivňující vztah učitelů k využívání digitálních technologií je věk učitelů a s ním často spojená jejich nízká digitální gramotnost, která mimo jiné neumožňuje učitelům nahlédnout jiné možnosti využití digitálních technologií při výuce, než je pouze konzumace digitálního obsahu. A protože spatřují na dětech mnoho negativních důsledků takového využívání digitálních technologií, logicky odmítají využívání digitálních technologií v mateřských školách při přímé pedagogické činnosti.

BIBLIOGRAFICKÉ ODKAZY

- [1] DOBIÁŠ, V. Real Use of Digital Technologies in Preschool Education. In: *Didinfo 2020*. Liberec: TUL, 2020, s. 30-37. ISBN 978-80-7494-532-8. Dostupné též: http://www.didinfo.net/images/DidInfo/files/Didinfo_2020.pdf
- [2] FRIESE, S. *Qualitative data analysis with ATLAS.ti*. London: SAGE, 2012, 274 s. ISBN 978-085-7021-304.
- [3] STRAUSS, A., CORBINOVÁ J. *Základy kvalitativního výzkumu: Postupy a techniky metody zakotvené teorie*. Boskovice: Albert, 1999, 196 s. ISBN 80-858-3460-X.
- [4] ŠVAŘÍČEK, R., ŠEĐOVÁ K. *Kvalitativní výzkum v pedagogických vědách*. Praha: Portál, 2007, 377 s. ISBN 978-80-7367-313-0.
- [5] LESSNER, D. Analýza významu pojmu Computational thinking. In *Journal of Technology and Information*. 2014, roč. 6, č. 1, 71-88. ISSN 1803537X.
- [6] MANĚNOVÁ, M. PEKÁRKOVÁ S. *Algoritmizace s využitím robotických hraček pro děti do věku 8 let*. Hradec Králové: UHK, 2020, 73 s. ISBN 978-80-7435-775-6

Obrátená výučba programovania na vysokej škole Flipped learning of programming course at university

Ján Guniš

UPJŠ v Košiciach, Prírodovedecká fakulta
Jesenná 5
040 01 Košice
Slovensko
jan.gunis@upjs.sk

Lubomír Šnajder

UPJŠ v Košiciach, Prírodovedecká fakulta
Jesenná 5
040 01 Košice
Slovensko
lubomir.snajder@upjs.sk

ABSTRACT

In the article, we present our experience and findings from the flipped learning of programming in Python at the Further study of informatics for in-service teachers at the Faculty of Science, P. J. Šafárik University in Košice. We exploited LMS Moodle to communicate with students. After the forced transition to the distance form of teaching (due to anti-pandemic COVID-19 measures), we also used the BigBlueButton videoconferencing system and its GreenLight extension. For delivery of learning content to students for their home preparation, we used interactive worksheets in the format of the open-source web application Jupyter Notebook. Another specific feature of this teaching was the participation of two teachers. The results of this case study show that students appreciate the interactive worksheets provided, which they could study at their own pace, formative assessed conceptual tests, discussions on problem-solving, publishing of video recordings, and solved problems from online lectures. From the teachers' point of view, we were able to focus more on explaining key concepts in our lectures, solving more difficult problems, discussing with students who were more active than if they had passively listened to the traditional interpretation by the teacher.

Keywords

Flipped learning. Programming education. Distance education. ConcepTest. COVID-19. Case study.

ABSTRAKT

V článku uvádzame naše skúsenosti a zistenia z obrátenej výučby programovania v Pythone v rozširujúcom štúdiu informatiky na Prírodovedeckej fakulte UPJŠ v Košiciach. Na komunikáciu so študentami sme použili LMS Moodle. Po nútenom prechode na dištančnú formu výučby (z dôvodu protipandemických COVID-19 opatrení) sme použili aj videokonferenčný systém BigBlueButton a jeho nadstavbu GreenLight. Pre sprístupnenie učebného obsahu študentom pre ich domácu prípravu na vyučovanie sme použili interaktívne pracovné listy vo formáte otvorenej webovej aplikácie Jupyter Notebook. Ďalším špecifikom výučby boli dvaja učelia. Výsledky tejto prípadovej štúdie ukazujú, že študenti najviac oceňujú poskytnuté interaktívne pracovné listy, ktoré si mohli preštudovať svojim tempom, formatívne hodnotené konceptuálne testy, diskusie k riešeniam úloh, zverejnenie videozáznamov a riešení úloh z online výučby. Z pohľadu učiteľov sme sa mohli v našej výučbe viac venovať vysvetľovaniu kľúčových pojmov, riešeniu náročnejších problémov, diskusií so študentami, ktorí boli aktívnejší, ako keby pasívne počúvali tradičný výklad učiteľa.

Kľúčové slová

Obrátená výučba. Výučba programovania. Dištančné vyučovanie. Konceptuálny test. COVID-19. Prípadová štúdia.

1 ÚVOD

Podobne ako väčšina pedagógov aj my neustále hľadáme nové učebné prístupy, nástroje a pomôcky pre skvalitnenie vyučovania programovania na vysokej škole. Zvlášť cenná je pre nás spolupráca s didaktikmi iných vyučovacích predmetov. Vzájomná výmena skúseností a postupov, najmä s didaktikmi fyziky, nás inšpirovala k aplikácii modelu obrátenej výučby [1] do výučby programovania. Tento model výučby sa nám zdal zvlášť vhodný najmä pre študentov, s ktorými nemáme denný kontakt, takže spoločný čas musíme využívať čo najefektívnejšie. Model obrátenej výučby sme využili pri študentoch prvého ročníka Rozširujúceho štúdia informatiky (RŠI).

Základný kurz programovania na RŠI sa u nás vyučuje tri semestre. V prvom semestri je programovacím jazykom Scratch [2], ďalšie dva semestre študenti programujú v jazyku Python [3]. Vďaka popularite tohto jazyka je k dispozícii množstvo rôznych prostredí, v ktorých je možné programy vytvárať. Zaujalo nás vývojové prostredie Jupyter Lab [4], v ktorom je možné vo formáte Jupyter Notebook vytvárať pythonovské interaktívne pracovné listy pre vyučovanie programovania. Učebné texty sa tak presúvajú z pozície statických materiálov (určených len na čítanie) do pozície živých dokumentov, s ktorými môže študent interagovať a modifikovať ich obsah.

Štandardnou súčasťou elektronickej podpory výučby na Prírodovedeckej fakulte UPJŠ v Košiciach je už dlhé obdobie aj LMS Moodle [5].

Vo februári 2020 sme začali testovať a experimentovať s videokonferenčným systémom BigBlueButton [6], v ktorom sme plánovali realizovať niektoré stretnutia Klubu učiteľov informatiky (KUI). Na základe protiepidemických COVID-19 opatrení sme od marca 2020 realizovali stretnutia účastníkov KUI už len výlučne formou webinárov využitím BigBlueButton. Tieto skúsenosti s BigBlueButton a náhla zmena situácie nás primáli veľmi skoro prejsť na výlučne dištančnú výučbu programovania na RŠI a aj na ostatných našich akademických predmetov.

V nasledujúcej časti článku opisujeme spôsob realizácie obrátenej výučby programovania na RŠI, naše skúsenosti a postrehy z vyučovania.

2 VÝCHODISKÁ PRE OBRÁTENÚ VÝUČBU PROGRAMOVANIA

2.1 Model obrátenej výučby

Najdôležitejším prvkom inovácie našej výučby programovania je obrátená výučba (angl. flipped learning, resp. flipped classroom, resp. inverted classroom). Podľa Bergmanna a Samsa [7] je obrátená výučba pedagogickým prístupom, pri ktorom sa priama výučba učiteľa presúva zo skupinového učebného priestoru do individuálneho učebného priestoru a výsledný skupinový učebný priestor sa transformuje do dynamického interaktívneho učebného prostredia, v ktorom pedagóg usmerňuje študentov, aby aktívne aplikovali poznatky a tvorivo sa zapájali do výučby študovaného predmetu.

Glazer v [8] zaraďuje model obrátenej výučby do tzv. zmiešanej výučby (angl. blended learning). Ide o didakticky premyslené efektívne spojenie individuálneho online vzdelávania s jednoduchšími aktivitami (napr. sledovanie videí s výkladom, čítanie učebného textu, jednoduché experimentovanie) realizovaných mimo kontaktnej výučby a náročnejších aktivít (napr. riešenie problémov, skúmanie, bádanie, aplikácia učiva, hodnotenie, diskusia, či realizácia projektov) realizovaných počas kontaktnej výučby.

Aby výučba prebiehala v zmysle modelu obrátenej výučby, musia byť splnené nasledovné štyri kľúčové prvky (pilieri) [9]:

- pružné prostredie (angl. flexible environment) umožňujúce realizáciu rôznych spôsobov vzdelávania a typov vzdelávacích aktivít,
- kultúra vzdelávania (angl. learning culture), kde v centre pozornosti nie je učiteľ, ale študent,
- vhodný výber učebného obsahu (angl. intentional content), kde cieľom učebných materiálov je dosiahnutie hlbšieho konceptuálneho porozumenia a zručnosti pri riešení problémov,
- učiteľ ako profesionál (angl. professional educator), ktorý má skúsenosti s riadením triedy, s využívaním interaktívnych metód a digitálnych technológií.

2.2 Konceptuálne testy

Dôležitou súčasťou obrátenej výučby sú konceptuálne testy, ktoré umožňujú diagnostikovať do akej miery študenti správne porozumeli vybraným konceptom. Mazur v svojej knihe [10] zameranej na rovesnícke učenie sa (angl. peer instruction) zavádza pojem conceptTests ako súbor niekoľkých krátkych konceptuálnych úloh k diskutovanému učivu. Tieto by mali jednak primäť študentov, aby pri premýšľaní používali a rozvíjali svoje argumenty; a tiež poskytnúť im, ako aj učiteľom zhodnotenie ako porozumeli kľúčovým pojmom aktuálneho učiva. Mazur na základe svojich skúseností zo svojich vysokoškolských kurzov fyziky odporúča, aby tieto úlohy:

- boli zamerané na jeden koncept,
- nedali sa priamočiaro vyriešiť (napr. dosadením do vzorca),
- obsahovali dobré relevantné odpovede (správne aj distrakory),
- boli jasne formulované,
- boli stredne náročné.

Spočiatku sa odporúčalo, aby konceptuálne testy obsahovali len testovacie položky s výberom odpovede (angl. multiple choice). Súčasná technológia umožňuje použitie aj iných typov položiek, napr. dotyková slepá mapa. Konceptuálne testy môžu byť použité na zistenie miery porozumenia jednak prvotných poznatkov učiva, jednak aktuálneho učiva.

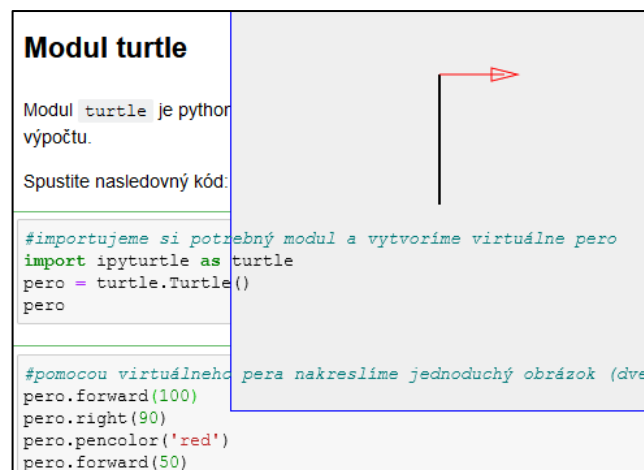
Podľa [11] konceptuálne testy zvyčajne pozostávajú z jednej až piatich položiek s možnosťou výberu z viacerých odpovedí. Primárnym účelom konceptuálnych testov je získanie prehľadu o aktuálnej úrovni porozumenia učivu celej triedy, nie konkrétneho študenta. Konceptuálne testy sú určené pre formatívne hodnotenie výučby, preto zvyčajne nie sú klasifikované, a ak, tak len s nízkou váhou.

2.3 Dvaja učitelia

Naše skúsenosti s výučbou programovania na RŠI ukazujú, že problematika programovania je pre týchto študentov pomerne náročná. Náročný je najmä druhý semester základného kurzu programovania, kde sa študenti prvýkrát stretávajú s textovým programovacím jazykom. Z tohto dôvodu boli na výučbe prítomní dvaja učitelia. Jeden z učiteľov primárne vedie výučbu, diskutuje so skupinou študentov a kladie otázky študentom. Sekundárny učiteľ rieši individuálne problémy a otázky študentov, prípadne vhodne dopĺňa komentármi primárneho učiteľa. Obaja učitelia tvoria edukačný obsah predmetu (študijné materiály, konceptuálne testy, zbierky úloh) a vyhodnocujú výučbu. Koncept dvoch učiteľov z prezenčnej výučby sme použili aj v dištančnej výučbe.

2.4 Jupyter Notebook

Prostredie Jupyter Lab je interaktívne vývojové prostredie pre vytváranie a správu dokumentov vo formáte Jupyter Notebook. Jupyter Notebook je otvorená webová aplikácia umožňujúca vytvárať a zdieľať dokumenty, ktoré obsahujú živý kód, texty, obrázky a vizualizácie priebehu výpočtu (Obrázok 1).



```
Modul turtle je python výpočtu.
Spustite nasledovný kód:

#importujeme si potrebný modul a vytvoríme virtuálne pero
import ipyturtle as turtle
pero = turtle.Turtle()
pero

#pomocou virtuálneho pera nakreslíme jednoduchý obrázok (dve
pero.forward(100)
pero.right(90)
pero.pencolor('red')
pero.forward(50)
```

Obrázok 1: Časť interaktívneho dokumentu v prostredí Jupyter Notebook s aktívnym kódom v jazyku Python a vizualizáciou priebehu výpočtu pomocou korytnačky (modul ipyturtle)

Pre jeho použitie stačí webový prehliadač. K dispozícii je množstvo online inštalácií, ktoré je možné bezplatne používať (napr. <https://jupyter.org/try>). Nevýhodou je, že sa nemôžeme spoľahnúť na dostupnosť prostredia, lebo bývajú pomerne často vďaka veľkej vyťažnosti nedostupné. Z tohto dôvodu sme sa rozhodli pre

vlastnú inštaláciu na serveri fakulty (<https://jupyter.ics.upjs.sk/>). Z dôvodu bezpečnosti je prístup chránený heslom. Nastavenie systému nám umožnilo distribuovať študentom pripravené interaktívne pracovné listy a prehliadať ich individuálne riešenia úloh z pracovných listov. Kód z prostredia Jupyter Notebook je možné prakticky bez zmeny preniesť do lokálneho vývojového prostredia.

2.5 Videokonferenčný systém BigBlueButton

BigBlueButton je open-source webový konferenčný server. Vďaka nadstavbe GreenLight [12] je možné vytvárať vlastné virtuálne miestnosti pre organizovanie konferencií alebo sa pripájať ku konferenciám ostatných používateľov kliknutím na prijatý odkaz. Systém umožňuje prenos diania na obrazovke prednášajúceho, vzájomný prenos video a audio komunikácie medzi účastníkmi konferencie, chat, priestor pre zdieľané poznámky, rýchle hlasovanie v anketách a záznam a publikovanie priebehu stretnutia. Vďaka licencií a z podobných dôvodov na požiadavku dostupnosti ako pri prostredí Jupyter Notebook sme sa rozhodli pre vlastnú inštaláciu systému (<https://bbb.science.upjs.sk/>). V momente zmeny prezenčnej výučby na dištančnú sme tak boli prakticky okamžite technicky pripravení.

2.6 LMS Moodle

LMS Moodle používame na fakulte vo výučbe dlhodobo (od roku 2003). Okrem niekoľkých predmetov, ktoré boli už od začiatku dizajnované pre dištančnú výučbu, ho využívame najmä ako elektronickú podporu prezenčnej výučby. Pri prechode na dištančnú formu výučby sa existujúci Moodle kurz spolu s videokonferenčným systémom BigBlueButton stal hlavným komunikačným centrom medzi učiteľmi a študentami.

3 NÁVRH A REALIZÁCIA OBRÁTENEJ VÝUČBY PROGRAMOVANIA

3.1 Cieľová skupina

Cieľovou skupinou bola skupina 12 študentov RŠI, z toho 6 žien a 6 mužov pričom 7 z nich bolo zo základných škôl a 5 zo stredných škôl. Študenti už absolvovali prvý z trojice predmetov základného programovania, ktorý bol realizovaný v jazyku Scratch [2] a poznajú základné koncepty programovania (premenná, cyklus, podmienené vetvenie, podprogramy s parametrom, udalosťami riadené programovanie). Druhý v poradí je predmet QPR2 Programovanie L2 – Python, výučbu ktorého v tomto článku opisujeme.

3.2 Obsah vzdelávania

Vzdelávanie sa realizovalo počas 11 týždňov v dvojhodinových blokoch. Obsah predmetu sme rozdelili do nasledujúcich častí:

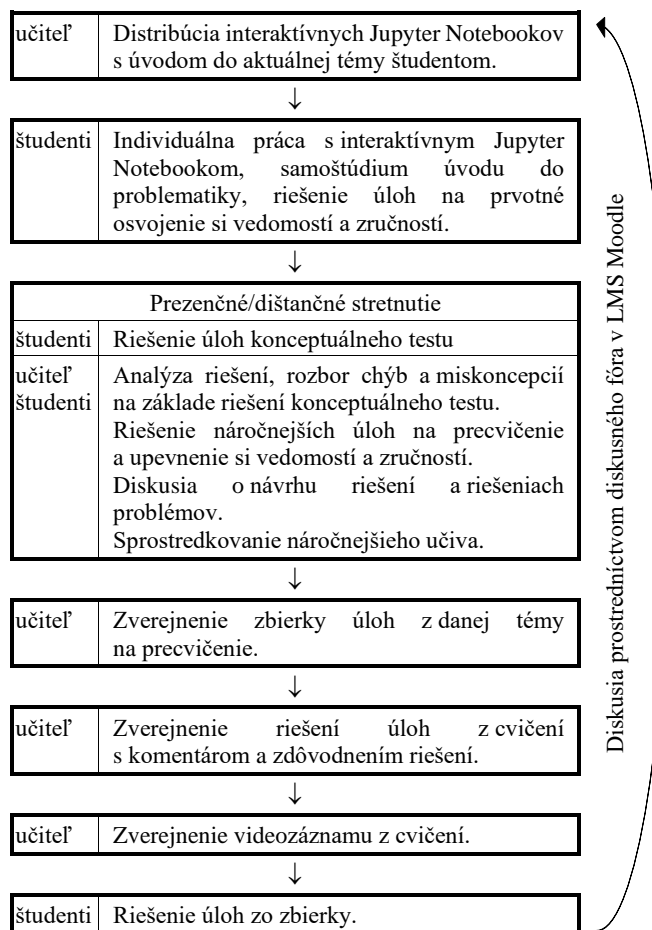
- Úvod do jazyka Python, vývojové prostredie, konzola, premenná,
- Korytnačia grafika, funkcie bez parametrov a návratovej hodnoty,
- Cyklus for a funkcia range,
- Funkcia s parametrom a návratovou hodnotou,
- Priebežný písomný test 1, Logické výrazy a príkaz vetvenia if,
- Reťazce, metódy reťazcov, funkcie pre prácu s reťazcami,
- Algoritmy na reťazcoch,
- Odchytávanie výnimiek,
- Generovanie a odchytávanie výnimiek,

- Priebežný písomný test 2, Zoznamy, metódy zoznamov, funkcie pre prácu so zoznamami,
- Algoritmy na zoznamoch.

3.3 Priebeh vzdelávania

Prvé štyri stretnutia výučby predmetu prebiehali prezenčnou formou. V dôsledku preventívnych opatrení v súvislosti s ochorením COVID-19 sa zvyšné stretnutia realizovali dištančne.

Jednotlivé stretnutia prebiehali podľa nasledovnej schémy (Obrázok 2)



Obrázok 2: Schéma priebehu obrátenej výučby

Študentom sme pred výučbou distribuovali interaktívne pracovné listy vo formáte Jupyter Notebook. Pracovné listy obsahovali definované ciele výučby, úvod do aktuálnej problematiky doplnený o demonštračné príklady. Okrem samotnej demonstrácie slúžili uvedené príklady a úlohy aj na experimentovanie, pretože študenti mohli uvedený kód upravovať a opätovne spúšťať (pracovné listy často obsahovali výzvu na nejakú zmenu kódu). Pracovný list končil mini zbierkou úloh, pri riešení ktorých mali študenti použiť práve získané vedomosti a zručnosti. Ak sa študentom nepodarilo úlohy vyriešiť, bola to pre nich okamžitá spätná väzba, že uvedenú problematiku nezvládli na dostatočnej úrovni a mali by sa jej ešte venovať. Je potrebné poznamenať, že úlohy, ktoré mali študenti v tejto fáze riešiť, neboli náročné a vo väčšine prípadov vyžadovali priame použitie poznatkov uvedených v pracovnom liste.

Na začiatku spoločných stretnutí študenti riešili test zameraný na konceptuálne porozumenie učivu. Úlohy nevyžadovali písanie

veľkého množstva kódu. Väčšinou išlo o analýzu krátkeho kódu a výber správnych možností z ponuky (Obrázok 3). Odpovede študentov sme vzápätí spoločne analyzovali s cieľom odhaliť ich miskoncepce. Pre spoľahlivejšie diagnostikovanie porozumenia študentov a zistenia ich miskonceptí sme zaradili položky s výberom viacerých správnych odpovedí, resp. s krátkou odpoveďou. Riešenie testu nebolo hodnotené známku, ani bodmi. Test slúžil na formatívne hodnotenie, ako spätná väzba z krátkeho obdobia učenia sa.

Uvedte pomocou ktorých z uvedených príkazov vložíme na koniec zoznamu `z = [1, 2, 3, 4]` hodnotu 50:

- `z[4] = 50` `z = z.append(50)`
 `z.append(50)` `z = z + [50]`

Uvedte pomocou ktorých z uvedených programových kódov vypíšeme prvky zoznamu `z = [1, 2, 3]`:

`for cislo in range(len(zoznam)):`
 `print(cislo)`

`for cislo in zoznam:`
 `print(cislo)`

`print(zoznam)`

`print(str(zoznam))`

`for cislo in range(len(zoznam)):`
 `print(zoznam[cislo])`

Obrázok 3: Konceptuálny test k téme zoznamy a metódy zoznamov

Samotné cvičenie bolo venované zvyšku preberanej témy, riešeniu náročnejších úloh. V rámci možnosti, ktoré nám situácia umožňovala, sme cvičenia viedli formou diskusií o analýze úloh a o ich riešeníach so snahou čo najviac zapojiť študentov. Z dôvodu obmedzeného internetového pripojenia niektorých študentov sme boli nútení minimalizovať dátový tok vypnutím kamier. Týmto sme síce obmedzili možnosť neverbálnej komunikácie, ale výhodou bolo, že priebeh cvičenia mohli študenti sledovať bez prerušovania.

Úlohy, ktoré sme na cvičeniach spoločne vyriešili, sme v krátkom čase po ukončení cvičení zverejnili spolu s riešeniami a súčasne aj s komentárom k riešeniam. Študenti tak dostali v kompaktnej podobe spracovanú časť cvičenia. Rovnako sme zverejnili aj zbierku úloh k danej téme a kompletný videozáznam priebehu cvičenia. Celkovo sme zverejnili 11 tematicky zameraných zbierok s celkovo 95 zadaniami úloh.

Približne v 1/3 a v 2/3 semestra sme študentom zadali test. Testy boli hodnotené bodmi, ktoré sa započítavali do celkového hodnotenia predmetu. Hodnotenie z testu slúžilo študentom zároveň ako spätná väzba z dlhšieho obdobia učenia sa. Ak študenti v týchto testoch zlyhali, stále mali možnosť predmet absolvovať s pomerne dobrým hodnotením. Bolo to však upozornenie, že ich dlhodobšia príprava na výučbu nie je dostatočná. V hodnotení riešení sme sa neobmedzili len na bodové skóre. Súčasťou spätnej väzby bol aj komentár, v ktorom študent dostal informáciu, kde a akú chybu spravil a ako by mali postupovať, aby odpoveď bola správna.

Študenti RŠI študujú popri zamestnaní a ich priama výučba prebieha len jeden deň v týždni. Majú teda značne obmedzené možnosti na vzájomne prezenčné diskusie. Tento nedostatok sme čiastočne kompenzovali diskusným fórom, v ktorom študenti navzájom medzi sebou a s učiteľmi diskutovali problematiku časti učiva, riešenia úloh a pod. Využitie tejto možnosti sme nechali na študentoch.

3.4 Overenie dosiahnutia cieľov vzdelávania

Študenti boli hodnotení formatívne aj sumatívne. Formatívne hodnotenie prebiehalo priebežne, počas celého semestra formou konceptuálnych testov na začiatku cvičení a spätnou väzbou od učiteľov v diskusnom fóre k riešeniam úloh zo zbierok úloh.

Sumatívne hodnotenie bolo realizované formou písomných testov na konci prvej a druhej tretiny semestra a záverečnou písomnou skúškou na konci semestra.

4 ZHODNOTENIE OBRÁTENEJ VÝUČBY PROGRAMOVANIA

Absolventov predmetu QPR2 Programovanie L2 – Python sme s odstupom času požiadali o spätnú väzbu. Dôvodom nebol len spätný pohľad na realizáciu výučby, ale aj snaha o vylepšenie kurzu v nasledujúcom akademickom roku. Nižšie uvádzame niektoré z odpovedí (vyznačené kurzívou) a naše komentáre k nim.

Dištančné vzdelávanie bolo pre študentov samo o sebe náročné. Zvlášť náročné bolo pre študentov učenie sa programovať: „*Programovanie je však náročným predmetom, je potrebné jeho dlhšie osvojovanie, precvičovanie a riešenie daných problémov, v rámci dištančnej výučby nie je v silách ani pedagóga, ani v nastavených podmienkach aby vyrovnal kvantitu prebratého učiva počas hodiny na úroveň prezenčnej výučby.*“

Absencia osobných interakcií, zdlháva a náročnejšia komunikácia, problém spontánne reagovať počas online stretnutí sú postrehy, ktoré študenti uvádzali pri hodnotení dištančnej výučby ako takej. Na druhej strane však študenti uvádzali aj fakt, že na dištančnú výučbu si postupne zvykli a nachádzali si spôsoby, ako ju efektívnejšie zvládajú.

Interaktívne pracovné listy študenti spätne hodnotili veľmi pozitívne. Za všetky jedna z odpovedí: „*V nadväznosti na predhádzajúcu odpoveď bol v tomto dobrý Jupyter, kde sme mali dopredu učivo, mohli sme si vyskúšať jednotlivé príkazy, a keď mi niečo nebolo jasné, mohla som sa potom na hodine už k tomu pýtať, lebo som už vedela, čo mi je jasné a čo nie.*“ Študenti ďalej vyzdvihovali možnosť, že na jednom mieste našli podstatné veci k danej problematike, uvedené poznatky si mohli ihneď prakticky overiť a nemuseli „študovať“ rozsiahle materiály z iných zdrojov. Táto možnosť ale nevyhovuje každému. Zaznamenali sme aj opačnú reakciu „*Toto mne osobne nesedí, nevyhovuje, neviem tomuto prísť na chuť. Je to niečo iné, možno inovatívne, no uprednostňujem najskôr výklad, vysvetlenie, nácvik až tak zadanie úloh.*“

Zaujímavé je porovnať reakcie na konceptuálne testy a na priebežné písomné testy. Cieľom oboch bolo poskytnúť študentom spätnú väzbu, formatívne hodnotenie. Obidve činnosti boli časovo obmedzené a vopred plánované. Okrem typu úloh bol rozdiel v tom, že hodnotenie priebežných písomných testov bolo súčasťou celkového hodnotenia. Kým konceptuálne testy študenti hodnotili spôsobom: „*riešenia konc.testov na začiatku vyučovacej hodiny boli zaujímavé a myslím si, že aj poučné*“, „*Áno toto je dobrá vec prejsť si veci, ktoré človek nepochopil keď riešil doma úlohy, ja by som bral aj viac času na analýzu niektorej problematiky ale viem*

že kvôli časovému harmonogramu sa to asi nedalo“ tak priebežné písomné testy sa takej obľube netešili. Hlavným problémom sa ukázal časový stres: „pri písaní priebežných písomiek mi robil problém stres a sledovanie času, v časovom strese neviem rýchlo reagovať“, „Programovanie na krátky vymedzený čas vnímam ako frustrujúce a demotivujúce“. Zatiaľ čo pri prezenčnej výučbe je učiteľ ten, kto kontroluje čas a je ochotný „sekundu“ počkať, kým študent dopíše vetu, v dištančnej výučbe sa o to stará systém, ktorý takýto ľudský prvok nemá. Zaznamenali sme aj nedovolenú spoluprácu, keď študenti odovzdali identický kód. V každom prípade budeme nastaveniu priebežných písomných testov venovať našu pozornosť aj do budúcnosti.

Možnosť vzájomných diskusií v diskusnom fóre študenti hodnotili pozitívne: „diskusné fórum mi pomohlo, keď som si nevedela sama rady, nápady kolegov a diskusie ma myslím si posunuli dopredu“, „Vynikajúca pomôcka, nie je čo vytknúť. Oceňujem rýchlu spätnú väzbu vyučujúceho.“ Prekvapilo nás však množstvo príspevkov, ktoré sa v diskusií objavilo. Celkovo sa počas semestra vo fóre objavilo 86 diskusných tém so 629 príspevkami. Potešilo nás, že príspevky neboli len typu „toto neviem, pomôžte mi“, ale aj typu „toto som vyriešil takto, ako ste to vyriešili vy?“ (Obrázok 4). Do diskusií sme sa zapojili aj my učiteľia, aby sme študentov usmernili, upozornili ich na problematiku časti ich riešení apod.



priemerne_teply_jupyter
autor [redacted] - Saturday, 23. May 2020, 22:38

Mohla by byť tretia verzia vypocitaj_priemerne_teply takto? Nie som si istá, či som správne pochopila, keď nie sú kompletne teploty za všetky dni, počíta to bez toho dňa s nekompletnými údajmi al. nepočíta vôbec a vyhodí chybu?

```
def vypocitaj_priemerne_teply3(teploty):  
    pocet_teplo=len(teploty)  
    teploty_mimo=pocet_teplo%3  
    if pocet_teplo%3==0:  
        priemerne_teply=[sum(teploty[i:i+3],teploty[i+2  
    else:  
        priemerne_teply=[sum(teploty[i:i+3],teploty[i+2  
    return priemerne_teply
```

print(vypocitaj_priemerne_teply3([15, 25, 14, 17, 28, 2
< [redacted] >

[Permalink](#) | [Upraviť](#) | [Odstrániť](#) | [Odpovedať](#)

Re: priemerne_teply_jupyter
autor [redacted] - Sunday, 24. May 2020, 09:07

Ja to mám podobne [redacted] len ja tú generátorovú notáciu nevyužívam. Klasika je pre mňa priehľadnejšia.

```
def vypocitaj_priemerne_teply(teploty):  
    priemerne_teply = []
```

Obrázok 4: Príspevok študenta v diskusií a časť reakcie iného študenta.

Koncept dvoch učiteľov, ktorý sme preniesli z prezenčnej výučby nemal v dištančnej výučbe rovnaký efekt. Potvrdili nám to aj študenti: „Prítomnosť dvoch učiteľov na prezenčnej forme bola

fajn, jeden vysvetľuje jeden chodí pomedzi nás, pri dištančnej forme sa to už stráca účinku.“. Sledovať hlavný (priebeh cvičenia a audio komentár učiteľa a spolužiakov) a vedľajší (textový čít s druhým učiteľom) komunikačný kanál súčasne je prakticky nemožné.

V hodnotení výučby študenti ocenili aj zbierku úloh (i keď niektorí by privítali aj riešenia úloh) a zverejnenie video záznamu stretnutí: „Zverejňovanie videozáznamov je super. Len škoda, že tam ma často napadli nejaké otázky, ktoré som už nemala komu položiť.“ :-), „Veľmi pozitívne toto hodnotím. Nakoľko sa človek občas nevyhol výpadku internetu, prúdu, či rušivých faktorov v pozadí“ alebo „Veľakrát som si pozeral prednášky, na veci ktorým som nie celkom rozumel alebo v tej kope informácií to človek v tú chvíľu nevie ani toľko spracovať. Potom keď si to pozrie druhý alebo aj tretíkrát je to už jasnejšie.“

Vo všeobecnosti realizáciu predmetu v daných podmienkach hodnotili študenti pozitívne. Podobne ako my, dištančné vzdelávanie však považujú za plán B, keď plán A (prezenčná výučba) nevyšiel. Model obrátenej výučby, kde časť zodpovednosti za učenie sa prenášame na študenta, vo väčšine prípadov prijali študenti kladne. Zvlášť pozitívne boli hodnotené vopred pripravené učebné zdroje, konceptuálne testy a formatívne hodnotenie. Vyskytujúci sa opačný názor však poukazuje na fakt, že tento model nie je pre všetkých študentov prirodzený a postupne si naň musia zvyknúť.

Z celkového počtu 12 študentov, predmet úspešne absolvovalo 10. Dvaja študenti si predmet opakovane zapísali.

5 ZÁVER

Kurz programovania považujeme za veľmi náročný pre študentov RŠI, obzvlášť v podmienkach vynúteného dištančného vzdelávania. Pre nás ako didaktikov informatiky bolo a je stále veľkou výzvou pripraviť a realizovať čo najkvalitnejší kurz programovania. Táto skúsenosť s obrátenou výučbou nám ukázala jej výhody, ale aj jej úskalia. Výsledky konceptuálnych testov na začiatku cvičení rýchlo odhalili nepripravenosť niektorých študentov, ktorí si vopred nepreštudovali úvod do danej problematiky. Online komunikácia mala oproti prezenčnej výučbe viaceré obmedzenia. Náročná pre učiteľov bola príprava interaktívnych učebných materiálov vo formáte Jupyter Notebook. Povzbudili nás pozitívne reakcie študentov na obrátenú výučbu programovania. Študenti okrem získania základných vedomostí a zručností pri programovaní v Pythone zažili na vlastnej koži nové prístupy k výučbe ruka v ruku s novými technológiami, ktoré budú môcť využiť pri vlastnej výučbe informatiky. Za silné stránky obrátenej výučby považujeme efektívnejšie využitie času na cvičeniach, kde sa môžeme sústrediť na zlepšenie porozumenia učiva, vysvetlenie náročnejších pojmov, riešenie náročnejších úloh a na vyššiu aktivitu študentov. Dôležitou súčasťou obrátenej výučby je aj domáca príprava študentov a jej manažment (vlastné návrhy riešení úloh, diskusie, príprava na nasledujúce cvičenie).

Povzbudení skúsenosťami a zisteniami z tejto výučby uvažujeme o implementovaní obrátenej výučby aj v ďalších predmetoch vysokoškolského štúdia a v kurzoch ďalšieho vzdelávania učiteľov.

POĎAKOVANIE

Tento článok bol vytvorený v rámci projektu KEGA 029UKF-4/2018 Inovatívne metódy vo výučbe programovania v príprave učiteľov a IT odborníkov.

BIBLIOGRAFICKÉ ODKAZY

- [1] PAŇKOVÁ, E., HANČ, J., ŠTRAUCH, P. Rola učiteľa v obrátenej výučbe. *Edukácia: vedecko-odborný časopis*. 2017, 2(1), 202-214. ISSN 1339-8725.
- [2] THE LIFELONG KINDERGARTEN GROUP AT MIT MEDIA LAB. *Scratch: Imagine, Program, Share* [online]. [cit. 2021-01-13]. Dostupné z: <https://scratch.mit.edu/>
- [3] THE PYTHON SOFTWARE FOUNDATION. *Python.org: The official home of the Python Programming Language* [online]. [cit. 2021-01-13]. Dostupné z: <https://www.python.org/>
- [4] THE JUPYTER PROJECT. *Jupyter Project* [online]. [cit. 2021-01-13]. Dostupné z: <https://jupyter.org/>
- [5] THE MOODLE PROJECT. *Moodle - Open-source learning platform: Moodle.org* [online]. [cit. 2021-01-13]. Dostupné z: <https://moodle.org/>
- [6] BIGBLUEBUTTON INC. *BigBlueButton - Open Source Web Conferencing* [online]. [cit. 2021-01-13]. Dostupné z: <https://bigbluebutton.org/>
- [7] BERGMANN, J., SAMS, A. *Flipped Learning: Gateway to Student Engagement*. Eugene, Oregon; Washington, DC: International Society for Technology in Education, 2014. ISBN 978-1-56484-344-9.
- [8] GLAZER, F.S. *Blended Learning: Across the Disciplines, Across the Academy (Higher Education)*. Stylus Publishing; Illustrated edition, 2011. ISBN 978-1579223243.
- [9] THE FLIPPED LEARNING NETWORK (FLN). *What Is Flipped Learning?: The Four Pillars of F-L-I-P™* [online]. 2014 [cit. 2021-01-14]. Dostupné z: https://flippedlearning.org/wp-content/uploads/2016/07/FLIP_handout_FNL_Web.pdf
- [10] MAZUR, E. *Peer Instruction: A User's Manual*. Pearson, 1996, 272 s. ISBN 978-0135654415.
- [11] EBERLY CENTER, CARNEGIE MELLON UNIVERSITY. *Using Concept Tests* [online]. [cit. 2021-01-14]. Dostupné z: <https://www.cmu.edu/teaching/assessment/assesslearning/conceptTests.html>
- [12] *GREENLIGHT 2.0 - BIGBLUEBUTTON* [online]. [cit. 2021-01-14]. Dostupné z: <https://bigbluebutton.org/2018/07/09/greenlight-2-0/>

Učebné materiály podporujúce humor a zábavu Learning Materials Supporting Humour and Fun

Jozef Hvorecký

Vysoká škola technická a ekonomická
Okružní 517/10
370 01 České Budějovice
Czech Republic
hvorecky@mail.vstecb.cz

ABSTRACT

True motivation of pre-service teachers forms a basis for their future success in education. Attractive and interesting two-way communication represents a tool for achieving it.

In our paper, we discuss ways in which humor and fun can support learners' attention. We are focused on inter-disciplinary oriented textbooks and teaching aids. Realize that many jokes exploit unexpected relationships. These relationships can be used as starting points for disclosing hidden faces of events and phenomena. Good teachers will point their students' attention to them and exploit them as a launching pad for their incoming explanation.

These efforts must be supported through educational material which must be an integral element of active-learning methodology. In our opinion, the key component of such materials is a systematic but not tedious writing style which motivates its reader to learn something new. Another important component is an attractive layout.

In chapter 3, we advise their authors how to organize their teaching materials to achieve desired learning objectives.

Keywords

Authorship of teaching aids. Motivation. Humor as a component of motivation. Inter-disciplinary relations. Classroom climate. Active learning. Student-centered learning.

ABSTRAKT

Správna motivácia žiakov je základom úspechu vo vzdelávaní. Cestou, ako ju dosiahnuť, príťažlivá obojstranná komunikácia.

V tomto článku sa venujeme humoru a zábave ako potenciálnym nástrojom vytvárania a udržiavania pozornosti učiacich sa. Sústreďme sa predovšetkým na problematiku tvorby medzipredmetovo zameraných učebných textov a materiálov. Vtipy totiž často využívajú nečakané súvislosti, vďaka ktorým je možné uviesť si neviditeľnú stránku dejov. To môže využiť učiteľ, upozorniť na ňu a vzápätí ju využiť ako propedeutického prvku v nasledujúcom vysvetľovaní látky.

Za kľúčové prvky takto zostavených učebných materiálov považujeme systematický, ale nie suchopárny štýl písania, a jeho inovatívne podávanie, ktoré musí motivovať čitateľa, aby sa snažil dozvedieť sa niečo nové. Podstatná je aj príťažlivá úprava textu.

Návody na to, ako to dosiahnuť, tvoria jadro článku a sú koncentrované v 3. kapitole.

Kľúčové slová

Tvorba učebníc a učebných pomôcok. Motivácia. Humor ako motivačný prvok. Medzipredmetové vzťahy. Klíma triedy. Aktívne vzdelávanie. Vzdelávanie orientované na študenta.

1 ÚVOD

Predstavte si nasledujúcu otázku:

Zajac je:

- zvíera,
- mrkvu.

Zaradili by ste ju – alebo jej podobnú úlohu – do svojho testu? Prečo áno alebo prečo nie?

Formulácia úlohy naznačuje, prečo je humor dôležitý: vyžaduje od poslucháča pozornosť a núti ho zaujať postoj. Či si to poslucháč uvedomuje alebo nie, vtip ho nasmeruje ku kritickému mysleniu.

Tabuľka 1: Mylné predstavy o použití humoru vo vyučovaní

Najčastejšie mylné predstavy	Protiargumenty (ako ich vidí moderná pedagogika)
Humor sa stotožňuje s rozprávaním vtipov	Humor má veľa podôb. Vo vyučovaní sa najčastejšie používa situačný humor s cieľom uvoľniť napätie. Môže to byť aj hovorený vtip, ale veľa dokáže aj gesto, kontext zadanej úlohy, ...
Učitelia by nemali používať humor vo vyučovaní, pretože na to nemajú primeraný tréning	Je pravda, že učitelia nemajú tréning v tejto oblasti. Ale na to, aby niekto použil humor, nemusí byť cirkusový umelec ani komediálny herec. Humor predsa využívame v bežnom živote. Chce to ešte jeden krok a použiť ho aj vo vyučovaní.
Učitelia by nemali používať humor vo vyučovaní, pretože na tom, čo sprostredkujú, nie je nič smiešne	Humor vo vyučovaní nezosmiešňuje predmet, ktorý učiteľ učí. Je to spôsob, ako ho komunikuje.
Použitie humoru vo vyučovaní je márne drahocenného času, ktorý má učiteľ v triede k dispozícii	Použitie humoru je prinajmenšom efektívnym nástrojom na upútanie a udržanie pozornosti. Pozitívne emócie sú významným podporným faktorom ovplyvňujúcim rozsah a trvácnosť zapamätania

Zdroj: [1]

Ako konštatuje Haverlíková v [1], je chybou, ak sa problematika humoru vo vzdelávaní neobjavuje ako súčasť prípravy budúcich učiteľov.

Kurzy humoru však na univerzitách existujú, hoci sú zriedkavé. Ako píše prof. Roth, autor jedného z nich [2], pri jeho zavádzaní sa

musel prebiť cez mnoho prekážok univerzitetnej byrokracie. Niektoré zo svojich reakcií dnes úspešne využíva v učebnej látke, pretože na protiargumenty odporcov vzdelávania v oblasti humoru odpovedal čarováním. Niektoré jeho odpovede boli tvrdé, až arogantné, čo však ich vtípnosť vhodne zastierala. Narúšal nimi „dôstojný priebeh“ diskusie a výhrady voči potrebnosti zaradenia prednášky o humore do študijného programu. Napríklad na otázku, či má daný predmet ekvivalent na iných univerzitách odpovedal, že síce nepozná žiadnu, ale je presvedčený, že niekde vo vesmíre určite existuje. Keď prídu mimozemšťania, nepochybne sa potešia, že ich prednáška má ekvivalent. Svojimi odpoveďami ukázal, že humor a zábava môže slúžiť aj ako prostriedok boja proti predpojatosti a zastaranému mysleniu. Roth [2] vo svojom článku zároveň uvádza, že humor posilňuje kreatívne myslenie, pomáha odhaľovať nelogickosti, zvyšuje súdržnosť skupiny, uvoľňuje stres a napätie, motivuje študentov, posilňuje ich pozornosť, vytvára klímu triedy a pomáha vytvárať vzťah k pedagógovi.

Ako vidieť, hoci je dostatok dôvodov na posilnenie prítomnosti humoru vo vzdelávaní, časté sú aj hlasy proti. Tabuľka 1 obsahuje najčastejšie argumenty odporcov, ako aj protiargumenty zástancov [1]. Ak chceme, aby v školách vznikla priaznivá klíma pre uplatnenie humoru počas vzdelávania, treba, aby ju systematicky rozvíjali nielen učители. Je žiaduce, aby sa mali o čo oprieť vo svojom úsilí: jednak by vedenie školy malo podporovať ich snaženie, jednak by mali mať k dispozícii nástroje, ktoré využijú počas práce v triede a mimo nej.

V tomto príspevku sa zameriame na to, ako k tomu môžu prispieť vhodné učebné materiály. Ukážeme aj tipy na ich vytváranie a zásady, ktorých by sa mali tvorcovia pridržovať.

2 HUMOR A ZÁBAVA PROSTREDNÍCTVOM UČEBNÝCH MATERIÁLOV

S humorom sa v našich učebniciach stretávame zriedkavo. Autor nepozná takú, v ktorej by sa vyskytol vtíp. Dokonca aj populárno-vedecká literatúra býva v tomto smere pomerne bezzubá. Akoby autori prevzali z tabuľky 1 názor, že používať humor nemôžu, pretože na tom, čo sprostredkujú, nie je nič smiešne. Zrejme aj preto sú populárno-vedecké publikácie často rovnako suchopárne ako učebnice analogických predmetov.

Z motivačných pomôcok, ktoré vysvetľujú učebnú látku inak, sa najčastejšie stretáme s vedeckými hračkami [3], t. j. objektmi umožňujúcimi podávať učebnú látku, jej pojmy a vzťahy medzi nimi v názornej podobe. Najčastejšie sú to objekty, ktorými môže žiak manipulovať. To, že môže experimentovať, zvyšuje pravdepodobnosť pochopenia princípu.

Vedecké hračky práve svojou konkrétnosťou stavajú istú bariéru zovšeobecneniu daného poznatku, tomu, aby sa takto získaný poznatok tvorivo rozvíjal ďalej v hlave učiaceho sa. Práve preto by menej tradičné formy vzdelávania (vrátane humoru) mali smerovať k zovšeobecniam. Mali by byť postavené ako otvorené – bez konečného, predurčeného výsledku. Treba, aby nový (hoci aj čiastkový) poznatok zapálil iskru, ktorú povedie ku kreativite, do istej miery nezávislej od tohto konkrétneho poznatku. Dobrá vedecká hračka je tá, ktorá práve toto evokuje. Cieľom je teda popri vzdelávacích prístupoch s pevne stanoveným vzdelávacím cieľom vytvárať aj paralelné, alternatívne, ktoré budú od dieťaťa vyžadovať aj vlastný prínos. Aj na Slovensku existujú a majú rôzne podoby, ako ukazujú nasledujúce príklady:

- súťaž *Schola Ludus: Obrázkový vedecký vtíp*, uskutočňovaná v rámci Týždňa vedy a techniky [4],
- komiks s matematickými úlohami [5],
- *Malované čítanie* (napr. [6]), v ktorom sú niektoré slová nahradené obrázkami,
- pracovný zošit zameraný na vzťahy medzi fyzikou, matematikou a slovenčinou [7].

Vzhľadom na pozitívny význam, aký má humor vo vzdelávaní (pozri napr. [8], [9]), je ich stále málo. Navyše sa zdá, že súťaž [4] skončila v r. 2015.

Malované čítanie predstavuje „seriálový“ typ mimoučebnicovej literatúry, v danom prípade zameraný na rozvoj čítania prvákov. Práve efektívnosť série podobne orientovaných kníh pre vydavateľa naznačuje, že takýto prístup má kladnú odozvu aj u detí a ich rodičov. V Českej republike, kde sa trh s učebnicami otvoril dávnejšie, vidieť v tejto oblasti oveľa vyššiu variabilitu. Hlavnou postavou série pracovných zošitov [10, 11, 12] je psí detektív Komisar Vrtápek. Samotná myšlienka predstavuje zrejme paródiu na televízny seriál Komisar Rex. Pracovné zošity obsahujú zaujímavé logické úlohy, ktoré majú deti riešiť v spolupráci s rodičmi. Využívajú varianty určitej, pevne zvolenej schémy tak, ako je to pre seriály typické. Tým zároveň naznačujú rodičom, čo od nasledujúceho dielu očakávať. Nápaditosť zadania zaručuje, že riešenia nebudú detského čitateľa nudiť a že aj rodič v nich nájde v nich skrytý humor, ktorý – vzhľadom na svoje menšie životné skúsenosti – dieťa nemusí postrehnúť. Ako príklad uvádzam niektoré z mien stoviek postáv: učiteľ Rákoska, obchodník Držgroš, či krčmár Štampřík. Viacúrovňový humor zabezpečuje, že keď bude rodič riešiť úlohy s dieťaťom, bude sa zabávať spolu s ním – hoci objekt zábavy nebude vždy totožný.

Uvedené pozorovania nám dovoľujú vytipovať základné črty podporných učebných materiálov, ktoré povedú dieťa k pochopeniu učebnej látky a umožnia učiteľom efektívne využívať ich zábavnejšie stránky:

- Systematický, ale nie suchopárny štýl,
- Inovatívne, zábavné podávanie poznatkov,
- Podpora motivácie dozvedieť sa niečo nové,
- Príťažlivá úprava.

Všimnite si, že dôraz kladieme na „pochopenie“, nie na „zvládnutie“ učebnej látky. Prirovnajme tento proces k činnosti vysokohorského nosiča. Odrastenejšie dieťa by zrejme *zvládlo* na príkaz dospelého vyniesť desaťkilový náklad na tatranskú chatu. *Pochopilo* by však zmysel a význam vynúteného počínania?

Rovnako sa treba dívať na „zvládnutie učebnej látky“. „Deti to zvládnu“ je bežným argumentom pri zahusťovaní učebných osnov. Ako naznačil predchádzajúci príklad, zvládnuť znamená zavišit výkon aj napriek nechuti. To nemusí stačiť na pochopenie zmyslu počínania. Cieľom vzdelávania teda nemôže byť „podávať výkony“. Kto nepochopí význam vzdelávania pre ďalší život, nebude ochotný (a možno ani schopný) vzdelávať sa v budúcnosti. Preto si treba položiť otázku, ako by mali vyzerat všetky učebné materiály, ak ich cieľom nie je zvládnutie látky, ale jej pochopenie: aký štýl podávania obsahu zvolit, ako ho zábavnejšie prezentovať, ako podporiť záujem dozvedieť sa viac a ako posilniť tieto ciele príťažlivou pôsobiacou úpravou učebných materiálov a pomôcok. Pri hľadaní odpovedí budeme vychádzať z povahy štyroch vyššie uvedených charakteristík.

3 ZÁBAVNEJŠIE VZDELÁVANIE

3.1 Systematický, ale nie suchopárny štýl

Dôvody suchopárneho, až nudného štýlu učebníc sú jednoduché. Veľa autorov pochádza z vysokých škôl. Kvalita ich akademickej činnosti sa hodnotí podľa ich počtu vedeckých publikácií [13]. Preto je ich primárnym cieľom dokonale zvládnuť spôsob písania vyznačujúci sa stručnosťou a rigoróznosťou. To je vo vedeckých článkoch potrebné a pochopiteľné, lebo adresáti sú špecialisti v danej oblasti, oboznámení s témou a jej kontextom. Netreba ich uvádzať do problematiky ani motivovať – keby neboli v príslušnej oblasti vzdelaní a motivovaní, článok by zďaleka obišli. (Aj autor to robí s článkami zo vzdialenejších vedných odborov a domnieva sa, že nie je sám). Autori, ktorí nie sú z vysokých škôl, sa snažia ich štýl napodobniť. Inak by takmer s istotou narazili pri recenznom pokračovaní, pretože najmenej jeden z budúcich recenzentov pracuje na VŠ.

Žiaci a študenti však nie sú partnermi tohto typu – sú to laici. Podávaná látka je pre nich nová, neznáma. Učebné materiály by mali vyvolať záujem o ňu. To sotva zvládne takýto text [14, s.150]: „*Podradený člen rozširuje alebo zužuje význam nadradeného člena a teda s ním zrastá nielen vo forme, ale aj obsahom. To sa odráža aj na ustálenom slovoslede členov tohto skladu. Narušenie ustáleného slovosledu prináša zmenu jeho štylistickej hodnoty, a ak nie je funkčne opodstatnené, považuje sa za jazykový i štylistický nedostatok.*“ Porozumie tomuto študent priemyslovky, ktorému je určený? (Nevraviac o tom, že túto vedomosť sotva uplatní v praxi.)

Opatrnosť pri tvorbe učebných materiálov je nevyhnutná aj kvôli tomu, že vzhľadom na počet predmetov je fakticky vylúčené, aby každý žiak mal o všetky predmety rovnaký – navyše vysoký – záujem. Treba ich motivovať [15] tak, aby ich cieľom nebol iba výkon, čiže spraviť skúšku alebo napísať písomku. Pre tých, ktorí nie sú nadšencami danej oblasti, dokážu motiváciu nabudiť len primerane jednoduché študijné materiály. Ich cieľom nemá byť dosiahnuť v mysli každého žiaka vedecky neotrasiteľné poznatky, ale pomôcť mu orientovať sa v pojmoch a súvislostiach vyučovaných predmetov. Cieľom všeobecného vzdelávania by mal byť absolvent schopný rozlišovať ich „definíčné obory“. Inak by mal v budúcnosti problém odlišiť matematický problém od chemického, keď sa bude chcieť obrátiť o radu na príslušného špecialistu. Je teda jasné, že písať učebnice pseudovedeckým štýlom je riziko, ktoré sa vypomstí na malom záujme o vyučovaný predmet.

To neznamená, že istý stupeň nezrozumiteľnosti („záhadnosti“) nie je prípustný. V rukopise [16] sa vyskytuje nasledujúci nezmyselný text. Je tam úmyselne, jeho zámerom je vysvetliť jadro úlohy:

Pteer a Onderj sú dvojjiký. Lduia si ich čtaso myíla. Onredj sa niekdey čujdue: „Ako si nás mžôu milýť? Mne sa to ndiky nesatlo.“

Textu ste zrejme porozumeli. Psychológovia [17] preukázali, že človek dokáže chápať slová, v ktorých sú iba prvé a posledné písmená na svojich miestach, kým zvyšné sú poprehadzované. Vďaka tomu je možné takýto text prezentovať na hodine matematiky a vzápätí položiť otázku: *Koľkými spôsobmi vieme zapísať dané slovo tak, aby zostalo – vo vyššie špecifikovanom zmysle – zrozumiteľné?*

Núkajú sa dve cesty:

- Uviest' vzorec a zadať úlohy, ktoré sa ním dajú riešiť,
- Postupne prísť k návodu na riešenie, pomocou ktorého budú schopní sami nájsť výsledok pre každé slovo bez toho, aby sa vzorec dozvedeli.

Začneme vzorcom, lebo jeho tvar je dostatočne odstrašujúci na to, aby odradil každého laika.

$$Z = \frac{N!}{\prod_{i=1}^k (r_i)!}$$

Vo vzorci:

- Z je počet „zrozumiteľných“ permutácií,
- N je počet vnútorných písmen,
- k je počet rôznych vnútorných písmen,
- r_i je počet výskytov i -teho vnútorného písmena.

V duchu „tradičného“ prístupu k vyučovaniu matematiky by vzápätí nasledovali úlohy typu: *Aký počet zrozumiteľných permutácií slova konkurencieschopnosť?*

Ak ste vzorec ignorovali a okamžite preskočili zrakom až sem, spravili ste správne. Pre vzdelávanie je užitočnejšie postupné odhaľovanie vedúcich k výpočtu.

V prvom kroku sa sústreďme iba na slová, v ktorých sú všetky vnútorné písmená rôzne. Pri analýze je rozumné začínať s kratšími slovami, v ktorých sa vnútorné písmená neopakujú, napríklad kotva, hnev, báseň. Žiaci sami a rýchlo nájdu všetky permutácie s požadovanou vlastnosťou. Až potom má zmysel začať sa zaujímať o ich počet, hlavne kvôli kontrole. Pri väčších počtoch permutácií sa vinou nepozornosti niektorá zámena vyskytne viackrát alebo vôbec nie. Žiaci časom zistia, že ak sa má podoba slova meniť, musí mať najmenej štyri písmená. Trojpísmenové slová majú iba jedno vnútorné písmeno, takže jeho umiestnenie v slove sa zmeniť nedá. Pri hľadaní odpovede na otázku, koľko ich je, sa skôr-neskôr dopracujú k súčinu $N \times (N-1) \times (N-2) \dots \times 3 \times 2 \times 1$, teda k faktoriálu čísla N . Ten sa skrátene zapisuje $N!$. Teda slovo s N rôznymi vnútornými písmenami, má $N!$ rôznych „čitateľných“ verzii. (To je čitateľ vo vyššie uvedenom zlomku, čiže sme ešte dosť ďaleko od finálneho výsledku.)

Tých, ktorých nebaví matematika, môže učiteľ poveriť hľadaním čo najdlhšieho slova, v ktorom sa neopakujú vnútorné písmená. S ostatnými môže v druhej etape začať skúmať, koľko permutácií majú slová s opakujúcimi sa vnútornými písmenami. Užitočné je rozčleniť triedu na skupiny. Každá bude vyhľadávať krátke slová, v ktorých sa opakujú vnútorné písmená a zisťovať počty ich permutácií. Svoje zistenia si budú navzájom oznamovať a vkladat' ich do tabuľky podobnej Tabuľke 2.

Tabuľka 2: Počet permutácií slov s opakujúcimi sa písmenami

Slovo	Počet vnútorných písmen	Opakuje sa	Faktoriál	Permutácie
Círn	2	2 R	2	1
Motor	3	2 O	6	3
Živica	4	2 I	24	12
Prorok	4	2 R + 2 O	24	6
Poloos	4	3 O	24	4
Kredenc	5	2 E	120	60
Prezrel	5	2 R + 2 E	120	30
Kolotoč	6	3 O	720	240

Zdroj: [15]

Dva stĺpce vpravo sú vodidlom pri hľadaní vzťahu. Odporúčame začať prípadmi, v ktorých sa opakuje iba jedno písmeno. Vtedy je počet permutácií polovicou hodnoty faktoriálu. To sa dá ľahko vysvetliť „farbením“ daného písmena. Keď napríklad každé z písmen O v slove motor zafarbíme inou farbou, permutácií, pri ktorých bude O na každom mieste inak zafarbené bude 24. Keď im vrátime pôvodnú čiernu farbu, dvojice splynú.

Nežiadajte od žiakov, aby vám bezhranične verili. Musia si to vyskúšať – navyše na rôznych slovách s opakujúcimi sa vnútornými písmenami. Najprv s dvomi, potom tromi. Nechajte ich vymýšľať takéto slová, prispějete tým k rozvoju ich slovnej zásoby.

Vďaka farbeniu opakujúcich sa písmen (a s príležitostným návodom od učiteľa) prídu na to, že pôvodný počet treba vydeliť faktoriálom výskytov opakujúceho sa písmena – každého zvlášť. Čo je vlastne to, čo vyjadruje predchádzajúci vzorec. Cieľom však neboli (a nemá byť) vzorec, ale spôsob výpočtu.

Samotný proces hľadania pritom nemusí byť priamočiary a neprerušovaný. Učiteľ môže behom neho robiť „odskoky“ t. j. prerušiť riešenie a sa venovať jeho vedľajším líniam. Vďaka nim klesne mentálna únava učiacich sa a vzrastie ich pozornosť. Príklady uvádzame v časti 3.3. Cieľom má byť aktívna forma vzdelávania, počas ktorej sa žiaci učia sami. Je známa ako *vzdelávanie orientované na študenta*.

3.2 Inovatívne podávanie poznatkov

Prednášanie je najčastejšou metódou podávania poznatkov na našich školách, vrátane univerzít [18]. Tok informácií počas neho je jednosmerný – od učiteľa k žiakovi. Ten je pasívnym prijímateľom. Jeho vedomosti sa preverujú porovnávaním tých, ktoré podal učiteľ, s tými, ktoré prezentuje žiak. Žiak dokáže tento rozdiel najľahšie a najrýchlejšie minimalizovať tak, že sa učiteľove výroky naučí naspamäť. Memorovanie je v takomto prostredí optimálnou stratégiou žiaka.

Každá didaktická metóda, ktorá dá žiakovi väčší priestor na samostatné získavanie poznatkov, ich rozvoj a prezentovanie, znamená prínos k modernizácii vzdelávania. Vyšší podiel žiaka na vlastnom vzdelávaní zvyšuje pravdepodobnosť, že sa jeho vedomosti upevnia a pretrvávajú [19]. Existuje kvalitatívny rozdiel v tom, či sa žiak naučí vzorec a podľa neho vyrieši pár úloh, alebo postupne vlastnou aktivitou príde na niektoré zákonitosti, ktorými sa permutácie vnútorných písmen riadia. Cieľom nemusí byť odhaliť celý vzťah, ale podieľať sa na jeho odhalení – napríklad nájdením slov s opakujúcimi sa vnútornými písmenami. Žiak s veľkou slovnou zásobou nájde povedzme štetec, meter, kajak. Toho, kto zaň bude považovať olovo, upozorní na chybu.

Žiakov, ktorí radi „počítajú pre počítanie“, môže učiteľ zamestnať zisťovaním, koľko permutácií majú slová Tennessee alebo Tatarstan. Pri takto chápanom vzdelávaní nie je cieľom kvantum vedomostí, ale potešenie z príspevku k spoločnému poznatku. Žiak, ktorý je dobrý v matematike, môže si nájsť partnera s širokou slovnou zásobou. Obidvaja sa presvedčia o dôležitosti tímovej práce. Tím môže byť väčší, dokonca ním môže byť celá trieda, odporúčané sú však troj- alebo štvorčlenné tímy [20].

Podobné tímy môžu vzniknúť vo všetkých učebných predmetoch, pričom projekty by mali simulovať reálne projekty, t. j. mali by využívať špecifické danosti a schopnosti členov. Napríklad, ak je predmetom projektu vedecká hračka, mal by byť v tíme žiak talentovaný v danej oblasti, ktorý ju navrhne a/alebo vysvetlí. K nemu však treba ďalších, ktorí sa vyjadria k tomu, či a ako jej princípom porozumeli, a budú hľadať spôsoby, ako všelijako sa dá

(správne aj nesprávne) pochopiť jej funkcia. Zachytia vlastné chyby pri práci s ňou. Budú si ich musieť uvedomiť a s pomocou ostatných ich odstrániť. Vďaka tomu sa posilní súdržnosť tímu a zlepši sa jeho vnútorná komunikácia. Ostatní členovia (a učiteľ) spoznajú rôzne štýly myslenia a tým zdokonalia aj svoje chápanie.

Rozvinutím tejto metódy je „prevrátená trieda“ [21]. V nej sa mení postavenie učiteľa a žiakov. V tradičnej triede:

- Učiteľ oboznámi žiakov s látkou,
- Žiak má následne pochopiť jej obsah a zmysel.

Keď sa na túto činnosť pozrieme z perspektívy Bloomovej taxonómie vzdelávacích cieľov (obr. 1), vidíme, že učiteľ vykonáva spodné (jednoduchšie) činnosti, kým vyššie, náročnejšie úrovne ponecháva na žiakov.



Obrázok 1: Bloomova taxonómia vzdelávacích cieľov

V prevrátenej triede je postup opačný. Žiaci sa najprv samostatne zoznámia s látkou. Potom svoje poznatky prednesú v triede a so spolužiakmi o nich diskutujú a ujasňujú si ich. Učiteľ pôsobí ako moderátor diskusie a vo vhodných, či potrebných momentoch do nej vstupuje s cieľom usmerniť ju alebo redukovať omyly a nedorozumenia. Tento spôsob výučby predpokladá prítomnosť neúplných poznatkov, čím vytvára priestor aj na humor a zábavu. Vznikajú spontánne. Zakazovať ich by bolo v protiklade s touto metodológiou vzdelávania. Priestor treba prenechať aj „kreatívnym“ riešeniam: *Ako zmeníš centimetre na metre? Škrtnem písmená centi*. Tie slúžia napríklad na iniciovanie diskusií.

3.3 Motivácia dozvedieť sa niečo nové

Zvedavosť je základom poznávania. Z toho vyplýva, že rigorózne chápanie toho, čo „má a musí byť“ obsahom daného vyučovacieho predmetu je vlastne protipedagogické. Rovnakou chybou je určovať, čo nemá byť obsahom daného predmetu. Autor tieto hranice často prekračoval (a sľubuje, že bude prekračovať). „Protipríkladom“ úlohy, v ktorej zisťujeme počet permutácií daných slov, je zostaviť text, ktorého ani jedno slovo sa nedá permutovať. Takého, v ktorom nie je ani jedno slovo dlhšie ako trojpísmenové. Tu je ukážka z rukopisu [16]:

Ján má rád Aňa, Aňa to vie. Sú pár: Ján je Lev, Aňa je Rak.

Ján ju zve: Aňa, je jún a ide nov. Pod' na pár dni do hôr!

Kam?

Mám tip – Choč, Šíp a Hrb. Na tri dni.

A čo tam?

Len tak – ja a ty, jar, les, cit a mach.

Les nie je Rim, ani bál. Len ker, trň, tis, buk a dub, sem-tam dym. Je tam aj vlk, had, výr, či myš, osy a iný hyd.

V knihe slúži na odľahčenie matematicky náročného (hoci inak neformálneho) textu. Čitateľovi dopraje chvíľu oddychu medzi obťažnejšími časťami. Väzba zadania na hlavnú líniu je voľná, jej

cieľom je zaujať čitateľa, pre ktorého matematika nie je koníčkom, ale ani nepriateľom.

Odskoky od hlavnej línie môžu byť rozmanité, mali by však tiež rozvíjať poznatkovú bázu, aj keď iným smerom. Napríklad: *Poznáš význam týchto trojpísmenových slov: alt, bas, cin, čip, dav, d'as, erb, fez, gén, hod, chrt, ión, jed, kyj, lom, mys, nit, ohm, psi, rýl, sup, šek, trs, ťah, umy, vrt, zob, žat'?* Použi aspoň tri z nich vo vete, ktorá bude mať všetky slová najviac trojpísmenové.

Žiak sa môže pýtať, kde nájde trojpísmenové slová. Samozrejme na internete [24]. Využívanie internetu netreba zakazovať. Obsahuje totiž obrovské množstvo faktov – v danom prípade zoznam gramaticky správnych krátkych slov. Úloha bude ľahšia iba zdanlivo: na žiakovi zostáva kreatívna zložka procesu – skomponovať zmysluplný text.

Na druhej strane nie je vhodné dávať zadania, ktorých riešenie sa dá opísať z internetu. Podobne ako v predchádzajúcej ukážke však existuje veľa takých, ktorých čiastočné riešenie sa dá nájsť, avšak na jeho dokončenie treba kreativitu a usilovnosť [25].

Často vzniká problém, ako rozprúdiť diskusiu, keďže žiaci na ňu nie sú zvyknutí. Trojpísmenovým slovom je aj UFO. Jeho kontroverznosť môže byť vhodným štartérom diskusie v triedach, ktoré sú inak mlkve: *Čo je UFO? Existuje alebo neexistuje?* Podľa prvotného názoru jednotlivcov učiteľ rozdelí triedu na skupiny – časť bude za, časť proti.

3.4 Príťažlivá úprava

Ako žiak postupuje z ročníka do ročníka, učebnice sa začínajú podobáť jedna na druhú. Grafika sa stáva monotónnejšou a obsah „vedeckejším“, ale fádnejším. Učebnice sú často „anonymné“, ich autori sú takmer neznámi. V prípade zle napísaných učebníc to má svoju logiku. Dobré píšuci autori by však mali byť populárni a mali by si užívať podobnú úctu ako dobrí spisovatelia.

Humor – nosná myšlienka tohto príspevku – v učebniciach takmer vždy chýba. Preto som pri písaní „pracovných zošitov“ (lepšie pomenovanie pre [7, 16] nemám) trval na tom, aby aj ilustrácie mali humorný nádych. Pred plynulým popisným textom som uprednostnil sokratovský dialóg.

Podobne ako prevrátenej triede, sokratovský dialóg vychádza z malej znalosti, či dokonca neznalosti predmetu, ktorý je objektom skúmania. Jeho spoznanie prichádza cez sériu otázok a odpovedí, ktoré si kladú diskutujúci. V [7, 16] sú iba dve postavy: autor Jožo a meňavec Améb. Améb existuje od veľkého tresku, kým Jožo len nepatrný zlomok tejto doby. Améb je očitý svedok – videl celý vývoj vesmíru. Jožo mu však dokáže vyjasniť, prečo sa vývoj dial tak, ako sa dial. Dejová línia však nie je priamočiara. Napríklad vo chvíli, keď sa rozprávajú o kvarkoch, vznikne otázka, prečo sa vo svete píšú kvarky s Q, len u nás s KV. Pretože sokratovský dialóg možno rozvíjať ľubovoľným smerom, Améb konštatuje: „Keby som bol písmeno Q, radšej ani neprídem na Slovensko, aby ma neprepísali“. Výsledkom je spoločný text, v ktorom nahrádza dvojicu KV písmenom Q. Tu je ukážka [7]: „*Ludia v Mosqe majú radi qasené uhorky, mrqu, redqičky a teqicu. Qasia ich v slanej qapaline, v ktorej je qantum vody s qasným octom a qapkou citrónovej štavu.*“

Aj ilustrácie by mali byť rešpektovať princípy dialógu. Nemali byť zostať iba mechanickým vyjadrením textu prostredníctvom grafiky. Aj ilustrátor musí dostať voľnosť. Pokiaľ jeho ilustrácie neprotirečia odbornej povahe predkladanej látky, mal by mať príležitosť nakresliť „čokoľvek“. Matúš Vizár to úspešne zvládol aj

v časti o písmene Q – vid' obr. 2. Je na ňom „titulná strana“ novín Qákoty a Qílenia.



Zdroj: [7, str. 26]

Obrázok 2: Slovenčina s Q

4 ZÁVER

Do reálnych aj virtuálnych učební treba priniesť viac humoru a zábavy. Vzdelávanie bez humoru je samo osebe na smiech. Výskum [27] ukázal, že 12 % študentov malo najradšej učiteľov, ktorí intenzívne využívali humor a 72 % malo najradšej tých, ktorí ho používali často. Iba štvrt percenta uprednostňuje učiteľov, ktorí ho nepoužívajú vôbec.

Najlepšie vtipy vznikajú spontánne ako reakcia na konkrétnu situáciu. Objavia sa však iba vtedy, keď si žiaci (a ich učiteľ) sú vedomí, že vtip padne na úrodnú pôdu, že autor vtipu bude ocenený a nie penalizovaný. Teda vtedy, keď klíma triedy/školy je pre humor priaznivá. K jej vytvoreniu môžu napomôcť učebnice a učebné materiály, ktoré žiakov aktivizujú. V neformálnej pracovnej atmosfére vznikajú vtipné situácie ľahšie ako v triede, kde sa všetko podriaďuje náladám jednotlivca.

Ako sme naznačili vyššie, situácia autorov učebných materiálov nie je optimálna. Chýba ich nezávislé odborné posudzovanie. Ak existuje, má skôr sporadický charakter [28], pričom situácia v Česku je o niečo lepšia ako na Slovensku. O situácii u nás píše Juščáková [29]: „*Vzhľadom k postaveniu učebnice v našej edukačnej praxi prekvapuje súčasný nízky záujem slovenskej pedagogickej vedy o teóriu tvorby učebnice a teóriu hodnotenia kvality učebníc. Neexistuje pracovisko, ktoré by sa zaoberalo výskumom učebníc porovnateľne so zahraničím.*“ Za týchto okolností je pochopiteľné, že hoci autori postupujú „podľa najlepšieho vedomia a svedomia“, ich úsilie nemusí viesť k optimálnemu výsledku.

Osobitne ťažko sa v takomto prostredí prekonávajú stereotypy o tom, ako má učebnica vyzerať. S tým má autor zaujímavú skúsenosť. Pretože učebnica, ktorá má byť preplácaná ministerstvom školstva, musí byť recenzovaná, vydavateľstvo RAABE poslalo [7] na posúdenie. Z pracoviska, ktoré radšej nebudem menovať, prišiel posudok v duchu: „*Takto sa učebnice nepíšu*“. Nad takto formulovaným dôvodom nesúhlasu zostal autor v šoku. Samozrejme, ak má byť učebnica pôvodná, musí byť iná ako predošlá. Načo písať takú, ktorá je veľmi podobná, či dokonca zhodná s predchádzajúcimi?

Všimnite si, že recenzent(i) nenamietali proti obsahu alebo chybám v ňom. To, proti čomu brojili, bola *inakosť*. Za týchto okolností sa nečudujeme, že jedným z najčastejších prehrškov v akademickom prostredí je plagiátorstvo [30]. Veď *inakosť* je vlastne nedostatkom – zostať v zajatí stereotypu sa stáva prednosťou.

V dôsledku toho je autor veľmi skeptický v názore na masovú zmenu podoby učebníc a učebných pomôcok v dohľadnom čase. Ak by však niekto chcel vedieť, ako všetkým sa dá humor implementovať do vzdelávania, nech si otvorí [31].

POĎAKOVANIE

Autor ďakuje všetkým, ktorí vydržali čítať článok až do konca.

BIBLIOGRAFICKÉ ODKAZY

- [1] Haverlíková, V. *Vedecký vtíp ako prostriedok vzdelávania*. In: Tvorivý učiteľ fyziky VII, Smolenice, 2014, s. 101-106
- [2] Roth, G. L. *Confessions from a guy who teaches humor* [Perspectives on Teaching]. New Horizons in Adult Education and Human Resource Development, 22(2), 2008, s. 55-62. <http://education.fiu.edu/newhorizons>
- [3] Teplanová, K. *Ako transformovať vzdelávanie: Stratégie a nástroje SCHOLA LUDUS na komplexné a tvorivé poznávanie a učenie*. Bratislava : Metodicko-pedagogické centrum, 2007. 120 s.
- [4] Súťaž Schola Ludus: Obrázkový kreslený vtíp. <http://www.scholaludus.sk/new/ofv/ovv2015.html>
- [5] Uhrín, M., Bero, P., Berová, Z. *Komiksová školička*. Bratislava, Pictures Talking, 2020, 60 s. [Komiksova skolicka \(liberaterra.sk\)](http://www.komiksova.skolicka.liberaterra.sk)
- [6] Dienerová, A., Kováčová, E. *Maľované čítanie: Opice nezbednice a iné príbehy*. Bratislava: IKAR, 2019, 72 s.
- [7] Hvorecký, J. *Meňavce: Veľký tresk*. Bratislava: Raabe, 2018, 98 s.
- [8] Gogová, L. *Humor ako súčasť sociálnej klímy v triede*. In: Gogová, L. (ed.) *Kľúčové kompetencie pre celoživotné vzdelávanie*, Prešov: Centrum celoživotného a kompetenčného vzdelávania Prešovskej univerzity, s. 18-26
- [9] Gogová, L. *Pozitívne aspekty humoru*. In: Gogová, L. (ed.) *Kľúčové kompetencie pre celoživotné vzdelávanie*: Centrum celoživotného a kompetenčného vzdelávania Prešovskej univerzity, s. 19-28
- [10] Strychová, P., Morkes, P. *Komisař Vrtáпка: Sebrané spisy 1*. Praha: Mladá Fronta, 2007, 112 s.
- [11] Morkes, P., Strychová, P., Strycha, T. *Komisař Vrtáпка: Sebrané spisy 2*. Praha: Mladá Fronta, 2009, 112 s.
- [12] Morkes, P., Strychová, P., Strycha, T. *Komisař Vrtáпка: Sebrané spisy 3*. Praha: Mladá Fronta, 2011, 112 s.
- [13] Slovenská akreditačná agentúra pre vysoké školstvo: *Štandardy pre vnútorný systém zabezpečovania kvality vysokých škôl*. Bratislava: SAAVŠ, 2020, 13 s., <https://saavs.sk/wp-content/uploads/2020/09/Standardy-pre-vnutorny-system-zabezpecovania-kvality-2.pdf?x74963>
- [14] Hincová, K., Húsková, A., *Slovenský jazyk pre 1.-4. ročník stredných škôl*, Bratislava: SPN, 2009, 368 s.
- [15] Petlák, E. *Motivácia v edukačnom procese*. Bratislava: Wolters Kluwer, 2019, 104 s.
- [16] Hvorecký, J. *Meňavce: Veľké čísla*. (rukopis)
- [17] Grainger, J., Whitley, C. *Does the human mind read words as a whole?* TRENDS in Cognitive Sciences Vol.8 No.2 February 2004, s. 58-59
- [18] R. Hall et al, *Odporúčanie pre zlepšenie školstva na Slovensku - To dá rozum*. Bratislava: MESA10, 2020. [Analýza zistení o stave školstva na Slovensku - To Dá Rozum \(todarozum.sk\)](http://www.todarozum.sk)
- [19] Turek, I. *Didaktika*, Bratislava: Wolters Kluwer, 2014, 598 s.
- [20] J. Hvorecký: *Timové projekty cez Internet*. Sborník konference *Belcom '06*, Praha, ČVUT, 2006, s. 31-38
- [21] Keengwe, J. *Promoting Active Learning through the Flipped Classroom Model* Hershey, IGI Global, 2014, 355 s.
- [22] Brozmanová, M. *Uplatnenie Bloomovej taxonómie vo vyučovaní prírodovedy na I. stupni ZŠ*. Bratislava: Metodicko-pedagogické centrum, 2012, 39 s.
- [23] Nikulinová, E. *Bloomova taxonómia v dejepise*. https://www.zborovna.sk/kniznica.php?action=show_version&id=140618
- [24] Referáty.sk: *Scrabble (Platné dvojpísmenové a trojpísmenové výrazy)*. <https://referaty.centrum.sk/ostatne/nezaradene/11644/?page=1>
- [25] Gdovinová, T. *Jedenásť učiteľov ukazuje, ako sa doma hrať s deťmi. Pripravili zoznam najzaujímavejších úloh zo školských predmetov*. Denník N, 15. 1. 2021. <https://dennikn.sk/2226571/jedenast-ucitelov-ukazuje-ako-sa-doma-hrat-s-detmi-pripravili-zoznam-najzaujimavejsich-uloh-zo-skolskych-predmetov/?ref=tit>
- [26] Flachbartová, L. *Sokrates, Aischinés a sokratovský dialóg*. Filozofia, Roč. 74, 2019, č.1, s. 51-61
- [27] Check, J. F. *Humor in education*. Physical Educator, 54(3), s. 165.
- [28] Kmecová, I. *Kvalita učebníc vo výchovno-vzdelávacom procese predmetu ekonomika*. In: Dnešní Trendy Inovací 2014: 5. mezinárodní vědecká konference a mezinárodní workshop. 1. vyd. Brno: B&M InterNets, s. r.o., 2014. s. 226-231
- [29] Juščáková, Zuzana: *Hodnotenie kvality šlabikárov učiteľmi*. PEDAGOGIKA.SK, ročník 11, č. 3, 2020, s. 151-168
- [30] Ferenčíková, S., Terek, M., Beňo, M., Hvorecký, J. *Relations between Public Moral, Academic Ethics and University System's Quality*. In: Proceedings of the International Conference Theory and Applications in the Knowledge Economy TAKE 2019. Vienna: WU Vienna University of Economics and Business, 2019, pp. 145-158. https://www.take-conference2019.com/wp-content/uploads/2019/07/2019-07-10_TAKE2019_Conference-Proceedings.pdf
- [31] Berk, R. A. *Professors are from Mars, students are from Snickers*. Sterling: Stylus Publishing, 2003, 184 s.

Programování robotů na základní škole

Robot programming at elementary school

Hana Hyksová

Univerzita Palackého v Olomouci

Žižkovo náměstí 5

771 40 Olomouc

Česká republika

hana.hyksova01@upol.cz

ABSTRACT

Robotics is a very good teaching aid for the development of pupil personality, logical thinking, creativity, systematics, teamwork, manual skills, creativity, critical thinking, the ability to solve not only problems but also tasks and diligence. Robotics helps to develop not only digital literacy, as well as mathematical or reading literacy. All of these competencies are important for employment.

We can program robots in various ways. We take into account not only the age of students, but also the availability and simplicity of robots for the youngest students. In higher grades, students work with more complex robots and create more complex programs. The article describes specific examples of programming various robotic toys and kits in different grades at primary schools based on my experience and the experience of primary school teachers. Due to the current situation where teachers are forced to teach online, the article also provides examples and possibilities of programming robots in distance learning.

Keywords

Robot. Programming. Robot programming. Algorithmization.

ABSTRAKT

Robotika je velmi dobrým pomocníkem při výuce pro rozvoj osobnosti žáka, logického myšlení, tvořivosti, systematickosti, týmové spolupráce, manuální zručnosti, kreativity, kritického myšlení, schopnosti řešit problémy i úlohy a pracovitosti. Robotika pomáhá rozvíjet nejenom digitální gramotnost, ale i matematickou či čtenářskou gramotnost. Všechny tyto kompetence jsou důležité k uplatnění na trhu práce.

Roboty můžeme programovat různými způsoby. V úvahu bereme nejen věk žáků, ale i dostupnost a jednoduchost robotů pro nejmenší žáky. Ve vyšších ročnících žáci pracují se složitějšími roboty a vytvářejí složitější programy. Článek popisuje konkrétní příklady programování různých robotických hraček a stavebnic v různých ročnících na základních školách na základě svých zkušeností a zkušeností učitelů ze základních škol. Vzhledem k současné situaci, kdy jsou učitelé nuceni vyučovat online, jsou v článku uvedeny i příklady a možnosti programování robotů při distanční výuce.

Klíčové slová

Robot. Programování. Programování robota. Algoritmizace.

1 ÚVOD

Roboti, robotika, robotické hračky, robotické stavebnice a STEAM výuka jsou velmi diskutovanými tématy v dnešním školství. Roboti jsou nedílnou součástí našeho života a pomocníky v různých

oborech i domácnostech. Robotika je obor, který se velmi rychle rozvíjí. Vzhledem k tomu, že některé obory mají obrovský nedostatek pracovníků, začínají roboti nahrazovat i lidské činitele.

Úkolem pedagogů je nejen výchova a vzdělávání, ale i příprava žáků na budoucí povolání. Právě příprava na budoucí povolání v sobě obsahuje robotiku. Robotika je obor, který v sobě zahrnuje řadu očekávaných výstupů Rámcového vzdělávacího programu, ze kterých jsou tvořeny Školní vzdělávací programy platné pro I. i II. stupeň základních škol, stejně jako školy střední. Mezi důležité pojmy patří: informatické myšlení, digitální gramotnost, algoritmizace, robotika, programování. Všechny tyto dovednosti nejsou potřebné pouze v informatice, ale i v jiných oborech a předmětech. Důležitými kompetencemi v současné době, a to nejen v oblasti robotiky, jsou:

- umět se rozhodovat a reagovat na změny
- pochopit problém a hledat řešení (informatické myšlení)
- logické myšlení a matematická gramotnost
- čtenářská a digitální gramotnost
- pozitivní vztah k sebevzdělávání
- práce v týmu
- kreativita při řešení problémů a úkolů
- prezentace a obhajoba vlastních myšlenek a řešení (schopnost diskuze)
- pracovitost a trpělivost (dotahovat práci do konce)
- manuální zručnost

Všechny tyto kompetence jsou velmi důležité a jsou nedílnou součástí robotiky. To jsou ty důvody, proč zařadit robotiku do výuky nejen na základních školách.

Vzhledem k současné situaci, kdy probíhá výuka převážně online, uvedu i příklady programování „bez robotů“, pomocí různých simulátorů.

2 VÝUKA STEAM

Hlavním tématem je jednoznačně oblast STEM výuky, dnes již více STEAM výuky, což je propojení vědy, technologií, inženýrství, umění a matematiky. Tyto obory mají budoucnost a jsou považovány za rozhodující pro rozvoj a růst ekonomik, pro udržení konkurenceschopnosti a trvale udržitelný rozvoj.

Dříve než se objevil koncept STEAM, používal se více koncept STEM, který byl zaměřený primárně na vědu, technologie, techniku a matematiku.

Rodiče, děti i učitelé zjistili, že tato metoda není dostačující a nepřináší očekávané výsledky.

STEAM označuje poměrně nový, ale čím dál více populárnější koncept výuky. Myšlenka novějšího konceptu STEAM výuky nespočívá v oddělení výuce technických předmětů a kreativních

předmětů, ale zapojuje do výuky kreativitu a nové způsoby vnímání technických předmětů.

Zkratka STEAM vznikla z prvních písmen anglických názvů Science – věda, Technology – technologie, Engineering – technika, Art – umění a Math – matematika. Může se zdát, že se jedná o složení pěti samostatných oborů, ale STEAM je více o speciálním přístupu k výuce technických předmětů zahrnující využití nejrůznějších kreativních aktivit, jako například hry, tance, hudby, vizuálního umění či divadla. STEAM tedy nepředstavuje klasický způsob vzdělávání, ale přibližuje výuku reálnému životu a nahrazuje příliš teoretické a izolované pojetí výuky vzájemným větším propojením předmětů.

3 ROBOTICKÉ HRAČKY

S robotickými hračkami, u kterých nepotřebujeme žádné konstrukční dovednosti, mohou žáci pracovat již od prvního ročníku. Některé robotické hračky žáci využívají již v mateřských školách. Na základních školách jsou nejvíce využívány Bee-Bot, Blue-Bot a Ozobot.

Další robotické hračky, které jsou v mnohem menší míře na školách Sphero, Dash, Housenka Code a pillar, Cubetto. Jedná se o robotické hračky, které jsou už konstrukčně sestavené. Tyto hračky žáci programují, ale nijak je konstrukčně nesestavují. Žáci vytvářejí algoritmus (jakýsi návod k řešení) a pomocí algoritmu sestavují program, což je posloupnost příkazů, kterým rozumí robot nebo počítač. Jednotlivé pokyny v programování se nazývají příkazy. Robot může vykonat jeden, či více příkazů. Žáci sestavují různé algoritmy a vykonávají různé příkazy v běžném denním životě, např. při oblékání, při chůzi do školy, při psaní domácí úlohy, při vaření čaje.

4 BEE-BOT A BLUE-BOT

Včelka Bee-Bot a beruška Blue-Bot je didaktická pomůcka vhodná pro rozvoj logického myšlení, prostorové orientace a plánování.

4.1 Popis Bee-Bot a Blue-Bot



Obrázek 1: Bee-Bot [autor]

Obrázek 2: Blue-Bot [autor]

Práce s včelkou nenásilně podporuje také jemnou motoriku a posiluje koncentraci pozornosti dítěte. Pohybuje se po hladké podložce ve čtyřech směrech. Včelku Bee-Bot lze naprogramovat na sekvenci až 40 kroků, berušku Blue-Bot až na 400 kroků.

Programování pohybu včelky Bee-Bot a berušky Blue-Bot po vytyčené trase umožňuje propojit nekonečnou zábavu s různými vzdělávacími aktivitami. Za pomoci hladké podložky se čtvercovou sítí rozvíjí včelka u dětí poznávání barev, zvířat, ročních období, či jejich matematické dovednosti, a mnohem víc. Můžeme využít i univerzální podložku s transparentními kapsami, do kterých si sami navolíme obrázky, čísla a různorodé úkoly. Lze samozřejmě zakoupit i tematicky zpracované podložky.

Nabíjení Bee-Bot je přes USB kabel. Lze ji ovládat přes telefon či tablet.

S interaktivní beruškou Blue-Bot si užijeme stejnou legraci jako s robotem Bee-Bot a přitom nemusíme zmáčknout ani jedno jeho tlačítko. K ovládní totiž stačí vlastnit tablet nebo počítač, který je vybaven Bluetooth a mít nainstalovanou Blue-Bot aplikaci, ve které si jednotlivé kroky navolíme. Ta je ke stažení zdarma a je dostupná pro operační systémy iOS, Android, Windows 7 a Mac OS.

Práci s Blue-Bot můžeme rozšířit o sadu pokynů TacTile čtečku. Děti mohou s její pomocí prozkoumat zákoutí geometrie a vytvářet efektivní algoritmy. Set obsahuje 25 destiček s pokyny.

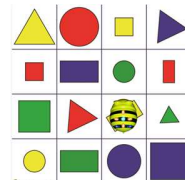
4.2 Bee-Bot emulator

Procvičování programování je důležité i v online výuce. Výuka programování Bee-Bot je možná pomocí emulátoru. Nepotřebujeme fyzicky danou robotickou hračku. Po spuštění emulátoru na stránkách: <https://beebot.terrapiologo.com/> vypadá prostředí takto:



Obrázek 3: Emulátor Bee-Bot [3]

Na obrázku vidíme ovládací šipky pro směr pohybu Bee-Bot, tlačítko stop, reset programu a návrat do původní pozice. Oranžové pole ve spodní části obrázku je bez šipek, neboť zatím není vytvořen žádný program. Můžeme vybírat různé podložky pro řešení úkolů.



Obrázek 4: Podložka Bee-Bot – tvary [3]

Program tvoříme klikáním na jednotlivé šipky na emulátoru. Po kliknutí na jakoukoli šipku se symbol šipky zapíše do oranžového pole ve spodní části obrázku. Program, který je zapsán na obrázku, je vytvořen 10 kroky.



Obrázek 5: Bee-Bot emulátor - zápis programu [3]

6 ROBOTICKÉ STAVEBNICE

Robotické stavebnice mají oproti robotických hračkám další přidanou hodnotu – podporují manuální zručnost, kreativitu a týmovou spolupráci, která je v dnešní době jednou ze základních kompetencí důležitých k uplatnění na pracovním trhu. Robotické stavebnice umožňují STEAM (spojení vědy, technologií, inženýrství, umění a matematiky) výuku v praxi.

Mezi robotické stavebnice, které se nejčastěji používají na základních školách, patří LegoMinstorms EV3, LEGO WeDo 2.0 a mBot. Robotické stavebnice a programovatelné destičky, které školy využívají v menší míře, ale nemají takové rozšíření, jsou Makeblock, Lego Boost, VEX IQ, Fischertechnik, Micro:bit. Práce s robotickými stavebnicemi Lego WeDo a mBot je vhodná pro žáky I. stupně základních škol. LegoMinstorms EV3 a Micro:bit je vhodné zařadit pro žáky II. stupně základních škol. Robotickou stavebnici VEX IQ můžeme zařadit do výuky jak na I. stupni (od 3. třídy), tak i na II. stupni základních škol. Vyšší třída VEX V5 (dřívější název VEX EDR) je sice vhodná pro střední školy, ale můžeme ji zařadit i pro šikovné žáky v 8. a 9. ročníku základních škol.

U robotických stavebnic žáci nevyužívají pouze programování, ale jsou třeba i konstrukční dovednosti, neboť si nejdříve robot musí správně sestavit. Jakmile mají robot správně sestaven, připojují různé senzory. Tyto senzory musí mezi ostatními součástkami správně rozeznat.

Ve škole se svými žáky pracují s robotickou stavebnicí VEX, proto je ukázka zaměřena na tuto stavebnici.

6.1 VEX IQ

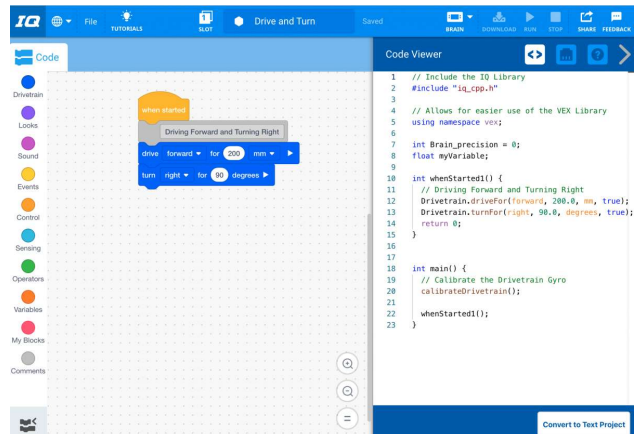
Robotická stavebnice VEX IQ obsahuje více než 800 strukturálních a pohybových komponent, 4 inteligentní motory, 7 senzorů a robotický mozek. Široká škála doplňkových dílů umožňuje postavit roboty větší, složitější i s více funkcemi. Při konstruování robota nepotřebujeme šroubovák. Robota je možno ovládat ručním ovladačem (s předem naprogramovaným kódem, např. pro ověření mechanických vlastností robota) nebo robota naprogramovat tak, aby fungoval samostatně pomocí inteligentních senzorů



Obrázek 12: VEX IQ [autor]

6.2 Programování VEX IQ

Programy a programová prostředí, která využíváme ve stavebnici VEX IQ, jsou VEXcode IQ či Robot C. V programovém prostředí VEXcode IQ programujeme pomocí bloků či příkazových řádků (SCRATCH). V současné době existuje blokové prostředí i textové prostředí. Žáci programují v blokovém prostředí a mohou si zobrazit textové příkazy k danému programu, který vytvořili. Programovací prostředí pro stavebnice VEX jsou ke stažení na stránkách: <https://www.vexrobotics.com>.



Obrázek 13: Program VEXcode IQ [4]

Při práci s robotickou stavebnicí pracují žáci vždy ve skupinách. K dané práci potřebují robot a vytvořený program si ověřují nahráním programu do robota a jeho spuštěním.

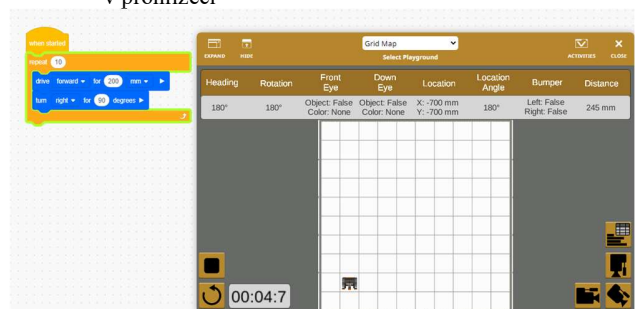
6.3 Simulátor VEXcode VR

Vzhledem k současné situaci i pro robotické stavebnice jsou vytvořeny simulátory. Pro robotickou stavebnici VEX IQ existuje simulátor též, jeho název je VEXcode VR.

VEXcode VR umožňuje kódovat virtuálního robota pomocí blokového kódovacího prostředí založeného na Scratch Blocks. VEXcode používá stejné programovací prostředí pro roboty VEX 123, VEX GO, VEX IQ i VEX V5. Díky STEAM učení můžeme pokračovat ve výuce programování i doma. K výuce nepotřebujeme reálné roboty. Není nutná žádná instalace softwaru. VEXcode VR funguje ve všech hlavních prohlížečích pro stolní počítače, notbooky i tablety.

Ve VEXcode VR můžeme programovat třemi různými způsoby:

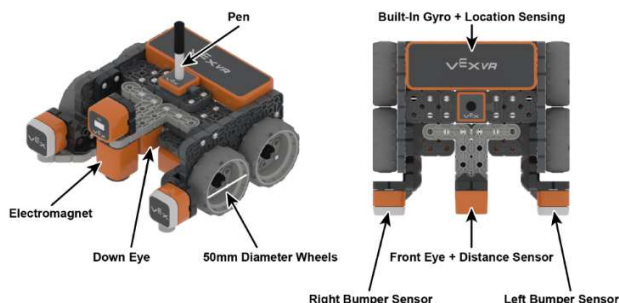
1. Bloky – blokové programování založené na Scratch Blocks
2. Bloky + text – blokové programování založené na Scratch Blocks a kód Pythonu, který se generuje při kódování v blocích pomocí prohlížeče kódu
3. Text – textové programování s Pythonem přímo v prohlížeči



Obrázek 14: Simulátor VEXcode VR [4]

Roboti jsou připraveni. K pohybu můžeme využít pouze hnací ústrojí. K řešení a pohybu v bludištích využíváme senzory. Na uměleckém plátně použijeme pero. Při různých výzvách a hrách

využijeme elektromagnet k interakci s herními objekty. V těchto výzvách můžeme využít několik virtuálních 3D herních hřišť. V současné době hřiště zahrnují mřížkový svět, umělecké plátno či zděné bludiště.



Obrázek 15: Virtuální robot [4]

7 ZÁVĚR

Robotické hračky i stavebnice si stále více nacházejí své místo ve výuce. Na trhu se objevují stále nové možnosti a nabídky nových robotických hraček a stavebnic. Některé jsou pro školy cenově dostupné, některé méně. Robotika je nedílnou součástí našeho života. Učitelé připravují své žáky na budoucí život a povolání. Tudiž robotika nemůže být opomíjena, ale naopak stále více rozvíjena a zařazována do výuky. Při jakékoli práci s robotickými hračkami či stavebnicemi rozvíjíme u žáků logické myšlení, kreativitu, matematickou a čtenářskou gramotnost, týmovou spolupráci, trpělivost, přesnost, pečlivost, soustředěnost a mnoho dalších dovedností. Tyto dovednosti rozvíjíme a uplatňujeme i v ostatních předmětech. Robotiku můžeme zařadit do jakéhokoli předmětu. Rozvojem logického myšlení v robotice rozvíjíme logické myšlení i v ostatních předmětech.

V současné situaci pandemie COVID, kdy je většina škol uzavřených a probíhá online výuka, je důležité zvážit i možnosti výuky programování online. K tomu nám pomáhají různé simulátory různých robotických hraček či robotických stavebnic. Zaměřit si na programování všech robotických hraček a stavebnic není možné. Z tohoto důvodu jsem se ve svém článku zaměřila na robotické hračky a stavebnice, které používám ve své učitelské praxi. Robotickými stavebnicemi řady VEX a programováním, včetně virtuálních robotů, se budu zabývat v dalších článcích.

BIBLIOGRAFICKÉ ODKAZY

- [1] Alza. <https://www.alza.cz/hracky>
- [2] AV MEDIA. <http://www.veskole.cz>. AV MEDIA, 2020.
- [3] Emulator Bee-Bot. <https://beebot.terrapinlogo.com/>
- [4] INNOVATION FIRST INTERNATIONAL . <https://www.vexrobotics.com/>
- [5] Ozobot. <https://ozobot.com/create/ozoblockly>
- [6] Scratch. <https://scratch.mit.edu/projects/19799927/>
- [7] STOFFOVÁ, V.– ZBORAN, M.: *Hravá forma stavby a programování robotů na základní škole*. In: Trendy ve vzdělávání, 11, 2018, č. 2, s. 130 – 139. ISSN 1805-8949

- [8] STOFFOVÁ, V.–TAKÁČ, O.: *Robotické stavebnice v přípravě učitelův informační výchovy (Robot kits in teachers preparation for information education)*. In Havelka, M., Chráska, M., Klement, M., Serafin, Č. (ed.): Trendy ve vzdělávání 2013. Olomouc : agentura GEVAK s.r.o., 2013. 315-322. s. ISBN 978-80-86768-52-6 / ISSN 1805-8949

Príprava na medzinárodný benchmarking vo vyučovaní úvodných kurzov programovania v jazyku Python

Preparation for international benchmarking in teaching introductory programming in Python

Jana Jacková, Dušan Mráz, Patrik Benyak

Katedra informatiky, Pedagogická fakulta KU v Ružomberku
Hrabovská cesta 1
034 01 Ružomberok
Slovensko

jana.jackova@ku.sk, 1dusan.mraz1@gmail.com, patrik@benyak.eu

ABSTRACT

The paper follows an international initiative of computer science teachers in introductory programming courses in several countries. For benchmarking the results of their students they used a set of 10 specified questions in versions for Java, C or Python programming languages as a part of final exams in their institutions. Since 2009 we have been gradually involved in solving this topic at three Slovak universities. During the years 2016–2019, we pilot-tested the international version of the test for the C language (translated into Slovak) in Slovakia. We have currently started to verify the use of this test for the Python language at the Catholic University in Ružomberok and at the Grammar School of P. O. Hviezdoslav in Kežmarok. For each question of the international test, we have created some alternative questions for the purpose of practice during classes. This should ensure that students are prepared better for the types of questions used in the international test.

Keywords

Programming. Python. Beginners. Assessment. Benchmarking. Examination. Test.

ABSTRAKT

Príspevok nadväzuje na medzinárodnú iniciatívu učiteľov informatiky z niekoľkých krajín, ktorí urobili v rámci výstupných testov v úvodných kurzoch programovania na ich školách aj vzájomný benchmarking pomocou sady 10 vyšpecifikovaných úloh v úprave pre programovacie jazyky Java, C a Python. Do riešenia tejto problematiky sme sa zapojili postupne od roku 2009 na troch slovenských vysokých školách. V rokoch 2016 až 2019 sme na Slovensku pilotne overili medzinárodnú verziu tohto testu pre jazyk C (preloženú do slovenčiny). V súčasnosti sme začali overovať použitie tohto testu pre jazyk Python na Katolíckej univerzite v Ružomberku a na Gymnáziu P. O. Hviezdoslava v Kežmarku. Ku každej úlohe medzinárodného testu sme vytvorili alternatívne úlohy na účely precvičenia počas vyučovania, čo by malo študentov lepšie pripraviť na typy úloh medzinárodného testu.

Kľúčové slová

Programovanie. Python. Začiatočníci. Hodnotenie. Benchmarking. Skúška. Test.

1 ÚVOD

Podľa [1, s. 35] sa pojmom benchmarking „označuje porovnanie školy ... s inou školou ..., ktorá má podobné charakteristiky“. Základom je podľa [2] „porovnanie vybraných ukazovateľov voči iným referenčným hodnotám, ktoré môžu byť buď historické ..., alebo môžu byť porovnané voči inému referenčnému subjektu

(napr. inej ... porovnateľnej organizácii). Toto porovnanie je vždy relatívne. Nemožno povedať, že vyššie alebo nižšie hodnoty ukazovateľov sú zlé alebo dobré. Najväčším prínosom benchmarkingu je, že odlišné hodnoty vyvolávajú otázky“ o príčinách týchto odlišných hodnôt, čo by malo iniciovať ďalší výskum.

Sheard et al. [3] uvádzajú, že benchmarking môže pomôcť pri zisťovaní kvality rôznych úvodných kurzov programovania na základe použitia rovnakej sady úloh v rámci časti záverečných testov/skúšok. Za úvodný kurz programovania sa typicky považuje taký programátorský predmet, ktorým sa začína výučba programovania na danej škole; na vysokej škole trvá obvykle 1 semester. Pri takomto benchmarkingu ide podľa [4, s. 103] o „explicitné meranie výsledkov v rôznych inštitúciách a ich porovnanie navzájom alebo so stanoveným benchmarkom.“ Pojem benchmark sa podľa [5] používa „pre jakýkoľvek štandard, referenčnú hodnotu, ktorou sa meria výkonnosť alebo sa pomocou nej výkonnosť porovnáva. Ako sme uviedli v [6], na benchmarking rôznych rovnako zameraných kurzov/predmetov v rôznych inštitúciách sa nemôžu ako ukazovatele použiť známky študentov dosiahnuté pri záverečnom hodnotení týchto kurzov. Výsledky študentov vo forme výsledného hodnotenia/známky za jednotlivé kurzy uskutočnené v rôznych inštitúciách totiž vôbec nezaručujú porovnateľnosť týchto výsledkov.

V [6] sme opísali medzinárodnú iniciatívu učiteľov informatiky, v ktorej bola v rámci výstupných testov v úvodných kurzoch programovania na ich školách (väčšinou ako časť skúšky uskutočnenej na konci konania kurzu) použitá aj sada 10 úloh – v úprave podľa programovacieho jazyka, ktorý sa pri výučbe programovania v danom kurze používal a vo vyučovacom jazyku používanom v danej inštitúcii. Tieto úlohy boli vyšpecifikované na základe viacročného výskumu programátorského vzdelávania [7], [3], [8]. Do skúmania problematiky vzájomného porovnania výsledkov vzdelávania v úvodných kurzoch programovania sme sa od roku 2009 postupne zapojili na troch slovenských vysokých školách: na Žilinskej univerzite v Žiline (ďalej ŽU) [7], Univerzite Mateja Bela v Banskej Bystrici (ďalej UMB) [6] a na Katolíckej univerzite v Ružomberku (ďalej KU) [9]. Na základe oboznámenia sa s medzinárodnou verziou 10 úloh, ktoré v roku 2015 uverejnili Simon et al. pre jazyk Java v prílohe [8, s. 146], i podrobnejšie opísali ich obsah v [4], sme požiadali Dr. Simona z The University of Newcastle, Australia, o verziu pre jazyk C. Jazyk C sa používal pri výučbe v úvodných kurzoch programovania na Katedre informatiky FPV UMB a na Elektrotechnickej fakulte ŽU. Úlohy

pre jazyk C sme preložili z angličtiny do slovenčiny a detailnejšie opísali v [6]. Výsledky pilotného používania a overovania tohto medzinárodného testu na UMB a ŽU v rokoch 2016 až 2019 sme publikovali v [9].

Podľa [10] a [6] takýto benchmarking umožňuje školám analyzovať ich vlastné výsledky z rôznych hľadísk: napr. výkon v 10 úlohách všetkých zúčastnených kurzov/inštitúcií (prehľad), porovnanie s priemerom, identifikácia anomálií, porovnanie v rámci jedného kurzu, porovnanie kurzov v rámci jednej inštitúcie, porovnanie s celkovým výkonom v rámci skúšky (táto skúška vo väčšine prípadov obsahuje aj ďalšie vlastné úlohy pripravené učiteľmi kurzu v danej inštitúcii), veľkosť skúšky vzhľadom na zaradenie 10 benchmarkingových úloh.

Vzhľadom na to, že na Katedre informatiky Pedagogickej fakulty KU sa ako úvodný programovací jazyk začal používať Python, rozhodli sme sa overiť aj použitie medzinárodnej sady 10 úloh [8, s. 146], [4] pre jazyk Python a požiadali sme Dr. Simona o poskytnutie verzie pre Python. Úlohy tohto testu sme preložili z angličtiny do slovenčiny a plánovali sme ho použiť v rámci výstupných testov alebo skúšok (ako ich časť) v úvodných kurzoch programovania v jazyku Python. Rámcový obsah tohto jednotného medzinárodného testu pozostávajúci z 10 úloh vyšpecifikovaných viacročnou medzinárodnou iniciatívou učiteľov informatiky uvádzame v kapitole 2. Vzhľadom na to, že niektoré typy úloh tohto testu nemusia byť pre študentov známe, pripravili sme pre potreby oboznámenia sa s typmi úloh a precvičenia aj alternatívne verzie úloh medzinárodného testu. Ich ukážku uvádzame v kapitole 2.2.

2 SADA 10 ÚLOH PRE BENCHMARKING

Sada úloh, o ktorej sme sa zmienili vyššie, predstavuje šablónu testu s 10 vyšpecifikovanými základnými programátorskými úlohami vo verziách pre rôzne programovacie jazyky v angličtine. Znenie úloh pre jazyk Java je uvedené v prílohe [8, s. 146], resp. [10, s. 159], a zadanie každej z týchto úloh je detailnejšie komentované v [4]. Schematický základný prehľad týchto úloh uvádzame v tabuľke 1. Učelia z rôznych krajín, ktorí mali záujem o použitie testu, si zabezpečili preklad testu do vyučovacieho jazyka používaného v ich škole. Ku každej verzii sú pripravené sprievodné informácie na použitie testu a vyhodnotenie (odporúčané riešenie, javová analýza pre pridelenie bodov), čím sa zabezpečia pokiaľ možno podobné podmienky použitia testu a rovnaký postup pri jeho vyhodnocovaní.

V pravej časti tabuľky 1 (benchmark) uvádzame celkový výkon študentov za jednotlivé úlohy (úspešnosť riešenia úlohy uvedená v percentách) vo väčšine vyriešených testov podľa [4] a poradie úspešnosti riešenia úlohy v tomto teste. Tento benchmark vychádza z výsledkov 13 skúšok na 7 vysokých školách z 5 krajín (Austrália: 3 školy, z nich v 2 školách Java, v 1 škole Python; Fínsko: 1 škola, Python; USA: 1 škola, 2 kurzy, Java; Holandsko: 1 škola, jazyk C; Kanada: 1 škola, Python). V týchto školách tvoril tento medzinárodný test od 16 do 47% záverečnej skúšky z predmetov (16%, 25 %, 29%, 36%, 43%, 45%, 47%) a samotná záverečná skúška prispievala do celkového hodnotenia predmetu (výslednej známky za predmet) takto: v 6 zúčastnených školách od 25 do 60 % (25%, 30%, 40%, 45%, 50%, 60%) a v 1 škole bola skúška iba nutnou podmienkou postupu, pričom jej výsledok neprispieval do celkového hodnotenia predmetu. Ak tento test budú používať aj ďalší a následne budú o svojich výsledkoch informovať autorskú skupinu Simon et al., tak na základe toho bude možné benchmark uvedený v tabuľke 1 aktualizovať. V prípade záujmu

o podrobnejšie informácie akým sa vyššie uvedených 7 vysokých škôl zapojilo do tohto benchmarkingu a aké výsledky dosiahli v jednotlivých školách, je možné sa o tom dozvedieť v [4], v kapitole 4 a [10], v kapitole 4.

Tabuľka 1: Prehľad 10 úloh pre benchmarking [4], [6]

koncepty	číslo (body)	%	ÚLOHY		benchmark	
			spôsobilosť	typ odpovede	%	poradie
výrazy <i>zad. v prirodz. jazyku</i>	1 (1b)	10	trasovať kód	výber <i>zo 4 odp.</i>	90	2
priradenie a postupnosť	2 (1b)	10			89	3
výmena a posun <i>hodnôt pri neznámom kontexte</i>	3 (3b)	10	písať kód <i>k zad. problému</i>	napísať kód	70	7
výber (vetvenie)	4 (1b)	20	trasovať kód	krátka <i>hodnota premennej</i>	96	1
	5 (1b)			výber <i>zo 4 odp.</i>	89	4
iterácia (cykly) a polia	6 (1b)	50	čítať kód <i>(objasniť účel prirodz. jazykom)</i>	krátka <i>hodnota premennej</i>	80	6
	7 (1b)			výber <i>z 5 odp.</i>	85	5
	8 (6b)		krátka <i>vlastnými slovami</i>	44	10	
	9 (6b)		modifikovať kód <i>na vyriešenie zad. problému</i>	napísať kód	66	8
<i>funkcia</i>	10 (6b)		písať kód <i>k zad. problému</i>		64	9

Podľa autorov testu by sa tieto úlohy nemali stať meradlom pre stanovenie relatívneho rebríčka inštitúcií v programátorskom vzdelávaní [4], a ani globálnym meradlom pre úvodné kurzy programovania. Test je možné použiť iba v predmetoch, v ktorých sa vyučuje programovanie v procedurálnych alebo objektovo-orientovaných jazykoch, aj to nie všetkých, pretože niektoré takéto úvodné programátorské predmety svojimi obsahovými štandardami nepokrývajú obsah testu. Napriek tomu aj na základe našich doterajších skúseností s využívaním tohto testu súhlasíme s jeho autormi, že tieto úlohy a ich používanie môžu byť pre mnohých učiteľov informatiky veľmi užitočné. Simon et al. [4] však odporúčajú starostlivé zváženie interpretácie získaných výsledkov, pretože na výkon študentov v konkrétnom kurze/predmete konkrétnej školy pôsobí veľa faktorov. Každé zistenie má byť interpretované vo svojom vlastnom kontexte, teda v kontexte konkrétneho kurzu/predmetu.

2.1 Úlohy pre jazyk Python

Problematika 10 úloh testu podľa [8, s. 146], resp. [10, s. 159], pozostáva z overenia porozumenia základných konceptov programovacieho jazyka, ktoré boli vyšpecifikované medzinárodnou iniciatívou učiteľov informatiky tak, aby mohli byť použité o. i. aj na čiastkový benchmarking v rámci rôznych úvodných kurzov programovania. Test vo verzii pre jazyk Python sme si vyžiadali od Dr. Simona a preložili do slovenčiny. V jeseni 2020 sme začali spolupracovať s Gymnáziom P. O. Hviezdoslava v Kežmarku, na ktorom začal vyučovať informatiku študent učiteľstva informatiky na KU Bc. P. Benyak. Prácu s týmto testom sme plánovali vyskúšať aj v rámci jeho výučby na tejto škole.

Na základe informácií v tabuľke 1 sú v tomto teste z obsahového hľadiska zastúpené tieto *tematické oblasti*: výrazy (1 úloha: č. 1), priradenie a postupnosť (1 úloha: č. 2), výmena obsahu premenných (1 úloha: č. 3), vetvenie (2 úlohy: č. 4, 5), iterácia (cykly) a polia (5 úloh: č. 6 – 10; úloha 10 je na funkciu). Autori testu odporúčajú pri úlohách na polia nahradiť v zadaní slovo „pole“ slovom „zoznam“, ak sa v danom používa názov zoznam.

Z hľadiska požadovaných *spôsobilostí* sú v teste úlohy na trasovanie kódu (5 úloh: č. 1, 2, 4, 5, 6), písanie kódu (2 úlohy: č. 3, 10), čítanie kódu v zmysle porozumenia účelu kódu a jeho objasnenie krátkou vetou (namiesto opisovania jednotlivých príkazov – „stromov“ rozpoznáť účel kódu ako celku – „les“; 2 úlohy: č. 7, 8) a modifikáciu kódu (1 úloha: č. 9).

Podľa *typu odpovede* sú v teste použité úlohy s výberom odpovede (4 úlohy: č. 1, 2, 5, 7), úlohy s krátkou odpoveďou (3 úlohy: č. 4, 6, 8) a úlohy s písaním kódu (3 úlohy: č. 3, 9, 10).

Podľa [4, s. 110] by mal učiteľ v prípade záujmu o použitie tohto medzinárodného testu v rámci záverečného testu/skúšky svojho kurzu/predmetu (úlohy medzinárodného testu by mali tvoriť iba časť skúšky) pracovať počas výučby so všetkými typmi úloh, ktoré sú v medzinárodnom teste. Študentov by nemali tieto typy úloh v záverečnom teste prekvapiť, mali by im byť známe. Nemalo by sa stať, že menej obvyklé typy úloh (napr. úlohy č. 7 a 8 na čítanie kódu s porozumením a vysvetlenie účelu kódu krátkou vetou) študenti nikdy predtým neriešili. Na tieto účely sme ku každej úlohe medzinárodného testu pripravili viacero alternatívnych úloh na použitie/precvičenie počas výučby predmetu.

2.2 Úlohy na precvičenie

Na základe odporúčaní uvedených v predchádzajúcej kapitole sme na precvičenie pripravili alternatívne úlohy, ktoré typovo korešpondujú s úlohami medzinárodného testu (publikovaný v prílohe [8, s. 146], resp. [10, s. 159]). Pri tvorbe nových úloh určených na oboznámene sa s typmi úloh tohto medzinárodného testu sme sa z hľadiska pokrytia tém, overovaných spôsobilostí a typov odpovede snažili pridržať šchematického prehľadu úloh, ktorý sme uviedli v tabuľke 1. V tejto kapitole uvádzame po jednom príklade možnej alternatívnej úlohy určenej na precvičenie, ktorá poradím i typovo podľa možnosti korešponduje s príslušnou úlohou medzinárodného testu [8], [10].

1. Ak máme k dispozícii viac ako 8 parkovacích miest a aspoň 30 litrov paliva, tak ktorý z nasledujúcich výrazov by definoval tento výrok?

- a) `parkovanie >= 8 or palivo > 30`
- b) `parkovanie >= 8 and palivo > 30`
- c) `parkovanie > 8 or palivo >= 30`
- d) `parkovanie > 8 and palivo >= 30`

2. Aké sú hodnoty v *listnate*, *ihlicnate* a *stromy* po vykonaní nasledujúceho kódu?

```
ihlicnate = 26
stromy = 50
listnate = stromy-ihlicnate
ihlicnate = 15
listnate = 8
```

- a) 24, 26, 50 b) 8, 26, 24 c) 24, 15, 50 d) 8, 15, 50

3. V obchode sa menia ceny pre tieto tri druhy ovocia: *jablka*, *hrusky* a *pomarance*. Napíšte časť kódu, ktorá zabezpečí presunutie cien tohto ovocia tak, že po vykonaní tohto kódu budú mať *jablka* takú cenu ako mali predtým *hrusky*, *hrusky* budú mať pôvodnú cenu pre *pomarance* a *pomarance* budú mať pôvodnú cenu pre *jablka*.

4. Aká bude hodnota premennej *b* po vykonaní nasledujúceho kódu?

```
a = 2
b = 4
if a < b:
    a = b*2
if a < b:
    b = a*2
else:
    a = 0
```

5. Uvažujte o nasledujúcej časti kódu, v ktorom sa do premenných *x*, *y*, *z* a *odpoved* ukladajú celočíselné hodnoty:

```
if x >= y:
    if x >= z:
        odpoved = x
elif z >= y:
    odpoved = z
else:
    odpoved = y
```

Ktorá z nasledujúcich možností pre hodnoty premenných *x*, *y* a *z* spôsobí, že sa do premennej *odpoved* priradí hodnota, ktorú má premenná *y*?

- a) `x = 2, y = 2, z = 3` b) `x = 1, y = 3, z = 2`
- c) `x = 2, y = 1, z = 3` d) `x = 3, y = 2, z = 1`

6. Aká hodnota bude v premennej *vysledok* po vykonaní nasledujúcich príkazov v kóde programu?

```
cisla = [2, 9, -5, 4, 0, -1, -3]
vysledok = 0
i = 0
for i in range(0, len(cisla), 2):
    vysledok = vysledok + cisla[i]
```

7. Čo je cieľom nasledujúcej časti kódu (aký má účel ako celok)?

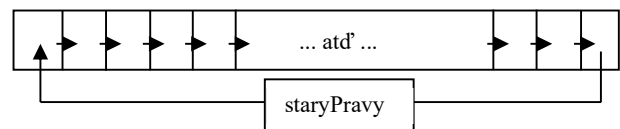
```
vysledok = 0
for i in range(0, len(pocitaj)):
    if pocitaj[i] % 2 == 0:
        vysledok = vysledok + 1
```

- a) zistiť najväčšie číslo v poli deliteľné dvomi
- b) pripočítať 1 ku každému párnemu číslu v poli
- c) sčítať párne čísla v poli
- d) spočítať párne čísla v poli
- e) zistiť index posledného párneho čísla v poli

8. Čo je cieľom nasledujúcej časti kódu (aký má účel ako celok)? Vyjadrite svoju odpoveď vo forme krátkej vety (podobným spôsobom akým sú uvedené možné odpovede pri úlohe 7).

```
vysledok = 0
cislo = 100
while cislo > 0:
    cislo = cislo // 2
    vysledok = vysledok + 1
```

9. Programátor chce presunúť všetky prvky zadaného poľa o jedno miesto doprava, pričom prvok, ktorý bol najviac vpravo sa presunie do pozície najviac vľavo, tak ako je znázornené v tejto schéme.



Tu je kód, ktorý vykoná taký posun pre pole s názvom *zoznam*:

```
dĺzka = len(zoznam)
staryPravy = zoznam[dĺzka - 1]
for i in range(dĺzka - 1, 0, -1):
    zoznam[i] = zoznam[i - 1]
zoznam[0] = staryPravy
```

Napríklad ak by na začiatku pole *zoznam* obsahovalo tieto celé čísla [7, 20, 5, 8, 3], tak po vykonaní tohto kódu by malo obsahovať [3, 7, 20, 5, 8].

Napište kód, ktorý v poli celých čísel *zoznam* nájde prvok s najvyššou hodnotou, vypíše poradie tohto prvku a jeho hodnotu.

10. Napište funkciu, ktorej treba zadať pole 7 znakov a ktorá zistí, do akej miery sa zadané pole zhoduje s poľom

```
A = [„p“, „r“, „o“, „g“, „r“, „a“, „m“].
```

Výsledkom funkcie má byť počet zhodných prvkov.

Napríklad pri zadaní poľa *znaky* [„a“, „b“, „c“, „d“, „e“, „f“, „g“] by funkcia mala vrátiť hodnotu 0, lebo sa s poľom P nezhoduje na žiadnom mieste.

Pri zadaní poľa *znaky* [„p“, „b“, „m“, „g“, „a“, „o“, „m“] by funkcia mala vrátiť hodnotu 3, lebo sa s poľom P zhoduje na prvom, štvrtom a poslednom mieste.

3 POUŽITIE TESTOVÝCH ÚLOH NA STREDNEJ ŠKOLE

Na základe výsledkov riešenia úloh medzinárodného testu [8], [10], ktorý sme priblížili v úvode kapitoly 2 a tabuľke 1, učiteľ zistí v svojich študentov znalosť základnej syntaxe a ich spôsobilosti v čítaní kódu s porozumením, trasovaní kódu, písaní kódu a modifikácii kódu.

K pilotnému vyskúšaní testových úloh pre Python na strednej škole nás viedla situácia, keď začiatkom školského roka 2020/2021 začal študent KU Bc. P. Benyak vyučovať na Gymnáziu P. O. Hviezdoslava v Kežmarku [11] informatické predmety (10 vyučovacích hodín týždenne v 3. a 4. ročníku). Ide o štvorročné gymnázium, na ktorom sa informatika vyučuje v 1. ročníku 2 hodiny týždenne a v 2. a 3. ročníku po 1 hodine týždenne. V posledných dvoch ročníkoch si žiaci môžu vybrať z voliteľných predmetov aj informatické predmety: v 3. ročníku 2-hodinový seminár z informatiky (ďalej SEI) a v 4. ročníku 4-hodinový SEI a 2-hodinové cvičenia z informatiky [12]. Z týchto hodín prevzal P. Benyak 10 vyučovacích hodín týždenne v 3. a 4. ročníku, a to povinnú tretiačku informatiku v jednej triede (2 skupiny) a všetky vyššie uvedené voliteľné informatické predmety. Tretiačky SEI navštevuje 16 žiakov, z ktorých 10 plánuje študovať informatiku na vysokých školách. Štvrtáčky SEI navštevuje 10 žiakov, z ktorých 8 plánuje študovať ďalej informatické alebo príbuzné smery. Nakoľko P. Benyak týchto žiakov predtým neučil a väčšina z nich plánuje maturovať z informatiky, rozhodol sa prebrať všetko predpísané učivo v rámci seminárov od začiatku u tretiačkov i štvrtákov.

Medzinárodný test pre Python v slovenčine sme použili v tomto školskom roku zatiaľ v 3. ročníku u tých žiakov z III.A a III.B, ktorí navštevujú aj voliteľný predmet SEI. Podľa rámcového učebného plánu školského vzdelávacieho programu [11, 12] títo žiaci majú týždenne 1 hodinu povinnej informatiky (každú triedu učí iný učiteľ) a 2 hodiny SEI, pričom v predchádzajúcich ročníkoch mali spolu 3 hodiny povinnej informatiky (1. ročník 2 hodiny, 2. ročník 1 hodina týždenne). Test ešte plánujeme použiť aj v rámci tohtoročného štvrtáckeho SEI.

Predpokladali sme, že skúsenosti a výsledky získané z použitia tohto testu na tretiačkom a štvrtáckom SEI prinesú pre nás nové podnety pre ďalšiu výučbu programovania. Je pre nás taktiež veľmi užitočné, že sme vytvorili ďalšie alternatívne úlohy typovo podobné ako v medzinárodnom teste, vďaka čomu môžeme žiakov lepšie pripraviť na priebežné skúšanie a kontrolné či

výstupné testy. Niektoré z týchto úloh určených na precvičenie počas bežného vyučovania sme uviedli v kapitole 2.2.

3.1 Testovanie a výsledky

V originálnom výskume [4], z ktorého sme čerpalí údaje do tabuľky 1, prebehlo testovanie na každej zo 7 zúčastnených (vysokých) škôl prezenčne v jednej učebni a študenti riešili test do papierových testovacích hárkov, pričom nemohli používať žiadne pomôcky (na lavici mali iba testovacie hárky a pero). Podobne prebehlo aj testovanie na ŽU a UMB (vo verzii pre jazyk C), o ktorom sme informovali v [9]. Vzhľadom na aktuálnu situáciu na Slovensku, začiatkom roka 2021 bolo možné vyskúšať test na gymnáziu v Kežmarku len dištančnou formou.

V januári 2021 absolvovalo medzinárodnú verziu testu pre Python v slovenčine 16 žiakov 3. ročníka na voliteľnom predmete SEI v rámci polročného testu. Keďže sa ešte stále nachádzame v covidovom období a žiaci sa vzdelávajú dištančne, tento test sme vytvorili na stránkach EduPage gymnázia v Kežmarku, odkiaľ ho mohli žiaci bezproblémovo vypracovať. Štandardne sa im test otvoril na začiatku hodiny a mali na jeho vypracovanie 45 minút. Na obrázku 1 uvádzame ukážku úlohy č. 6 medzinárodného testu pre Python v prostredí EduPage v slovenčine.

Otázka

6. / 10

Áká hodnota bude v premennej *vysledok* po vykonaní nasledujúcich príkazov v kóde programu?

```
cisla1 = [1, -5, 2, 0, 4, 2, -3]
cisla2 = [1, -5, 2, 4, 4, 2, 7]
vysledok = 0
j = 0
while j < len(cisla1):
    if cisla1[j] != cisla2[j]:
        vysledok = vysledok + 1
    j = j + 1
```

Obrázok 1: Úloha č. 6 medzinárodného testu pre Python

Na porovnanie zadania úlohy č. 6 z obrázku 1 uvádzame pôvodné zadanie úlohy č. 6 pre jazyk Java [8, s. 146], resp. [10, s. 159] v angličtine:

Q6. What will be the value of *result* after the following code statements are executed?

```
int[] nums1 = { 1, -5, 2, 0, 4, 2, -3 };
int[] nums2 = { 1, -5, 2, 4, 4, 2, 7 };
int result = 0;
int j = 0;
while (j < nums1.length)
{
    if (nums1[j] != nums2[j])
    {
        result = result + 1;
    }
    j = j + 1;
}
```

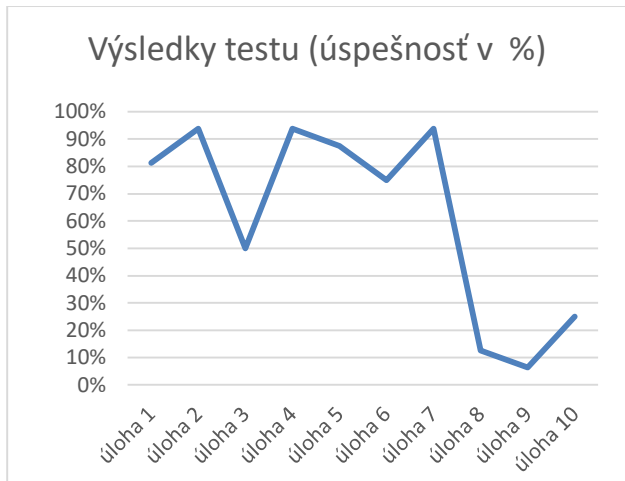
Ako môžeme vidieť z porovnania zadani úlohy č. 6 v testoch pre jazyk Python (Obrázok 1), jazyk Java alebo jazyk C nižšie, texty zadani pre túto úlohu sú vo verziách pre rôzne programovacie jazyky podobné.

Zadanie úlohy č. 6 pre jazyk C [6, s. 8] preložené do slovenčiny:

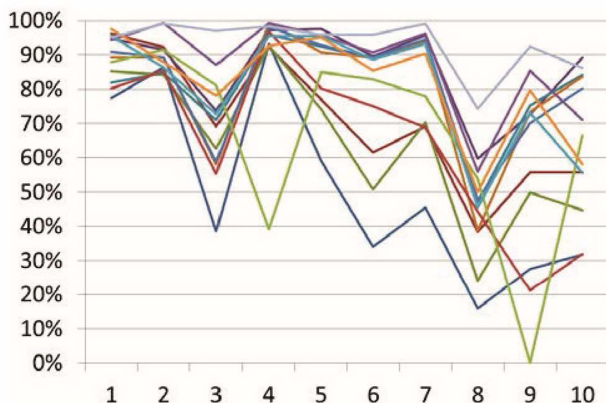
Úloha 6 (1 bod): Aká hodnota bude v premennej *vysledok* po vykonaní nasledujúcich príkazov v kóde programu?

```
#define DLZKA 7
int cisla1[DLZKA] = { 1, -5, 2, 0, 4, 2, -3 };
int cisla2[DLZKA] = { 1, -5, 2, 4, 4, 2, 7 };
int vysledok = 0;
int j = 0;
while (j < DLZKA) {
    if (cisla1[j] != cisla2[j]) {
        vysledok = vysledok + 1;
    }
    j = j + 1;
}
```

Na obrázku 2 uvádzame celkové výsledky nášho pilotného vyskúšania medzinárodného testu v Pythone riešeného skupinou 16 žiakov 3. ročníka gymnázia v rámci voliteľného predmetu SEI. Pre základné porovnanie našich výsledkov z obrázku 2 uvádzame na obrázku 3 výsledky medzinárodného testu uskutočneného v rámci 13 skúšok na 7 rôznych vysokých školách v 5 krajinách (Austrália, USA, Kanada, Fínsko, Holandsko) vo verziách pre programovacie jazyky Java, Python a C [4], [10].



Obrázok 2: Priemerná úspešnosť riešenia úloh medzinárodného testu skupinou stredoškôlkov dištančne



Obrázok 3: Priemerná úspešnosť riešenia úloh testu v 13 skúškach podľa [10, s. 155]

3.2 Diskusia

Podľa prehľadu celkových výsledkov medzinárodného testu na obrázku 2 môžeme konštatovať, že žiaci gymnázia najlepšie zvládli úlohy č. 2, 4 a 7, všetky s priemernou úspešnosťou riešenia 93,8 %. Úlohy č. 5, 1 a 6 boli taktiež percentuálne dobre zvládnuté, s úspešnosťou okolo 80 %. Ako možno vidieť zo schematického prehľadu testu v tabuľke 1, pri týchto šiestich úlohách bolo potrebné vybrať buď jednu z ponúkaných odpovedí (úlohy č. 1, 2, 5, 7) alebo napísať jednu hodnotu (úlohy č. 4, 6). V úlohe č. 3 žiaci dosiahli úspešnosť 50 %. Na rozdiel od vyššie vymenovaných lepšie zvládnutých úloh, v tejto úlohe museli žiaci napísať časť kódu programu. V úlohách č. 10, 8 a 9 boli žiaci najmenej úspešní (25 %, 12,5 % a 6,3 %). V úlohe č. 10 (úspešnosť 25 %) bolo potrebné napísať časť kódu programu, funkciu. Úloha č. 8 (úspešnosť 12,5 %) si síce vyžadovala iba stručnú odpoveď formulovanú vlastnými slovami do krátkej vety, ale v tejto úlohe bola sledovaná spôsobilosť žiakov čítať kód v zmysle porozumenia účelu kódu a jeho objasnenie krátkou vetou (namiesto opisovania jednotlivých príkazov – „stromov“ rozpoznať účel kódu ako celku – „les“). Táto spôsobilosť bola sledovaná aj v úlohe č. 7, avšak vzhľadom na to, že žiakom v nej boli ponúknuté možnosti na výber odpovede, tak tam boli oveľa úspešnejší (93,8 %). Tento veľký rozdiel v úspešnosti riešenia úloh č. 7 a 8 evokuje potrebu precvičovania väčšieho počtu úloh na čítanie kódu s porozumením bez možnosti výberu z odpovedí. V úlohe č. 9 s najnižšou dosiahnutou úspešnosťou (6,3 %) bolo potrebné modifikovať kód.

Aj keď výskum uvedený v [4] bol realizovaný v iných podmienkach (vysoké školy, prezenčne, papierové testy), keď sa pozrieme na benchmark uvedený v tabuľke 1, tak aj v [4] boli najúspešnejšie riešené úlohy č. 1, 2, 4, 5, 6, 7 s úspešnosťou od 96 % (č. 4) po 80 % (č. 6), v poradí úspešnosti č. 4, 1, 2, 5, 7, 6. V týchto šiestich úlohách dosiahla naša skupina žiakov gymnázia úspešnosť od 93,8 % (č. 2, 4, 7) po 75 % (č. 6). Poradie úspešnosti riešenia úloh č. 5, 6 a 3 u našich žiakov je rovnaké ako v tabuľke 1. Aj vo výskume [4] bola dosiahnutá v posledných troch úlohách testu najnižšia úspešnosť (úlohy č. 9, 10 a 8, s úspešnosťou od 66 % po 44 %).

Z našich prvotných výsledkov použitia medzinárodného testu na vzorke 16 respondentov gymnázia vyplýva, že podobne ako vo výsledkoch výskumu [4], aj naši žiaci si potrebujú zlepšovať svoje algoritmicke myslenie pri písaní kódu k riešeniu zadaných problémov. Pri rozvoji algoritmickeho myslenia taktiež môže pomôcť častejšie zaraďovanie úloh typu č. 7 a 8 (na čítanie neznámeho kódu, so zisťovaním účelu kódu ako celku) do bežného vyučovania, napr. v rámci programátorskej rozcvičky na začiatku vyučovania.

Vzhľadom na to, že sme na Slovensku získali skúsenosti aj s podmienkami testovania podľa [4], ktoré sme opisali v [9] (2 vysoké školy, prezenčne, papierové testy, jazyk C), domnievame sa, že ak by žiaci gymnázia absolvovali test spoločne prezenčnou formou v škole, pravdepodobne by pri zabezpečení rovnakých podmienok na jednom mieste pre všetkých žiakov dosiahli v niektorých úlohách testu odlišné výsledky. Pri riešení testu dištančnou formou by bolo vhodné zistiť, či žiakom stačil čas 45 minút na riešenie. Je možné, že pri poskytnutí dlhšieho času na riešenie (odporúčame 60 minút) by bola úspešnosť riešenia v posledných troch úlohách vyššia. Vplyv dostatku poskytnutého času na riešenie úlohy na vyššiu školskú úspešnosť žiakov potvrdzuje aj teória mastery learning [13].

Podľa [4] by mal tento test tvoriť iba časť skúšky úvodných kurzov programovania. Preto odporúčame, aby si učiteľ predmetu urobil pre svoju skúšku špecifikačnú tabuľku [14], v ktorej by zmapoval pokrytie učiva kurzu podľa vzdelávacích štandardov tohto predmetu otázkami medzinárodného testu a potom by ju doplnil úlohami na overenie ďalších požadovaných spôsobilostí úspešných absolventov predmetu, ktoré tento test neoveruje. Tým by sa zabezpečila obsahová validita skúšky, vyvážené zastúpenie oblastí učiva, ale aj taxonómie úloh a náročnosť myšlienkových operácií.

4 ZÁVER

V našom príspevku sme sa zaoberali využitím jednotného medzinárodného testu [8], [10] s 10 úlohami použitím v štúdiách [3, 4], [9, 10]. Tento test bol pôvodne vytvorený pre jazyk Java (publikovaný v prílohách [8, s. 146], [10, s. 159]) a jeho úlohy boli detailnejšie analyzované v [4]. Test bol pre účely benchmarkingu konvertovaný do programovacích jazykov C, C#, Visual Basic, Python [3], [8]. Základný schematický prehľad úloh testu sme spolu s prvým medzinárodným benchmarkom podľa [4] uviedli v tabuľke 1. Na účely benchmarkingu sa test odporúča použiť na konci výučby úvodného programátorského predmetu, ako časť výstupnej skúšky.

V prípade, že chceme tento medzinárodný test využiť pre akýsi náš interný benchmarking vo vzťahu k publikovanému benchmarku podľa [4], bolo by dobré pripraviť na našich študentov/žiakov tak, že počas bežnej výučby precvičíme podobné typy úloh, aké sú v teste. V kapitole 3 sme opísali naše prvé využitie tohto testu na strednej škole v jazyku Python. Na účely precvičenia jednotlivých typov úloh sme ku každej úlohe medzinárodného testu pripravili niekoľko vlastných alternatívnych úloh pre jazyk Python. Po jednej z týchto alternatívnych úloh sme uviedli v kapitole 2.2.

Medzinárodný test máme vo verziách pre jazyky C [6] a Python preložený do slovenčiny spolu s ďalšími informáciami na vyhodnotenie každej úlohy (riešenie a javová analýza). Tieto informácie zabezpečia pokiaľ možno rovnaký postup pri vyhodnocovaní testu, ak by sme chceli porovnať naše získané výsledky s benchmarkom uvedenom v [4].

V tomto školskom roku sme začali overovať využitie týchto úloh vo výučbe informatiky na gymnáziu a plánujeme s touto aktivitou ešte pokračovať. Medzinárodný test i alternatívne úlohy, ktoré sme k nemu vytvorili, plánujeme využiť vo vyučovaní programovania a didaktiky informatiky aj na našej fakulte v letnom semestri 2020/2021. Domnievame sa, že aj keď vzdelávacie štandardy pre stredné školy na Slovensku na vyučovanie tematickej oblasti Algoritmické riešenie problémov [15, 16] sa môžu odlišovať od štandardov v iných krajinách, môže priniesť práca s týmto testom učiteľovi informatiky ďalšie podnety pre svoju prácu. Žiakom/študentom môže priniesť účasť na teste v prípade úspešného vyriešenia motiváciu ďalej sa venovať programovaniu.

Ako sme uviedli vyššie, tento medzinárodný test nemá byť jediným nástrojom na hodnotenie pri výstupnej skúške z úvodného kurzu programovania, mal by tvoriť iba jej časť. Učiteľia, ktorí s ním už pracovali, mali vo výstupnej skúške zaradené aj ďalšie úlohy, ktoré korešpondujú s výučbou daného predmetu a zohľadňujú cieľové požiadavky na predmet. Záujemcovia o slovenskú verziu medzinárodného testu pre jazyk Python alebo pre jazyk C a pokyny na jeho vyhodnotenie nás môžu kontaktovať na e-mail jana.jackova@ku.sk.

POĎAKOVANIE

Ďakujeme Dr. Simonovi z The University of Newcastle, Australia, za poskytnutie anglickej verzie testu pre jazyk Python. Príspevok bol vytvorený v rámci riešenia projektu KEGA 018UMB-4/2020 Implementácia nových trendov v informatike do výučby algoritmického myslenia a programovania v predmete informatika v sekundárnom vzdelávaní.

BIBLIOGRAFICKÉ ODKAZY

- [1] KALAŠ, I., KABÁTOVÁ, M., PEKÁROVÁ, J., SLAVÍČKOVÁ, M., TOMCSÁNYI, P. *Základy pedagogického výskumu*. Bratislava : Štátny pedagogický ústav, 2011. ISBN 978-80-8118-082-8. Dostupné tiež na: https://www.statpedu.sk/files/sk/o-organizacii/projekty/projekt-dvui/publikacie/zaklady_pedagogickeho_vyskumu.pdf
- [2] ManagementMania.com. *Benchmarking* [online]. [cit. 2021- 01-29]. <https://managementmania.com/en/benchmarking>
- [3] SHEARD, J., SIMON, DERMOUDY, J., D'SOUZA, D., HU, M., PARSONS, D. Benchmarking a set of exam questions for introductory programming. In *Proceedings of the Sixteenth Australasian Computing Education Conference - Volume 148 (ACE '14)*. Darlinghurst : Australian Computer Society, 2014, p. 113 – 121. DOI: <https://dl.acm.org/doi/10.5555/2667490.2667504>
- [4] SIMON, SHEARD, J., D'SOUZA, D., KLEMPERER, P., PORTER, L., SORVA, J., STEGEMAN, M., ZINGARO, D. Benchmarking Introductory Programming Exams: Some Preliminary Results. In *Proceedings of the 2016 ACM Conference on International Computing Education Research (ICER '16)*. New York, USA : ACM, 2016, p. 103 – 111. DOI: <https://doi.org/10.1145/2960310.2960337>
- [5] ManagementMania.com. *Benchmark* [online]. [cit. 2021- 01-29]. <https://managementmania.com/en/benchmark>
- [6] JACKOVÁ, J., KANÁLIKOVÁ, A. Benchmarking v záverečných testoch z úvodných kurzov programovania. In *Obzory matematiky, fyziky a informatiky*, Roč. 46, č. 4 (2017), s. 1 – 12. ISSN 1335-4981.
- [7] LISTER, R., CLEAR, T., SIMON, BOUVIER, D. J., CARTER, P., ECKERDAL, A., JACKOVÁ, J., LOPEZ, M., MCCARTNEY, R., ROBBINS, P., SEPPÄLÄ, O., THOMPSON, E. Naturally occurring data as research instrument: analyzing examination responses to study the novice programmer. In *SIGCSE Bull.* 41, 4 (January 2010), p. 156 – 173. DOI: <https://doi.org/10.1145/1709424.1709460>
- [8] SIMON, SHEARD, J., D'SOUZA, D., LOPEZ, M., LUXTON-REILLY, A., PUTRO, I. H., ROBBINS, P., TEAGUE, D., WHALLEY, J. How (not) to write an introductory programming exam. In *Proceedings of 17th Australasian Computing Education Conference (ACE 2015)*. Sydney : Australian Computer Society, 2015, p. 137 – 146. Dostupné tiež na: <https://openrepository.auckland.ac.nz/handle/10292/9142>
- [9] JACKOVÁ, J., KANÁLIKOVÁ, A. Introductory Programming Exams and Their Benchmarking in Slovakia. In *17th International Conference on Emerging eLearning Technologies and Applications (ICETA 2019)*. IEEE, 2019, p. 279 – 284. <https://ieeexplore.ieee.org/document/9039988>
- [10] SIMON, SHEARD, J., D'SOUZA, D., KLEMPERER, P., PORTER, L., SORVA, J., STEGEMAN, M., ZINGARO, D. Benchmarking Introductory Programming Exams: How and Why. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '16)*. New York, NY, USA : ACM, 2016, p. 154 – 159. DOI: <https://doi.org/10.1145/2899415.2899473>
- [11] Gymnázium P. O. Hviezdoslava v Kežmarku. *Ponuka štúdia* [online]. [cit. 2021- 01-07]. <https://gymnaziumkk.sk/uchadzaci/ponuka-studia/>
- [12] Gymnázium P. O. Hviezdoslava v Kežmarku. *Rozvrh hodín* [online]. [cit. 2021-01-07]. <https://gymphkk.edupage.org/timetable/>
- [13] JACKOVÁ, Jana. *Mastery learning – od teórie k praxi*. Ružomberok : VERBUM – vydavateľstvo KU, 2020. 106 s. ISBN 978-80-561-0768-3.
- [14] ŠNAJDER, L., GUNIŠ, J. *Tvorba úloh a hodnotenie v predmete informatika*. Bratislava : Štátny pedagogický ústav, 2009. ISBN 978-80-8118-012-5. Dostupné tiež na: https://www.statpedu.sk/files/sk/o-organizacii/projekty/projekt-dvui/publikacie/tvorba_ulo_h_hodnotenie.pdf
- [15] Štátny pedagogický ústav. *Informatika – gymnázium so štvorročným a päťročným vzdelávacím programom* [online]. [cit. 2021- 01-07]. https://www.statpedu.sk/files/articles/dokumenty/inovovany-statny-vzdelavaci-program/informatika_g_4_5_r.pdf
- [16] Štátny pedagogický ústav. *Cieľové požiadavky na vedomosti a zručnosti maturantov z informatiky* [online]. Bratislava : ŠPÚ, 2019 [cit. 2021- 01-07]. https://www.statpedu.sk/files/sk/svp/maturitne-skusky/platne-od-sk-r-2018/2019/cp_informatika_2019.pdf

Súťaž iBobor pre nevidiacich žiakov nižšieho sekundárneho vzdelávania: Vedomosti a zručnosti potrebné na riešenie úlohy ako dôležitý faktor ovplyvňujúci jej obťažnosť

Bebras challenge for lower secondary blind students: Knowledge and skills needed to solve a task as an important factor influencing its difficulty

Eudmila Jašková
ODI KDMFI UK Bratislava
Mlynská dolina
851 04 Bratislava
Slovensko
jaskova@fmph.uniba.sk

ABSTRACT

A special category for lower secondary blind students has been in the Bebras challenge since 2013. During its existence we have adapted 63 tasks for blind students. But in many cases their difficulty was not correctly determined. We have analysed the influence of various factors on the real difficulty of tasks. We consider the knowledge and skills needed to solve the task to be the most important factor. However, it is necessary to take into account what is specified in the task assignment and what is the goal of the task. By analysing these assignment properties, we found several groups of tasks with common characteristics. It turned out that it is important not only what cognitive operations the competitor must perform to solve the task, but also such details as the number of actions required, the number of objects in the sequence, the size of the grid when programming the movement of the object and the like.

Keywords

Contest. Difficulty. Blind. Tasks. Factors.

ABSTRAKT

Od roku 2013 existuje v rámci súťaže iBobor kategória **Nevidiaci ZŠ**. Je to špeciálna kategória určená pre nevidiacich žiakov nižšieho sekundárneho vzdelávania. Počas siedmich rokov sme pre nevidiacich žiakov vytvorili, respektíve adaptovali už 63 úloh. Vychádzajúc z úspešnosti žiakov pri riešení viacerých úloh sme však museli skonštatovať, že ich reálna obťažnosť bola iná, ako sme pôvodne predpokladali. Analyzovali sme vplyv rôznych faktorov na reálnu obťažnosť úloh. Dôležité sú nepochybne vedomosti a zručnosti potrebné na riešenie úlohy. Tiež je potrebné si všimnúť vstupné dáta a podmienky určené v zadaní úlohy, ako aj vzdelávací cieľ. So zreteľom na tieto aspekty sme zoskupili úlohy s podobnými charakteristikami. Zistili sme, že nie sú dôležité len kognitívne schopnosti žiakov, potrebné na vyriešenie úlohy, ale aj na prvý pohľad možno menej podstatné detaily, ako počet požadovaných operácií, počet objektov v postupnosti, s ktorou sa

pracuje, veľkosť plochy pri programovaní pohybu objektu a podobne.

Kľúčové slová

Súťaž. Obťažnosť. Nevidiaci. Úlohy. Faktory.

1 ÚVOD

Súťaž iBobor netreba špeciálne predstavovať. Na Slovensku sa koná každoročne už viac ako 10 rokov. Jej poslaním je vzbudiť záujem študentov o informatiku a rozvíjať ich algoritmicke a logické myslenie [1]. Okrem toho si pri riešení úloh žiaci atraktívnym spôsobom osvojujú informatické koncepty a rozvíjajú informatické myslenie [2]. Americký matematik Carey [3] tvrdí, že testy a kvízy sú efektívnym nástrojom na učenie sa, ak študenti v krátkom čase získajú spätnú väzbu a dozvedia sa správne riešenie. Súťaž iBobor spĺňa tieto kritériá. Súťažiaci sa dozvedia svoje skóre bezprostredne po odoslaní riešenia a so vzorovými riešeniami s vysvetleniami postupu sa oboznámia v priebehu dvoch týždňov. Pri precvičovaní vedomostí a zručností je nápomocný aj archív úloh z predchádzajúcich ročníkov súťaže.

Súťaž iBobor nie je len nástrojom na učenie sa, ale aj na výskum a vyhodnocovanie informatických kompetencií osvojených žiakmi. Je to vďaka tomu, že môžeme získať údaje od veľkého množstva respondentov rôznych vekových kategórií, daností a schopností.

Vzdelávanie je základným právom každého človeka [4], preto je dôležité, aby sa aj žiaci s rôznymi druhmi postihnutia mohli zapojiť do súťaže určenej pre široké spektrum žiakov. Skupina nevidiacich žiakov je však špecifická, pretože títo žiaci používajú počítač iným spôsobom ako bežní žiaci. Prispôsobenia pravidiel a zadaní úloh určených pre nevidiacich sme predstavili v niekoľkých publikáciách [5, 6]. Stručne ich priblížime v nasledujúcej kapitole.

Kapitola 3 obsahuje prehľad niekoľkých výskumov zameraných na analyzovanie obťažnosti úloh súťaže iBobor určených pre intaktných účastníkov. Tieto výskumy boli pre nás inšpiráciou pre náš výskum, ktorý popisujeme v kapitolách 4 a 5. V poslednej kapitole zhrnieme naše zistenia a načrtujeme plány do budúcnosti.

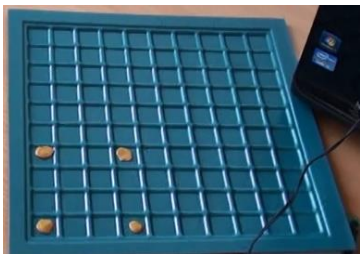
2 IBOBOR PRE NEVIDIACICH

Nevidiaci žiaci používajú pri práci s počítačom čítač obrazovky, ktorý im postupne interpretuje zobrazené informácie. Prístupné sú pre nich len informácie v textovom a zvukovom formáte. Obrázky musia byť slovné opísané. Na ovládanie počítača nepoužívajú myš, ale výlučne len klávesnicu. Musia teda poznať všetky potrebné klávesové príkazy. Takýto sekvenčný prístup k informáciám si väčšinou vyžaduje viac času. Úlohy v súťaži iBor obsahujú viacero prvkov, ktoré sú pre nevidiacich žiakov neprístupné. Sú to napríklad obrázky bez textovej alternatívy, objekty odlišené pomocou farieb, interaktívne prvky, s ktorými je možné manipulovať iba pomocou myši a iné. Bolo preto nevyhnutné vytvoriť pre nevidiacich žiakov špeciálnu kategóriu a mierne upraviť pravidlá [7, 8]. Vytvorili sme dve kategórie: jednu pre nevidiacich žiakov nižšieho sekundárneho vzdelávania (Nevidiaci ZŠ) a druhú pre nevidiacich žiakov vyššieho sekundárneho vzdelávania (Nevidiaci SŠ). Aby mali nevidiaci účastníci viac času na riešenie úloh, riešia 9 úloh (3 ľahké, 3 stredne ťažké a 3 ťažké) v časovom intervale 40 minút. Majú tak v priemere o 60% viac času ako intaktní žiaci, ktorí majú za rovnaký čas vyriešiť 15 úloh (5 ľahkých, 5 stredne ťažkých, 5 ťažkých).

V ďalšom texte sa zameriame na kategóriu Nevidiaci ZŠ, v ktorej sa dá súťažiť od roku 2013. V prvých štyroch ročníkoch existencie tejto kategórie mali účastníci zobrazené zadania úloh v textovom dokumente a nie vo webovom prehliadači, ako to bolo v prípade intaktných žiakov. Bolo to z toho dôvodu, že vtedajší nevidiaci žiaci vedeli používať textový editor bez problémov, ale s webovým prehliadačom nemali dostatočné skúsenosti. Webový prehliadač sa na zobrazenie úloh pre nevidiacich žiakov začal používať až od roku 2017, kedy sa na druhý stupeň ZŠ dostali žiaci, ktorí sa naučili používať webový prehliadač už počas primárneho vzdelávania. Používanie textového editora však bolo dovolené na zápis poznámok a jednotlivých krokov riešenia. Okrem toho môžu nevidiaci žiaci používať aj ceruzku a eurobal na kreslenie reliéfnych obrázkov (viď Obrázok 1) a reliéfnu tabuľku (Obrázok 2).



Obrázok 1: Fólia s reliéfny obrázkom, ktorý nakreslil nevidiaci žiak pri riešení úlohy.



Obrázok 2: Reliéfna tabuľka.

Úlohy pre nevidiacich žiakov [9] vznikli adaptáciou úloh určených pre ich intaktných rovesníkov (kategórie Benjamíni a Kadeti [10]). Pri adaptácii úloh sme sa snažili zachovať podstatu problému a vykonať len nevyhnutné zmeny. Niektoré úlohy sa dali použiť v pôvodnom znení, ale väčšina úloh si vyžadovala vykonať menšie, či väčšie úpravy nasledujúceho charakteru.

- Nahradiť obrázky obsahujúce relevantnú informáciu textom.
- Nepoužívať farby na odlišenie objektov.
- Používať tabuľky so správnym lineárnym poradím, prípadne ich nahradiť textom alebo zoznamom.
- Nepoužívať interaktívne úlohy, ktoré sa dajú riešiť len pomocou myši.
- Používať kratšie postupnosti objektov, pretože nevidiaci účastníci si musia tieto postupnosti zapamätať.
- Zmeniť úroveň obtiažnosti, ak je potrebné si pamätať väčšie množstvo informácií.

3 ŠTÚDIE ZAMERANÉ NA ANALÝZU OBTIAŽNOSTI ÚLOH

Ako sme uviedli v predchádzajúcej kapitole, žiaci riešia úlohy rôznej obtiažnosti – ľahké, stredne ťažké a ťažké. Úlohy s rozdielnou obtiažnosťou sú hodnotené rozdielnym počtom bodov. Je preto dôležité, aby autori úloh dokázali správne odhadnúť úroveň obtiažnosti. Analýze obtiažnosti úloh určených pre intaktných žiakov sa venovalo vo svojich článkoch množstvo autorov z rôznych krajín, v ktorých dlhoročne prebieha súťaž. Spomenieme niektoré z nich.

Piaty autori z piatich rôznych univerzít [11] analyzovali dáta o úlohách, ktoré v súťaži riešilo 115 400 študentov z ročníkov 3–12 v siedmich rôznych štátoch. Ich štúdiá poskytujú zaujímavý pohľad na celý rad otázok týkajúcich sa možného vplyvu rôznych školských systémov a pohlavia na úspešnosť účastníkov. Uvedomili si, že odhadovanie náročnosti nových úloh je otvoreným problémom. Podľa autorov je potrebné lepšie pochopiť a charakterizovať rozsah algoritmických stratégií použitých pri súťaži a faktorov, ktoré spôsobili veľké rozdiely v úspešnosti pri riešení úloh medzi rôznymi krajinami. Autori tvrdia, že koncepčný obsah nemá zásadný vplyv na náročnosť úloh.

Van der Wegt a kolektív analyzovali reálnu obtiažnosť úloh v rámci holandskej súťaže Bebras. Výsledky publikovali v niekoľkých štúdiách [12, 13, 14]. Využili rôzne nástroje pri predpovedaní náročnosti úloh. Pokúsili sa tiež analyzovať vzťah medzi náročnosťou obsahu, náročnosťou stimulov a náročnosťou úlohy. Dospeli k záveru, že obtiažnosť obsahu je najjasnejšou položkou pri predpovedaní obtiažnosti. Pomocou kvantitatívnych metód dokázali potvrdiť tendenciu, že úlohy týkajúce sa reprezentácií údajov a dátových štruktúr žiaci riešia lepšie ako úlohy týkajúce sa algoritmov a programovania.

Vaníček [15, 16] sa zaujímal o faktory, ktoré preukázateľne zvyšujú obtiažnosť úloh. Podľa neho to je formalizovaný zápis, úlohy obsahujúce štruktúry, optimalizačné úlohy a úlohy vyžadujúce schopnosť čítať s porozumením. Naproti tomu, nepotvrdilo sa, že by mali na vyššiu obtiažnosť úloh vplyv také faktory, ako je dĺžka zadania, použitie technickej terminológie, algoritmov, diagramov a záporných otázok.

Tomcsányi [17] zistil, že ťažké sú úlohy obsahujúce postupnosť príkazov v nejakom nekonvenčnom vymyslenom programovacom jazyku. Najjednoduchšie sa ukázali úlohy spojené s každodenným

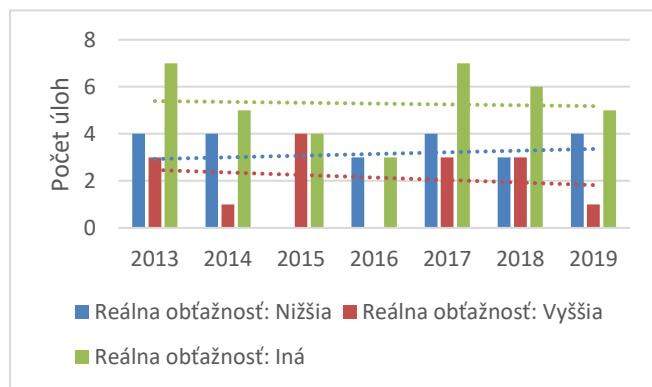
životom a úlohy, pri riešení ktorých majú študenti zistiť pôvodné informácie na základe šifrovacieho kľúča.

Tomcsányi a Tomcsányiová [18] zdôrazňujú, že účastníci sú často neúspešní pri riešení úloh preto, že nečítajú zadanie s porozumením a nedávajú si pozor, aby sa pri riešení úloh nedopustili chýb.

Všetky vyššie spomenuté štúdie sú pre nás cennou inšpiráciou pri skúmaní faktorov ovplyvňujúcich náročnosť úloh pre nevidiacich.

4 POPIS VÝSKUMU

Počas siedmich rokov konania súťaže pre nevidiacich žiakov a následného analyzovania výsledkov sme pozorovali, že reálna obťažnosť mnohých úloh sa líšila od predpokladanej obťažnosti. V priemere to platilo o viac ako 50% úloh (Obrázok 3, zelená bodkovaná čiara). Zistili sme tiež, že počet úloh s nižšou reálnou obťažnosťou ako sa očakávalo, z roka na rok mierne narastal (modrá bodkovaná čiara predstavuje trendovú priamku). Na druhej strane, počet úloh s ťažšou reálnou obťažnosťou, ako bola predpokladaná, mierne klesal (červená bodkovaná čiara predstavuje trendovú priamku). Zelená bodkovaná čiara predstavuje trendovú priamku pre úlohy, ktoré mali inú (nižšiu alebo vyššiu) reálnu obťažnosť, ako sa pôvodne predpokladalo. Vidíme, že táto priamka sa pohybuje okolo hodnoty 5, ale mierne klesá.

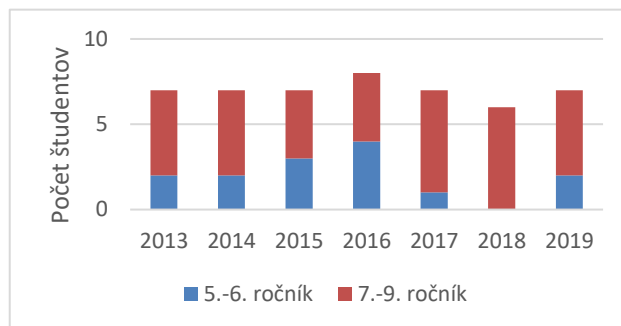


Obrázok 3: Graf ilustruje odchýlky reálnej obťažnosti úloh od predpokladanej.

Aby sme dokázali v budúcnosti lepšie predpovedať obťažnosť úloh, rozhodli sme sa preskúmať vplyv niektorých faktorov [19]. Považujeme za dôležité sa zmieniť o tom, že okrem iného sme skúmali aj také faktory ako je vek účastníkov, dĺžka zadania úlohy a príslušnosť úlohy k tematickému celku podľa platných osnov informatiky [20]. V tomto článku sa však obmedzíme len na *vedomosti a zručnosti potrebné na riešenie úlohy ako dôležitý faktor ovplyvňujúci jej obťažnosť*.

Použili sme stratégiu prípadovej štúdie, pričom jednotlivými skúmanými prípadmi boli samotné úlohy a ich zadanie (63 úloh). Pre každú úlohu sme vyjadrili reálnu obťažnosť Q ako percentuálny počet nesprávnych riešení a chýbajúcich riešení [17, 21]. Úlohy s hodnotou Q menšou ako 30 % sme vyhodnotili ako **ľahké**, úlohy s reálnou obťažnosťou od 30 % do 70 % ako **stredne ťažké** a úlohy s reálnou obťažnosťou od 70 % ako **ťažké**.

Pri určovaní reálnej obťažnosti sme brali do úvahy len výsledky žiakov jednej školy. Bolo to z toho dôvodu, že sme mali v tejto škole možnosť pozorovať žiakov počas reálneho behu súťaže. Tiež sme mohli so žiakmi a ich učiteľmi diskutovať o úlohách bezprostredne po skončení súťaže. Získali sme tak cenné výskumné dáta v podobe terénnych zápiskov a interview. Nevidiacich žiakov sme mali možnosť pozorovať aj na bežných hodinách informatiky a vedeli sme, že sa venovali všetkým tematickým celkom tak, ako sa to vyžaduje v ŠVP predmetu informatika. Priemerný počet nevidiacich žiakov, ktorí sa do súťaže zapojili býval každý rok v priemere od 6 do 8 nevidiacich žiakov (Obrázok 4) druhého stupňa ZŠ.



Obrázok 4: Počty nevidiacich účastníkov súťaže v jednotlivých ročníkoch súťaže.

Pripomíname, že graf na obrázku 4 reprezentuje len počty nami skúmaných nevidiacich účastníkov z jednej, nami zvolenej, školy. Pre úplnosť uvádzame, že v rokoch 2017 až 2019 sa súťaže zúčastnili aj nevidiaci a ťažko slabozrakí žiaci z iných škôl, či už študujúci segregovane v špeciálnych alebo integrovane v bežných školách (Tabuľka 1). O týchto žiakoch sa nám však nepodarilo získať podrobnejšie informácie.

Tabuľka 1: Celkový počet účastníkov v rokoch 2017 až 2019.

Rok	Segregovaní	Integrovaní	Spolu
2017	22	0	22
2018	20	3	23
2019	23	1	24

5 VÝSLEDKY VÝSKUMU

Pre každú zo 63 úloh sme určili jej reálnu obťažnosť Q a podľa získaných hodnôt sme úlohy rozdelili do troch skupín (ľahké, stredne ťažké, ťažké). Získali sme tak 26 ľahkých úloh, 20 stredne ťažkých a 17 ťažkých úloh. V rámci každej skupiny sme definovali zručnosti potrebné na ich vyriešenie.

Podľa niektorých autorov (Van der Wegt, Vaniček) sú vedomosti a zručnosti potrebné na riešenie úlohy dôležitým faktorom ovplyvňujúcim jej obťažnosť. Okrem toho je tiež dôležité, aké informácie sú dané v zadani a čo je cieľom úlohy. Analyzovaním týchto aspektov sme sa pokúsili nájsť niektoré charakteristiky úloh na jednotlivých stupňoch obťažnosti, ktoré uvádzame nižšie.

Na riešenie ľahkých úloh sú potrebné nasledujúce vedomosti a zručnosti:

- poznať pravidlá bezpečného používania technológií,
- vedieť vybrať objekt spĺňajúci špecifické požiadavky,
- poznať využitie periférnych zariadení,

- pracovať s hierarchickým usporiadaním objektov (maximálne tri úrovne hierarchie),
- vedieť nájsť najkratšiu cestu (medzi lineárne usporiadanými objektami v počte maximálne päť objektov),
- vedieť upravovať vstupný text pomocou postupnosti klávesov (oprava jednej chyby),
- vedieť zmeniť usporiadanie prvkov vstupnej postupnosti podľa požiadaviek (ak je potrebná jedna výmena),
- vytvárať kombinácie čísel spĺňajúcich dané požiadavky (maximálne štyri čísla a súčet je do pätnásť).

Na riešenie stredne ťažkých úloh sú potrebné nasledujúce vedomosti a zručnosti:

- vedieť upravovať vstupný text, ak je daný postup modifikácie (ak je modifikáciou otáčanie odzadu a potrebné sú maximálne dve modifikácie),
- použiť opakovanie postupnosti príkazov ovládajúcich pohyb objektu po štvorcovej sieti (bez otáčania objektu na mieste),
- vedieť zmeniť usporiadanie prvkov vstupnej postupnosti podľa požiadaviek (ak zisťujeme počet výmen a je možná výmena dvojíc),
- vytvárať kombinácie čísel spĺňajúcich dané požiadavky (viac ako štyri hodnoty a súčet väčší ako dvadsať).

Na riešenie ťažkých úloh sú potrebné nasledujúce vedomosti a zručnosti:

- vedieť upravovať vstupný text, ak je daný postup modifikácie (ak je modifikáciou otáčanie odzadu a aj posun písmen v abecede),
- použiť opakovanie postupnosti príkazov ovládajúcich pohyb objektu po štvorcovej sieti (s otáčaním objektu na mieste),
- použiť opakovanie postupnosti príkazov ovládajúcich pohyb objektu po štvorcovej sieti (s parametrom príkazu),
- vedieť zmeniť usporiadanie prvkov vstupnej postupnosti podľa požiadaviek (ak zisťujeme počet presunov na voľné miesto).

Môžeme si všimnúť, že úlohy s rôznym stupňom obťažnosti majú značne podobné charakteristiky. Pre ilustráciu uvádzame tri úlohy (Tabuľka 2), ktoré si vyžadujú podobné vedomosti a zručnosti, ale každá z nich má inú reálnu obťažnosť.

Všetky tri uvedené úlohy si vyžadujú zmeniť usporiadanie prvkov vstupnej postupnosti podľa požiadaviek. V úlohe **Zubné kefy** je potrebné určiť jednu dvojicu objektov, ktoré je potrebné vymeniť. V úlohe **Výmeny** bolo potrebné určiť minimálny počet výmen susedných objektov tak, aby boli tri objekty rovnakého typu vedľa seba. Po každej výmene teda stačilo overovať pozíciu troch objektov. V úlohe **Zlomená ruka** bolo potrebné určiť počet presunov, aby boli všetky objekty usporiadané podľa veľkosti. Po každom presune bolo potrebné skontrolovať polohu všetkých objektov.

Mohli by sme uviesť mnoho ďalších príkladov úloh s rôznym stupňom obťažnosti vyžadujúcich podobné informatické kompetencie, ale stanovený rozsah článku nám to nedovoľuje. Aj bez toho je však zrejmé, že na obťažnosť úloh majú vplyv aj nasledujúce charakteristiky:

- počet objektov v postupnosti,
- počet úrovní v hierarchii objektov,
- počet vykonávaných krokov v postupe,
- veľkosť štvorcovej siete, po ktorej sa pohybuje programovateľný objekt,
- prítomnosť parametra v príkaze,
- nutnosť predstavovať si natočenie objektu pri programovaní jeho pohybu.

Tiež je rozdiel, či sa v úlohe vyžaduje vykonať špecifikovanú postupnosť úkonov (ľahké) alebo je potrebné určiť ich počet (stredne ťažké), alebo je potrebné vykonávať určité výpočty počas vykonávania postupnosti krokov (ťažké).

Tabuľka 2: Úlohy s rôznym stupňom reálnej obťažnosti vyžadujúce podobné zručnosti.

<p>Zubné kefy (Ľahká úloha)</p> <p>Mama bobrica má štyri malé bobry: trojročnú Anitu, päťročného Bruna, sedemročného Cyrila a jedenásťročného Danku.</p> <p>Malé bobry majú zubné kefy zavesené v kúpeľni od najmenej po najväčšiu. Raz večer si ich omylom dali na nesprávne miesta. Keď prišla mama bobrica, kefy našla uložené takto: Anitina, Cyrilova, Brunova, Dankina.</p> <p>Ktoré dve kefy musí mama bobrica vymeniť, aby boli zavesené v správnom poradí?</p> <p>A. Brunovu a Dankinu B. Cyrilovu a Brunovu C. Dankinu a Anitinu D. Anitinu a Cyrilovu</p>
<p>Výmeny (Stredne ťažká úloha)</p> <p>Na dvore stoja zvieratká usporiadané v jednom rade takto: pes mačka myš pes mačka mačka pes Iba zvieratká, ktoré stoja vedľa seba si môžu vymeniť miesto. Najmenej koľko výmen je potrebných, aby všetci psi stáli tesne vedľa seba?</p> <p>A: 3 B: 4 C: 5 D: 6</p>
<p>Zlomená ruka (Ťažká úloha)</p> <p>Bobor David chce utriediť vetvičky, ktoré sú zoradené v rade vedľa neho tak, že najbližšie k Davidovi je vetvička so 4 lístkami a najďalej je vetvička s 3 lístkami. Nasledujúca postupnosť čísel reprezentuje zoradenie vetvičiek: 4 2 1 5 3</p> <p>Pomôž Davidovi zoradiť vetvičky podľa počtu lístkov tak, že najbližšie k nemu bude vetvička s 5 lístkami, vedľa nej so 4 lístkami, atď. a najďalej bude vetvička s jedným lístkom. David má zlomenú ruku, takže môže vždy presunúť len jednu vetvičku buď na pomocné miesto, ktoré si pripravil mimo radu alebo na uvoľnené miesto v rade vetvičiek.</p> <p>Aký je minimálny počet presunov, ktoré musí David spraviť?</p> <p>A: 4 B: 5 C: 6 D: 7</p>

6 ZÁVER

V článku sme sa venovali výskumu zameranému na analýzu obťažnosti úloh súťaže iBobor určených pre nevidiacich žiakov z hľadiska požadovaných informatických kompetencií. Podľa dosiahnutých výsledkov nevidiacich žiakov jednej školy sme určili reálnu obťažnosť všetkých 63 úloh, ktoré žiaci riešili v rámci súťaže realizovanej v rokoch 2013 - 2019. Následne sme analyzovali vplyv potrebných vedomostí a zručností na úroveň obťažnosti úloh.

Zistili sme, že nie sú dôležité len kognitívne schopnosti žiakov, potrebné na vyriešenie úlohy, ale aj počet požadovaných operácií, počet objektov v postupnosti, s ktorou sa pracuje, veľkosť mriežky pri programovaní pohybu objektu a podobne.

Sme si vedomí, že naše zistenia nemožno zovšeobecňovať, nakoľko sme sa obmedzili len na výsledky žiakov jednej školy. Bolo to z toho dôvodu, že sme mali možnosť na tejto škole dlhoročne sledovať priebeh súťaže iBobor. Tiež sme mali pomerne podrobné informácie o žiakoch, ako aj o priebehu výučby informatiky.

Slabou stránkou nášho výskumu je aj malý počet zúčastnených žiakov. V budúcnosti preto plánujeme overiť naše zistenia s väčším počtom nevidiacich žiakov z iných škôl.

Aj napriek uvedeným nedostatkom sme presvedčení, že uvedené zistenia nám pomôžu v ďalších ročníkoch vytvárať primerané úlohy pre nevidiacich žiakov a presnejšie predpovedať ich obťažnosť.

POĎAKOVANIE

Tento článok vznikol v rámci projektu financovanom agentúrou KEGA (Kega 018UK-4/2019). Radi by sme sa poďakovali učiteľom a žiakom so zrakovým postihnutím, s ktorými sme realizovali náš výskum.

BIBLIOGRAFICKÉ ODKAZY

- [1] DAGIENĚ V., FUTSCHEK G. Bebras International Contest on Informatics and Computer Literacy: Criteria for Good Tasks. In: Mittermeir R.T., Syslo M.M. (eds.) *Informatics Education - Supporting Computational Thinking*. ISSEP 2008. Lecture Notes in Computer Science, vol 5090. Springer, Berlin, Heidelberg (2008).
- [2] DAGIENĚ, V., STUPURIENĚ G. Exploring Bebras Tasks Content and Performance: A Multinational Study, *Informatics in Education*, Vol. 16, No. 1, 39–59, Vilnius University (2017).
- [3] CAREY, B. *How we learn*, Random House, New York (2014).
- [4] UN Convention on the Rights of Persons with Disabilities. United Nations, <http://www.un.org/disabilities/documents/convention/convoptprot-e.pdf>
- [5] JAŠKOVÁ, E., KOVÁČOVÁ, N. Bebras contest for blind pupils. In: *Proceedings of the 9th Workshop in Primary and Secondary Computing Education*, London, UK, ACM DL New York, NY, USA (2015).
- [6] JAŠKOVÁ, E., KOVÁČOVÁ, N. Contest for Blind Pupils – Universal Design of Tasks In: *Proceedings of the Conference Universal Learning Design*, Linz, 79–97, Masaryk University, Brno (2016).
- [7] ALLMAN, C. B. *Test Access. Making Tests Accessible for Students with Visual Impairments: A Guide for Test Publishers, Test Developers, and State Assessment Personnel*. 4th Release, American Printing House for the Blind, Louisville, Kentucky (2009).
- [8] Overview of exam access arrangements. http://www.rnib.org.uk/sites/default/files/Overview_of_exam_access_arrangements_May_2014.doc
- [9] iBobor pre nevidiacich. <http://vin.edu.fmph.uniba.sk/iBobor.html>
- [10] iBobor. <http://www.ibobor.sk>
- [11] IZU, C. et al., Exploring Bebras Tasks Content and Performance: A Multinational Study. *Informatics in Education*, 2017, Vol. 16, No. 1, 39–59, Vilnius University (2017).
- [12] VAN DER VEGT, W. Predicting the difficulty level of a Bebras task In: *Olympiads in Informatics*, Vilnius University, (2013), <https://ioinformatics.org/journal/INFOL127.pdf>
- [13] VAN DER VEGT, W. How Hard Will this Task Be? Developments in Analyzing and Predicting Question Difficulty in the Bebras Challenge. In: *Olympiads in Informatics*, Vilnius University (2018), https://ioinformatics.org/journal/v12_2018_119_132.pdf
- [14] VAN DER VEGT, W. et al. Analysing Task Difficulty in a Bebras Contest Using Cuttle, In: *Olympiads in Informatics*, 145-165 (2019).
- [15] VANÍČEK J. Bebras Informatics Contest: Criteria for Good Tasks Revised In: Gülbahar Y., Karataş E. (eds) *ISSEP 2014*. Lecture Notes in Computer Science, vol 8730. pp. 17–28, Springer, Cham (2014).
- [16] VANÍČEK J. What Makes Situational Informatics Tasks Difficult? In: Brodrik A., Tort F. (eds) *Informatics in Schools: Improvement of Informatics Knowledge and Perception*. ISSEP 2016. Lecture Notes in Computer Science, vol 9973. Springer, Cham (2016).
- [17] TOMCSÁNYI, P. Obťažnosť úloh v súťaži iBobor, In: *DidInfo 2009*, Univerzita Mateja Bela, Banská Bystrica (2009).
- [18] TOMCSÁNYIOVÁ, M., TOMCSÁNYI, P. Analýza riešení úloh súťaže iBobor v kategórii Benjamín v šk. roku 2012/13. In: *DidInfo 2013*, Univerzita Mateja Bela, Banská Bystrica (2013).
- [19] KOSTOVÁ, N. *Obťažnosť úloh v súťaži Informatický bobor pre nevidiacich*, diplomová práca, FMFI UK, Bratislava (2020).
- [20] JAŠKOVÁ, E., KOSTOVÁ N. (2020) Difficulty of Bebras Tasks for Lower Secondary Blind Students. In: Kori K., Laanpere M. (eds) *Informatics in Schools. Engaging Learners in Computational Thinking*. ISSEP 2020. Lecture Notes in Computer Science, vol 12518. Springer, Cham. https://doi.org/10.1007/978-3-030-63212-0_1
- [21] CHRÁSKA, M. *Metody pedagogického výskumu: Základy kvantitatívneho výskumu*, Grada, Praha (2007).

Čo sa skrýva v dátach? Dátová analýza a prezentácia dát (metodika pre SŠ)

What is hidden in the data? Data analysis and data presentation (methodology for high schools)

Alžbeta Kanáliková

KRIS, FEIT, Žilinská univerzita

Univerzitná 1

010 26 Žilina

Slovensko

alzbeta.kanalikova@feit.uniza.sk

ABSTRACT

The article presents a set of methodologies that focus on data, data analysis and data presentation. The set of methodologies was created within the innovative project of the IT Academy and also wants to contribute to innovative approaches to teaching informatics. The methodologies are intended for the third year of secondary schools and grammar schools for the subject of informatics or other related informatics subjects. The methodologies have been developed by the so-called "the research principle", which aims to transfer more activity to students. The students acquire knowledge and skills through research and find the answers to questions. The topic of data analysis, which the methodologies deal, is very important today. This topic, together with the "research principle" in methodologies, is one of the many innovative solutions that contribute to the IT Academy project.

Keywords

Data, Data analysis, Information, Knowledge, Inquiry-based instruction, Methodology, Data presentation.

ABSTRAKT

Článok predstavuje súbor metodík, ktoré sú zamerané na dáta, dátovú analýzu a prezentáciu dát. Súbor metodík vznikol v rámci inovatívneho projektu IT Akadémie a rovnako chce prispieť k inovovaným prístupom výučby informatiky. Metodiky sú určené pre tretí ročník stredných škôl a gymnázií pre predmet informatika alebo iné príbuzné informatické predmety. Metodiky sú vytvorené tzv. bádateľským princípom, ktorého cieľom je presunúť väčšiu aktivitu na študentov. Študenti získavajú vedomosti a zručnosti výskumom a nachádzajú odpovede na otázky. Téma analýzy údajov, ktorej sa metodiky venujú, je dnes veľmi dôležitá. Táto téma je spolu s „princípom výskumu“ v metodikách jedným z mnohých inovatívnych riešení, ktoré prispievajú k projektu IT Akadémia.

Kľúčové slová

Dáta, Dátová analýza, Informácia, Znalosť, Bádateľský princíp výuky, Metodika, Prezentácia dát.

1 ÚVOD

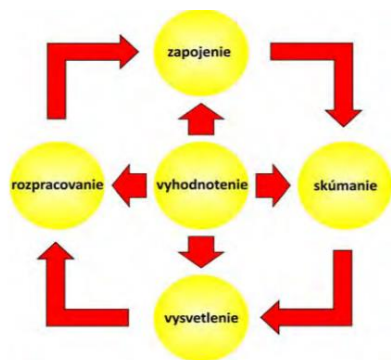
Dáta, ich analýza a ich prezentácia, resp. interpretácia sú veľmi dôležité súčasti informatiky. V tejto súvislosti je preto viac ako vhodné naučiť študentov stredných škôl dáta chápať, analyzovať,

vyberať z nich informácie a tvoriť znalosti pre rôzne účely. Cieľom článku je predstaviť súbor metodík pre informatiku.

Predstavovaný súbor metodík bol vytvorený v rámci projektu IT Akadémia a má za úlohu viesť študentov do sveta analýzy dát a ich interpretácie a prezentácie prostredníctvom známeho nástroja MICROSOFT EXCEL. Metodiky sú určené pre študentov tretieho ročníka stredných škôl. Pri aplikovaní metodík študenti prostredníctvom tzv. bádateľského (poznávacieho) princípu hľadajú odpovede na otázky, riešia úlohy a zadania čím si rozširujú nielen svoje vedomosti ale aj zručnosti. Súbor metodík je tvorený z piatich metodík, ktoré nasledujú za sebou. Metodiky si vyžadujú základnú znalosť nástroja MS EXCEL a tvorby prezentácie. Môžu byť vhodné aj pri výučbe nástroja MS EXCEL. Cieľom metodík je analýza a prezentácia dát a MS EXCEL je softvérovým nástrojom pri tejto analýze.

2 BÁDATEĽSKÝ PRINCÍP

Metodika, ktorá je predstavovaná v tomto článku je vytvorená prostredníctvom tzv. bádateľského princípu. Bádateľský princíp má názov od slova „bádať“, v slovníku slovenského pravopisu znamená skúmať, odborne, vedecky skúmať [1], t. z. hľadať odpovede na otázky, ktoré sa týkajú nových vecí. Teda ide o „bádanie“ v procese učenia, ktoré ako vedecké bádanie študentov podnieti, motivuje ku kladeniu otázok k danej téme, k hľadaniu odpovedí a zároveň k zdôvodneniu svojich zistení a obhajovaniu svojich argumentov [2]. Tento spôsob výučby je veľmi vhodný v prírodovedných predmetoch ako sú matematika, fyzika, chémia a aj informatika. Študent spoznáva, hľadá odpovede, skúša, testuje, skúma atď. a tým sa učí nové veci, nadobúda nové vedomosti. Viacerí autori hovoria ako napr. Prof. Douglas Llewellyn (ktorý opisuje šesť stupňový model bádania), že základom tohto princípu je konštruktivistický prístup. Konštruktivizmus vychádza z toho, že študent konštruuje poznatky na základe skúseností počas aktívnej činnosti. V anglicky hovoriacich krajinách je známy model tzv. 5E, t. z. *Engage* – Zapojenie, *Explore* – Skúmanie, *Explain* – Vysvetlenie, *Elaborate* – Rozpracovanie a *Evaluate* – Hodnotenie [3, 4]. Celý model je znázornený na obr. 1.



Obrázok 1: Pätfázový model výučby [3]

- **Zapojenie (Engage)** – Je úvodná fáza, kde učiteľ vzbudí záujem.
- **Skúmanie (Explore)** – Študenti sa zapoja do procesu bádania, na základe rôznych aktivít.
- **Vysvetlenie (Explain)** – Študenti sa sústreďujú na určitý aspekt, ktorý bol skúmaný. Učiteľ zdôrazní pojmy, poznatky, žiaci vysvetľujú napr. formou dialógu.
- **Rozpracovanie (Elaborate)** - Tu učiteľ pomáha študentom získané poznatky zovšeobecniť na nové situácie, nové príklady a pod.
- **Vyhodnotenie (Evaluate)** – Učiteľ hodnotí porozumenie jednotlivých pojmov, je to aj časť na otázky.

Uvedených päť fáz môže byť doplnených napr. o ďalšie dve fázy napríklad v úvode sa doplní ešte fáza *Elicit* – Získavanie a v závere sa doplní fáza *Extend* – Rozšírenie (v tomto prípade nazývame tzv. model 7E) [3]. Model piatich fáz je základom aj pre vytvorenú metodiku, ktorá je určená pre výučbu informatiky na strednej škole.

3 PREDSTAVENIE SÚBORU METODÍK

Súbor metodík dátovej analýzy a prezentácie dát bol vytvorený v rámci projektu IT Akadémia a ako už bolo spomenuté je určený pre výučbu informatiky v 3. ročníku stredných škôl.

Metodika využíva pre študentov už známy softvér MS EXCEL (predpokladom je, že študenti už ovládajú základnú prácu s tabuľkovým procesorom). Cieľom metodík je viesť študentov do analýzy dát, spoznať dáta, získať z dát informácie a znalosti pomocou ich analýzy prostredníctvom softvérového nástroja (MS EXCEL) a nakoniec tieto dáta vedieť správne interpretovať a prezentovať. Súbor metodík sa delí na 5 nasledujúcich metodík:

- **Dáta, získavanie dát, typy dát a jednoduchý výber dát.**
- **Formátovanie a vyhľadávanie dát.**
- **Dátová analýza I – Podmienky výberu dát.**
- **Dátová analýza II – Kontingenčné tabuľky.**
- **Prezentácia analýzy dát.**

Každá metodika je rozdelená podľa bádateľského princípu 5E do piatich častí. Časová dotácia každej metodiky je plánovaná na dve vyučovacie hodiny za sebou do týždňa z predmetu informatika (alebo príbuzných predmetov), t. z. celá časová dotácia je 10 vyučovacích hodín. V každej metodike je možné niektoré časti skrátiť a prispôbiť aj na jednohodinovú časovú dotáciu, ak učiteľ vyberie iba niektoré aktivity. Súbor metodík je vhodný aj pre špecializované odbory stredných škôl alebo gymnázií napr.

pre odbor informačné technológie na strednej škole, alebo technické gymnázia, lýcea a pod. V metodike sa nachádza dostatok zadanií a príkladov pre riešenie a učiteľ si môže hodiny prispôbiť pre prácu so študentmi v špecializovaných odboroch.

V každej z metodík sú aj krátke video návody pre študentov, ktoré vysvetľujú prácu s dátami v MS EXCEL (napr. filtrovanie dát, formátovanie dát, vytváranie kontingenčných tabuliek a pod.). Pre každú metodiku je pripravený pracovný list pre študentov, ktorý zahŕňa hlavne samostatnú, alebo skupinovú prácu študentov. Rovnako sú pripravené aj datasety (súbory s dátami, čiastočne spracovanými) pre študentov a učiteľov a pomôcky pre učiteľa – obrázky, tabuľky pre študentov a pod.

V každej z metodík sú uvedené ciele metodiky z hľadiska vedomosti, rozvíjané zručnosti študentov, definované didaktické problémy, prerekvizity potrebné na realizáciu metodiky (softvér, internet, súbory, pomôcky) a diagnostika cieľov.

V nasledujúcej kapitole budú podrobnejšie predstavené jednotlivé metodiky.

4 DÁTA, DÁTOVÁ ANALÝZA A PREZENTÁCIA DÁT

Metodiky sú usporiadané od definovania základných pojmov – dáta, informácie, znalosti, cez úpravu, analýzu dát až po interpretáciu dát a prezentáciu výsledkov analýzy dát.

4.1 Dáta

Názov prvej metodiky zo súboru metodík je: **Dáta, získavanie dát, typy dát a jednoduchý výber dát**. Táto metodika je úvodná a zaoberá sa pojmami: dáta, informácie a znalosti. Hlavným cieľom je prostredníctvom príkladov dát v priložených datasetoch pochopiť pojem dáta, rozdiel medzi dátami a informáciami a znalosťami, ďalej nastaviť dátam správny typ a upraviť dáta prostredníctvom automatických filtrov. Pribeh metodiky je nasledovný:

V úvode hodiny sú študenti zapojení prostredníctvom individuálnej práce a práce v skupinách do vyriešenia jednotlivých úloh, v ktorých majú hľadať význam dát a neskôr rozlíšiť či ide o dáta, informácie alebo znalosti. Na obrázku 2 je známa tzv. Znalostná pyramída, ktorá vysvetľuje hierarchiu medzi spomínanými pojmami (Dáta, Informácie, Znalosti).



Obrázok 2: Znalostná pyramída [5]

Študenti majú ako jednu z prvých úloh v tabuľke 1 napísať aké majú uvedené údaje význam a následne v časti „Skúmanie“ skúmať či daný údaj sú dáta, informácia, alebo znalosť:

Tabuľka 1: Úloha rozlíšiť dáta, informácie, znalosti

Uvedené údaje	Význam
10. 7. 2019	
8,00	
31	
31°	
Je teplo	

Učiteľ riadi diskusiu a odpovede študentov a neskôr ukáže možnosti rozlíšenia dát od informácií a znalostí prostredníctvom vhodných otázok.

V časti „Vysvetlenie“ sa rozvíja diskusia o možnostiach získavania dát a zdrojov dát a o type dát. Úlohou učiteľa je viesť diskusiu tak aby študenti sami pomenovali zdroje dát, t. z. odkiaľ a ako dáta získavame a aké typy dát môžeme získať (číselné, textové, časové a iné). V časti „Rozpracovanie“ študenti prostredníctvom krátkeho inštruktážneho videa získajú inštrukcie k výberu dát prostredníctvom automatických filtrov v MS EXCEL. V nasledujúcich zadaniach majú na priložených datasetoch (v súboroch) aplikovať automatické filtre. Počet úloh postačí na precvičenie ale rovnako aj na domácu úlohu.

4.2 Formátovanie a výber dát

Druhá metodika zo súboru metodík sa zaoberá ako už je z názvu zrejme **formátovaním a výberom dát**, resp. vyhľadávaním dát a informácií v existujúcich dátových súboroch. Študenti sa naučia dáta vyberať a vyhľadávať pomocou funkcií Excelu. Konkrétne sa používajú: podmienené formáty dát, funkcie na vyhľadávanie: HLOOKUP, VLOOKUP (horizontálne a vertikálne vyhľadávanie dát v existujúcej tabuľke). Priebeh metodiky je nasledovný:

V úvode hodiny si majú študenti vybrať z existujúcej tabuľky nameraných teplôt v Bratislave za určité časové obdobie informácie a znalosti, ktoré nájdu v tomto datasete. Následne učiteľ vedie krátku diskusiu. V časti „Skúmanie“ je pre študentov pridaný súbor s rovnakými dátami ale upravený prostredníctvom podmieneného formátovania. Študenti si majú prostredníctvom inštruktážneho videa naštudovať postup zapísania podmienok vo formátovaní a splniť zadanú úlohu. Časť z uvedeného súboru s podmieneným formátovaním je na obrázku 3.

Mesiac	Deň	Hodina	Minúta	Teplota
júl	8	0	0	24,23
júl	8	1	0	24,18
júl	8	2	0	23,86
júl	8	3	0	23,64
júl	8	4	0	23,42
júl	8	5	0	23,32
júl	8	6	0	22,51
júl	8	7	0	22,25
júl	8	8	0	22,01
júl	8	9	0	21,23
júl	8	10	0	20,72
júl	8	11	0	20,19
júl	8	12	0	19,59
júl	8	13	0	18,77
júl	8	14	0	18,56
júl	8	15	0	18,35
júl	8	16	0	17,77
júl	8	17	0	16,52
júl	8	18	0	15,63
júl	8	19	0	14,96
júl	8	20	0	14,37

Obrázok 3: Súbor dát s podmieneným formátovaním

V časti „Vysvetlenie“ je v metodike snaha o rozšírenie vedomostí o podmienenom formátovaní. Vysvetlené sú ďalšie možnosti podmieneného formátovania – vytváranie nových pravidiel. Následne študenti prostredníctvom inštruktážneho videa majú

naštudovať vyhľadávanie informácií v dátach. Video je návodom pre prácu s funkciami na vyhľadávanie HLOOKUP a VLOOKUP. Nasledujúca časť „Rozpracovanie“ obsahuje úlohy pre študentov, v ktorých si môžu študenti precvičiť vyhľadávanie prostredníctvom týchto funkcií. Záver hodiny je v metodike venovaný predstaveniu výsledkov úloh, ktoré študenti riešili v priebehu hodiny. V metodike sú k jednotlivým úlohám pripravené aj ďalšie súbory s dátami pre jednotlivé úlohy, ktoré môžu študenti riešiť.

4.3 Dátová analýza

Nasledujúce dve metodiky sa venujú dátovej analýze a výberu dát:

Prvá metodika dátovej analýzy má názov **Dátová analýza I** – Venuje sa výberu dát prostredníctvom podmienok a rozšírených filtrov.

Úvod tretej metodiky začína zadaním úlohy pre študentov, v ktorej majú zistiť ako pracuje funkcia IF (podmienky). V časti „Skúmanie“ študenti dostanú ďalšiu úlohu zapísať viacnásobnú podmienku, resp. zloženú podmienku prostredníctvom tzv. relačných operátorov <, >, =, <>. Príklad dát so zloženými podmienkami je na obrázku 4.

=IF([@[Vek zákazníka]]<25;"Mladý(<25)";IF([@[Vek zákazníka]]<35;"Mladý dospelý (25-34)";IF([@[Vek zákazníka]]<65;"Dospelý (35-64)";IF([@[Vek zákazníka]]>64;"Seniori(64+)"))

	D	E	F	G	H
	Zákazník	Vek zákazníka	Veková skupina	Pohlavie zákazníka	Krajina
3	11065	24	Mladý(<25)	F	Poľsko
5	11065	24	Mladý(<25)	F	Poľsko
5	11065	24	Mladý(<25)	F	Poľsko
3	11087	40	Dospelý (35-64)	F	Poľsko
3	11112	35	Dospelý (35-64)	F	Česká republika
5	11112	35	Dospelý (35-64)	F	Česká republika
4	11185	53	Dospelý (35-64)	F	Slovensko
6	11185	53	Dospelý (35-64)	F	Slovensko
4	11197	51	Dospelý (35-64)	F	Poľsko

Obrázok 4: Príklad výberu dát prostredníctvom logických podmienok

Tretia časť metodiky „Vysvetlenie“ rozširuje znalosti študentov formuláciou zložených podmienok na konkrétnych príkladoch. Výber dát je okrem podmienok doplnený aj o výber prostredníctvom rozšírených filtrov, ktoré sú rovnako definované prostredníctvom podmienok. Ako sa vytvárajú rozšírené filtre študentov inštruuje inštruktážne video. Príklad súboru s vytvorením rozšírených filtrov je na obr. 5.

A	B	C	D	E	F	G
Meno	Priezvisko	Vek	Znamenie	Auto	Farba očí	Stav
	ová	>30				
Vyberáme všetkých čo sú žena, resp. majú priezvisko so zakončením na ová a majú vek nad 30rokov						

Obrázok 5: Príklad výberu dát cez rozšírené filtre

V časti „Rozpracovanie“ sú pre študentov pripravené ďalšie príklady na realizáciu rozšírených filtrov. V časti „Hodnotenie“ sa študentom zopakujú možnosti výberu dát, ktoré sa študenti naučili. Na precvičenie sú na domácu úlohu pripravené pre študentov ďalšie úlohy.

Ďalšia metodika v poradí štvrtá, má názov **Dátová analýza II** - Je zameraná na výber dát prostredníctvom kontingenčných

tabuliek a vytváranie nových tabuliek dát prostredníctvom spájania dát z viacerých tabuliek podľa určitých kritérií.

V úvode tejto metodiky študenti dostanú súbor s kontingenčnou tabuľkou. Majú na začiatku zistiť čo je to kontingenčná tabuľka. V časti „Skúmanie“ majú študenti za úlohu zdefinovať pojem kontingenčná tabuľka podľa svojich zistení z internetu a iných zdrojov. Na zadanom príklade, ktorý dostali majú zistiť aký sa urobil výber dát. Ak vyjde čas môžu nájsť aj návod na vytvorenie kontingenčnej tabuľky. Všetky výsledky si študenti zapisujú do pracovného listu. V časti „Vysvetlenie“ učiteľ konfrontuje so študentmi čo je kontingenčná tabuľka. Postup pre praktické vytvorenie kontingenčnej tabuľky v MS EXCEL (výber menu) je v pracovnom liste. Ak študenti sami nenašli inštruktážne video ponúkne učiteľ link, ktorý je v metodike uvedený. Okrem kontingenčnej tabuľky upriami študentov aj na možnosť výberu dát do jednej tabuľky, ktorá vznikne prostredníctvom spájania viacerých tabuliek. V tomto prípade je rovnako k dispozícii inštruktážne video. Učiteľ by sa mal snažiť byť skôr ako sprievodca alebo poradca v tom, čomu študenti nerozumejú.

V časti „Rozpracovanie“ študenti skúšajú v úlohách urobiť rôzne výbery, napríklad vytvorenie viacerých kontingenčných tabuliek na rovnakých dátach, alebo spojiť dáta z viacerých tabuliek do jednej tabuľky a pod. Príklad výberu prostredníctvom kontingenčnej tabuľky je na obrázku 6.

Súčet z Objednávka v Eur	Označenia stĺpcov					
Označenia riadkov	Bratislava					
	Nitra	Poprad	Trnava	Zvolen	Celkový súčet	
borovica		21000	32500			53500
buk	13500	10500	27000			51000
dub	18000	25000	17000		17000	77000
hrab		5000		19000		24000
smrek	35500			6000	10000	51500
smrekovec		17000			35000	52000
Celkový súčet	67000	57500	65000	57500	62000	309000

Obrázok 6: Príklad výberu dát cez kontingenčnú tabuľku

V závere metodiky v časti „Hodnotenie“ študenti si svoje výsledky zo skupín hodnotia navzájom a následne ich prezentujú (napríklad dvojice, trojice). Systém hodnotenia študentov môže učiteľ premyslieť ešte dokonalejšie.

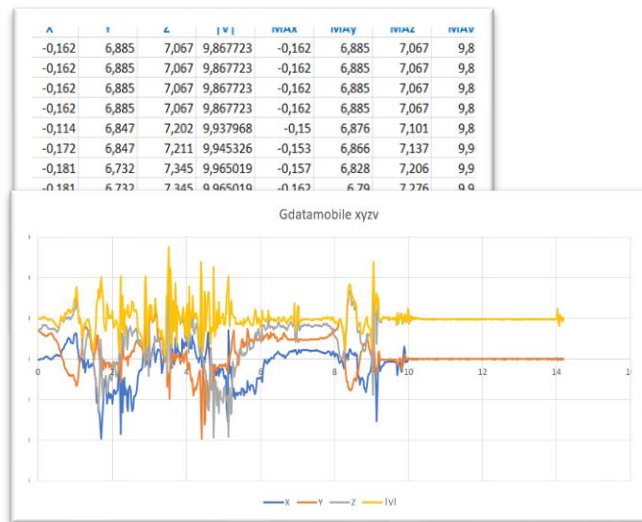
Metodika obsahuje aj ďalšie úlohy, ktoré si študenti môžu precvičiť doma a tak si upevniť svoje zručnosti výberu dát prostredníctvom kontingenčných tabuliek alebo výber dát prostredníctvom spájania tabuliek.

4.4 Prezentácia dát

Posledná metodika zo súboru metodík sa venuje prezentácii výsledkov z výberov a analýz dát prostredníctvom grafov a prezentácie (offline ale aj online).

V úvode tejto metodiky študenti dostanú súbor s reálnymi dátami z G - senzora, resp. z mobilnej aplikácie, ktorá prostredníctvom zabudovaného gyroskopu sledovala a zaznamenávala dáta. Tieto dáta sú v súbore zobrazené aj do grafu (súradnice x, y, z) a je možné vidieť pohyb mobilného telefónu, resp. režim kedy mobil bol iba položený. Študenti tieto dáta majú preskúmať a určiť polohu mobilu. V nasledujúcej časti „Skúmanie“ študenti dostanú ďalší súbor dát a majú preskúmať grafické vyjadrenie dát v tomto súbore. V pracovnom liste majú študenti tabuľku, v ktorej sú úlohy, ktoré majú zistiť a zaznamenať. V pracovnom liste je aj

konkrétny postup pre vytvorenie grafu (pre študentov by to už nemala byť nová informácia). V časti „Vysvetlenie“ učiteľ rozvíja teóriu o prezentácii, všetky informácie majú aj študenti v pracovnom liste – je to postup čo má prezentácia obsahovať, aké majú byť tabuľky, grafy a rovnako aj odporúčenia na čo je potrebné pri vytvorení prezentácie myslieť (napr. kde prezentujem, pre koho prezentujem, kedy prezentujem a hlavne čo je hlavné aby sa na dátach prezentovalo). Následne je vysvetlený postup vytvorenia prezentácie v dokumentoch GOOGLE v prípade ak prezentáciu chceme vytvoriť pre online priestor (pre vytvorenie prezentácie napr. v *Powerpointe* sa predpokladá, že študenti už znalosti, resp. zručnosti majú). V časti „Rozpracovanie“ študenti dostanú prístup k dátam na GOOGLE disku kde sú dáta zo senzora gyroskopu v mobile a v skupinách majú vytvoriť prezentáciu, ktorá bude prezentovať tieto dáta. V závere, resp. záverečnej časti „Hodnotenie“ učiteľ vyzve jednu skupinu študentov, aby prezentovala svoju prezentáciu o dátach z gyroskopu. V metodike je aj odporúčenie pre učiteľov zvážiť prezentovanie vlastných dát skupiniek študentov. Na prípravu prezentácie študenti potrebujú viac času a preto je vhodné ak prezentáciu odprezentujú, resp. vytvoria do ďalšieho týždňa (ak v plánovaní vyučovacích hodín je ešte možné venovať čas ďalších vyučovacích hodín). Študenti môžu použiť aj vlastné dáta z G-senzora, alebo aj iné nájdené dáta o to viac bude prezentácia zaujímavejšia. Na obrázku 6 je príklad dát z G – senzora, ktoré boli vygenerované mobilnou aplikáciou.



Obrázok 7: Dáta z G – senzora v tabuľke a grafe

Je na učiteľovi ako si prispôbí jednotlivé úlohy a ktoré úlohy využije.

5 ZHODNOTENIE A ZÁVER

Dáta sú všade okolo nás a ich počet dnes extrémne narastá. Existuje veľké množstvo zariadení, ktoré tieto dáta produkujú. Na druhej strane dáta sú veľmi zaujímavé ak ich vieme dobre spracovať, vybrať a zistiť z nich veľké množstvo informácií a znalostí. Z tohto dôvodu je potrebné viesť študentov k analýze dát a naučiť ich dátam porozumieť.

Tento článok prezentuje súbor metodík, ktoré sa snažia viesť študentov do sveta dát, ich spracovania, výberu a hľadania

informácií a znalostí praktickým spôsobom. Pre študentov je dôležité, aby boli motivovaní a sami za sprievodu učiteľa nadobudli základné zručnosti pri práci s dátami a pri ich analýze. Navrhovaný súbor metodík chce prispieť k rozvíjaniu informatického myslenia a k inovácií informatických predmetov. Metodika ešte neprešla komplexným hodnotením ale veríme, že po vyhodnotení a odstránení nedostatkov bude prínosom pre učiteľov vo vedení hodín a študentom v ich učení v predmete informatika a jej príbuzných predmetoch.

V každej z navrhovaných metodík je použitý bádateľský princíp, ktorý vedie hlavne k aktivite študentov. Študenti nie sú pasívni, musia hľadať informácie a rozmýšľať a to ich posúva ďalej, rozširuje ich vedomosti a zručnosti. Ako už bolo spomínané viackrát metodiku si môžu učitelia upraviť, využiť z nej iba niektoré časti, alebo úlohy. Ďalším možným rozšírením súboru metodík by bolo doplnenie súboru metodík napríklad o štatistickú analýzu dát, o exploratívnu analýzu dát alebo aj pri rozšírení softvérových nástrojov použiť analýzy, ktoré sa používajú v dataminingu.

BIBLIOGRAFICKÉ ODKAZY

- [1] Slovník slovenského pravopisu, elektronické vydanie,
<http://www.slex.sk/index.asp>
- [2] LUKÁČ, S., ŠNAJDER, E., GUNIŠ, J., JEŠKOVÁ, J.,
Bádateľsky orientované vyučovanie matematiky
a informatiky na stredných školách, vedecká monografia,
Košice 2016, Dostupné tiež na:
<https://unibook.upjs.sk/img/cms/2016/pf/bov.pdf>
- [3] KIREŠ, M., JEŠKOVÁ, Z., GANAJOVÁ, M.,
KIMÁKOVÁ, K., Bádateľské aktivity v prírodovednom
vzdelávaní, Štátny pedagogický ústav, Bratislava 2016,
Dostupné tiež na:
https://www.statpedu.sk/files/articles/nove_dokumenty/ucebnice-metodiky-publikacie/badatelске-aktivity/01cast_a_web.pdf
- [4] LIEWLLYN, D., Inquire within: Implementing inquirybased
science standards. Thousand Oaks, CA: Corwin Press 2002.
Dostupné tiež na:
<https://edis.ifas.ufl.edu/pdf/WC/WC07600.pdf>
- [5] GALLO, P. a kolektív, Znalostné systémy ako cesta
zvyšovania konkurencieschopnosti slovenskej ekonomiky,
Bookman, Prešov 2017, Dostupné tiež na:
http://www.dominanta.sk/Znalostne_systemy.pdf

Faktory zvyšujúce žiacky záujem o programovanie na kurze tvorby počítačových hier

Factors increasing pupils' interest in programming at a game-making course

Mgr. Mária Karpelová
Katedra didaktiky matematiky, fyziky
a informatiky Univerzity Komenského
Mlynská Dolina F1
842 48 Bratislava
Slovensko
karpelova2@uniba.sk

ABSTRACT

The objective of the paper is to present the preliminary results of a research among teachers of a game-making course which incorporated elements of gamification into its model. For the purpose of research, interviews were carried out with the most experienced teachers. The structured interview was focused on the course's model, on teaching programming as a part of a game-making process and the process how pupils gradually become autonomous when creating their own games. The main objective of the paper is to summarise teachers' experience, characterise factors improving pupil's interest in programming and what role the gamification elements play in the process according to teachers. The analysis of interviews identified that the four most important factors were the opportunity to improve, the creation of own games, interaction with others and motivation from gamification elements of the course's system.

Keywords

Game creation. Programming games. Pupils' interest in programming. Opportunity to improve. Interaction with others. Gamification.

ABSTRAKT

Zámerom príspevku je predstaviť priebežné výsledky výskumu, ktorý bol realizovaný medzi lektormi kurzu tvorby počítačových hier. Koncepcia modelu kurzu obsahuje niektoré prvky gamifikácie. Výskum bol realizovaný metódou rozhovorov so „služobne“ najskúsenejšími lektormi kurzu. Štruktúrovaný rozhovor bol zameraný na model kurzu, vyučovanie programovania v rámci tvorby hier a samotný proces, počas ktorého sa žiaci postupne osamostatnia pri tvorbe vlastných hier. Cieľom tohto príspevku je zhrnúť skúsenosti lektorov, vyšpecifikovať a charakterizovať faktory ovplyvňujúce záujem žiakov o programovanie a stanoviť akú úlohu podľa lektorov zohráva gamifikačný element. Analýza rozhovorov ukázala, že medzi štyri najvýznamnejšie faktory patria zlepšovanie sa, tvorba vlastných hier, interakcia s ostatnými a motivácia vo forme gamifikačných prvkov kurzu.

Kľúčové slová

Tvorba počítačových hier. Programovanie hier. Žiacky záujem o programovanie. Zlepšovanie sa. Interakcia s ostatnými. Gamifikácia.

1 ÚVOD

Pre všetky deti je hra vitálnou súčasťou života. Deti, ktoré dnes navštevujú základné školy, sú vystavované digitálnym médiám a technológiám viac než ktorákoľvek predchádzajúca generácia. Preto nie je prekvapením, že sú medzi nimi počítačové hry také obľúbené a že je mnoho snáh implementovať hry či herné prvky do vzdelávacieho procesu, aby ho obohatili. Pre vyučovanie programovania je hra výnimočná tým, že sama o sebe predstavuje komplexný funkčný algoritmus, ktorý žiaci môžu vďaka rôznym prostriediam vytvárať, upravovať či pozorovať, a tým sa učiť programovať v zmysluplnom vzdelávacom kontexte [1, 2]. Tvorba počítačových hier vytvára tiež podmienky, v ktorých žiaci prirodzene vyhľadávajú informácie [3] a nápadne sa podobá činnosti hrania hry či už tým, že normalizuje neúspech ako prirodzenú súčasť skúsenosti a berie ho ako zdroj informácií či spätnú väzbu alebo tým, že pri hraní hry aj pri tvorbe hier s časom prirodzene narastá náročnosť činnosti [4]. Viacerí edukátori a výskumníci poukazujú na rôzne benefity programovania počítačových hier alebo iných vlastných projektov v didaktických prostrediach; pomáhajú predstavovať a sprístupňovať žiakom základné programátorské koncepty [5, 6, 7], podporujú rozvoj výpočtového myslenia [8, 9], dizajnerskeho myslenia, herných gramotností [10], čo sú dôležité schopnosti potrebné pre úspech v digitálnej dobe, na ktorú máme žiakov pripravovať.

1.1 Zámer výskumu

Realizovaný výskum, priebežné výsledky ktorého chceme v tomto príspevku predstaviť, je súčasťou autorkinho dizertačného výskumu. Jeho zámerom je skúmanie poznávacieho procesu žiakov končiacich ročníkov základných škôl pri vyučovaní programovania tvorbou vlastných počítačových hier.

Hlavným cieľom tohto výskumu bolo lepšie porozumieť tomu, ako žiaci na kurze tvorby hier používajú návody pri učení sa, ako prebieha vzdelávací proces na tomto kurze od používania návodov ku konečnému osamostatneniu sa pri tvorbe hry a čo sú podľa lektorov významné časti tohto vzdelávacieho procesu. Keďže v rámci dizertačného projektu chceme tiež navrhnúť a vyvinúť vzdelávací kurz programovania prostredníctvom tvorby hier, veríme, že výsledky tohto kvalitatívneho výskumu poskytnú užitočný pohľad na vzdelávací proces typický pre takúto metódu vyučovania programovania. Taktiež veríme, že vďaka nim bude možné dôkladnejšie sa pripraviť na dizertačný výskum a zvoliť vhodný postup a výskumnú metódu.

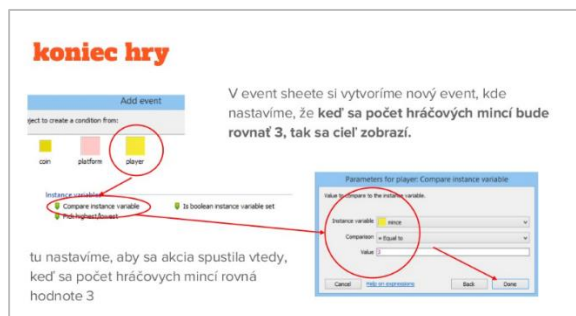
Pri analýze súboru kódov získaných z rozhovorov sa medzi témami opakovane vyskytovali isté faktory, ktoré podľa lektorov pozitívne prispievali k úspešnosti žiakov pri vzdelávaní sa na kurze GameCraft. Tento príspevok prezentuje práve výsledky tohto aspektu výskumu.

2 O KURZE GAMECRAFT

GameCraft je kurz vývoja počítačových hier určený pre žiakov vo veku 8 – 15 rokov, ktorý bol počas posledných piatich rokov organizovaný na viacerých základných školách v Bratislave a postupne sa rozšíril aj do Trnavy a Pezinka. V marci 2020 reagoval na zhoršujúcu sa pandemickú situáciu a presunul sa do online priestoru, vďaka čomu kurz mohol byť sprístupnený aj žiakom z iných častí Slovenska.

2.1 Priebeh kurzu

Kurzy prebiehajú v populudňajších hodinách a spolu trvajú dve hodiny. Skupinu tvorí maximálne 15 detí a dvaja školení lektori. Každá škola má vlastné špecifiká pre organizáciu žiackych kolektívov, no samotný kurz má všade rovnakú štruktúru; je rozdelený na dve 50-minútové pracovné úseky, medzi ktorými je 20-minútová prestávka. Počas pracovných úsekov si žiaci vyberajú svoju činnosť slobodne a môžu ju obmieňať. Na výber majú rôzne úlohy, ktoré sa charakterovo líšia podľa toho, na ktorý aspekt tvorby hier sú zamerané a ako je konkrétna úloha štruktúrovaná.



Obrázok 1: Ukážka z návodu k úlohe v Construct 2 [11]

Úlohy, ktoré predstavujú nový obsah alebo prácu s novým nástrojom sú sprevádzané návodom, ktorým sa žiaci riadia, čím sú zároveň vedení k samostatnej práci. Výsledné produkty jednotlivých návodov sú rôzne, no pri úlohách v rámci programovania ide často o naprogramovanie nejakého konkrétneho herného prvku alebo zjednodušenej verzie hry, ktorú žiaci často následne vylepšujú pridaním vlastnej hernej grafiky alebo ďalších herných prvkov. Pri tomto procese lektor môže pomôcť niečo vysvetliť alebo ukázať, no jeho hlavnou úlohou je žiakov sprevádzať v ich učení – pomáhať im pri prekonávaní prekážok a pri riešení problémov a konzultovať s nimi a hodnotiť ich nápady.

2.2 Model kurzu

Kurz je založený na individualizovanom vzdelávaní, pričom každý žiak pracuje vlastným tempom na zvolenej činnosti. Tieto činnosti sú zamerané na tri rôzne aspekty tvorby počítačových hier – na programovanie, tvorbu hernej grafiky a herný dizajn. Ak sa žiak venuje programovaniu hier, naučí sa tvoriť jednoduché hry v didaktickom prostredí Scratch, v hernom softvéri Construct 2 a eventuálne sa môže presunúť do Unity 3D alebo sa môže venovať základom tvorby webových stránok. Ak sa žiak venuje hernej grafike, zlepšovať sa môže v bitmapovej alebo vektorovej grafike, ktorú môže využiť v hrách. Herný dizajn je zameraný na získanie

pohľadu dizajnéra, žiak sa učí vytvárať príbehy alebo mechaniky do hier a balancovanie hry [12]. Žiaci si nemusia vybrať jednu cestu, ale väčšinou po čase začnú sami k niektorej z nich prirodzene inklinovať. Preto je na kurze podporovaná spolupráca na vlastných projektoch. Jednak sa takýmto spôsobom učia plánovať väčšie projekty či komunikovať v tíme, no zároveň sa žiaci môžu venovať hlavne tomu, čo ich na tvorbe najviac baví, bez toho, aby tým utrpel výsledok ich snaženia – implementovaná počítačová hra.

2.3 Gamifikácia v kurze

Obsah kurzu je zhromaždený v jednom online systéme. Každý žiak má do tohto systému vlastný prístup a prostredníctvom neho môže regulovať svoje štúdium, sledovať svoj pokrok a zdieľať projekty s ostatnými.

Jednou z charakteristík tohto kurzu je tiež využitie gamifikácie v systéme výučby. Kapp definuje gamifikáciu ako pridávanie herných princípov a elementov do neherných prostredí a situácií [13]. Môže ísť o rôzne herné prvky, no najčastejšie ide o tzv. *PBL* – point (body), badges (odznaky) a levels (levely) [14]. Tieto prvky sú prítomné aj v tomto systéme. Jednotlivé úlohy sú rôzne ohodnotené a keď ich žiak splní, získa zlatky (v systéme „coins“) a skúsenosti (v systéme „EXP“ z ang. *experience points*). Pomocou skúseností môžu žiaci a rodičia sledovať, ako sa posunuli v jednotlivých aspektoch tvorby – či pracovali na programovaní, grafike alebo dizajne a nazbierané zlatky si môžu vymeniť za hodnotné odmeny ako poukážky do rôznych obchodov alebo tričká s originálnymi motívmi.

Okrem úloh je možné v systéme sledovať rebríčky všetkých žiakov kurzov naprieč skupinami za určité obdobie podľa rôznych charakteristík. Aj za dobré umiestnenie v takomto rebríčku majú žiaci možnosť získať odmenu.

Ďalšou významnou časťou platformy je možnosť nahrávať dokončené projekty do systému. Týmto spôsobom žiaci z rôznych skupín môžu medzi sebou zdieľať, čo vytvorili a získať spätnú väzbu vo forme palca hore. Hra môže byť ocenená aj tak, že bude pridaná medzi obľúbené projekty v osobitnej časti platformy medzi ostatné úspešné projekty.

Napriek tomu, že pre dizertačný výskum bol najvýznamnejším aspektom kurzu proces vyučovania programovania a tvorby hier, bolo dôležité vnímať tieto gamifikačné prvky, keďže priamo súviseli so spôsobom organizácie kurzu.

3 METODOLÓGIA

Pre dosiahnutie cieľov priebežného výskumu sme sa rozhodli použiť metódu kvalitatívneho výskumu – štruktúrovaný rozhovor. Otázky boli zamerané na tri aspekty kurzu: na model kurzu a jeho zhodnotenie, ďalej na spôsob vyučovania programovania a fázy, ktorými žiaci prechádzajú pri učení a na samotný proces tvorby hier ako takej, ako prebieha a ako sa líši u začiatočníkov a u pokročilejších žiakov.

Všetky otázky boli otvorené a každá z otázok mala niekoľko podotázok. Rozhovory sa niesli v príjemnej atmosfére, a keďže autorka s občianskym združením zastrešujúcim tieto vzdelávacie kurzy spolupracuje už štvrtý rok, väčšinu respondentov poznala alebo spolu s nimi vyučovala na kurzoch, čo, veríme, že tiež prispelo k ich otvorenosti a úprimnosti počas rozhovorov a pravdivosti výpovedí.

3.1 Charakteristika respondentov

Vzorka respondentov bola zložená z 10 lektorov, ktorí buď učili na kurze dlho alebo mali skúsenosti s veľkým počtom skupín. Z týchto

lektorov boli dvaja lektori mladší než 20 rokov, piati boli starší než 20 a zároveň mladší než 26 rokov a dvaja boli starší než 26 (Tabuľka 1). Najstarší respondent mal 35 rokov a najmladší respondent mal 17 rokov a bol zároveň jeden z prvých, kto sa stal pomocným lektorom po tom, čo roky navštevoval GameCraft ako žiak. Medián je 23,5. Medzi respondentmi boli 7 muži a 3 ženy.

Tabuľka 1: Prehľad vekového rozloženia lektorov

Vek	Počet
Mladší než 20 rokov	2
21 – 26 rokov	5
Starší než 26 rokov	3

Pokiaľ ide o ich profesijnú orientáciu, šiesti z lektorov sa pohybujú v odvetví informatiky alebo tvorby hier; štyria z nich študujú alebo chcú študovať informatiku alebo tvorbu hier, jeden lektor vyučuje informatiku na základnej škole a jeden je junior tester vo firme, ktorá navrhuje mobilné hry. Ďalší štyria lektori sa pohybujú mimo tohto odvetvia, napríklad v štátnej správe alebo v službách.

Počas vyučovania na kurze sa lektori zvyknú vyprofilovať podľa toho, ktorý z aspektov tvorby preferujú alebo ktorý im je najbližší. Okrem dvoch, ktorí preferenciu bližšie nešpecifikovali, boli medzi nimi piati považujúci za svoju prednosť programovanie, dvaja herný dizajn a jeden herný grafiku.

Jednotlivé skupiny často združovali žiakov s rôznou úrovňou skúseností a schopností, keďže nový žiak mohol začať navštevovať kurz v akejkol'vek fáze roka. Preto bolo pre skúmanie vzdelávacieho procesu dôležité zistiť aj to, aké bolo zloženie žiakov v skupinách, ktoré učili. Zo žiakov zameraných na programovanie, mali úplných začiatokov vo svojich skupinách ôsmi z lektorov, všetci desiat lektori mali stredne pokročilých žiakov, ktorí sa práve začínali osamostatňovať a deviat lektori mali pokročilých žiakov, ktorí už pracovali úplne samostatne a tvorili vlastné hry. Zo softvérov na vyučovanie programovania všetci lektori pracovali so žiakmi v didaktickom prostredí Scratch, ako aj v hernom softvéri Construct 2 a štyria lektori pracovali so žiakmi aj v Unity 3D.

Pokiaľ ide o dĺžku skúseností s vyučovaním na kurze, najdlhšie vyučujúci respondent začal učiť v roku 2017/2018, ďalší traja lektori začali v roku 2018/2019 a šiesti lektori začali učiť v roku 2019/2020. Deviat respondentov v súčasnosti stále spolupracujú s kurzom GameCraft a jeden spolupracuje s občianskym združením v inom vzdelávacom projekte.

3.2 Zber a spracovanie údajov

Všetky rozhovory boli realizované online prostredníctvom videohovorov s jednotlivými lektormi a boli nahrávané. Po absolvovaní rozhovorov boli jednotlivé nahrávky doslovne prepísané do textovej formy. Pri referovaní k výpovediam sme sa snažili zachovať ich autenticnosť neupravením použitých kolokvializmov.

Ďalším krokom bolo manuálne kódovanie údajov v softvéri na kvalitatívnu analýzu údajov. Pri spracovaní údajov sme nemali preddefinované kódy, naopak, použili sme metódu otvoreného kódovania, kde jednotlivé kódy vznikali výskytom medzi výpovedami [15].

Následným kategorizovaním jednotlivých kódov medzi najvýznamnejšie kategórie vystúpila práve skupina kódov

charakterizujúca faktory, ktoré mali aktivizujúci efekt na žiakov počas priebehu kurzu alebo zvyšovali žiacky záujem o napredovanie v programovaní a tvorbe hier ako takej. Vzťahy medzi týmito faktormi sme interpretovali technikou „vyloženia kariet“ [16]. Vzťahy medzi týmito faktormi sme hľadali taktiež využitím myšlienkových máp. V ďalšej časti bude predstavená interpretácia výsledkov tejto analýzy.

4 VÝSLEDKY

Pri prvotnej analýze kódov vzniklo takmer desať rôznych kódov pomenúvajúcich faktory zvyšujúce žiacky záujem o programovanie alebo tvorbu hier, no hlbšou analýzou bolo možné všetky zahrnúť do jednej zo štyroch hlavných kategórií: zlepšovanie sa pre úspech, realizácia vlastnej hry, interakcia s ostatnými a gamifikačný systém. Prvé tri kódy sa vo výpovediach respondentov vyskytovali v podobnej miere; v Tabuľke 2 je možné vidieť, že sa najčastejšie vyskytol kód zlepšovanie sa pre úspech (19), ďalej interakcia s ostatnými (18) a za ňou realizácia vlastnej hry (17). Najmenší počet výskytov mal gamifikačný systém (9).

Tabuľka 2: Početnosť výskytu analyzovaných kódov v rozhovoroch s respondentmi



4.1 Zlepšovanie sa pre úspech

Kategória *Zlepšovanie sa pre úspech* by mohla byť opísaná slovom „učenie“. Môže ísť o učenie sa pomocou štruktúrovaných úloh a návodov, v rámci spolupráce medzi žiakmi alebo za pomoci lektora. Žiacky záujem je hnaný túžbou uspieť sám pred sebou. Takmer každý z respondentov sa vyjadril, že zlepšiť sa v niečom alebo niečo dokázať bol pre deti dôležitý moment – bez ohľadu na to, o aký veľký úspech išlo: „Také ľahké veci, ale deti to nadchnú, že to vedia spraviť.“

Podľa respondentov, túžba zlepšovať sa nebola charakteristická iba pre prirodzene nadanejšie deti, ale aj pre tých, ktorým to programovanie až tak nešlo, no chceli tomu venovať energiu: „Mala som aj chalana, ktorý aj na súradniciach (v Scratchi) strávil celú hodinu. Iné úlohy robil aj dve hodiny, mal problém to pochopiť. Ale snažil sa, naďalej to chcel robiť.“ Toto zlepšovanie môže smerovať k posunu do náročnejšieho programovacieho prostredia alebo k tvorbe vlastnej hry v prostredí, v ktorom sa učí. Iný respondent spomínal: „mal som žiaka, teraz je asi deviatka, on robil dosť dlho v Scratchi, a práve už chcel niečo lepšie, tak prešiel do Constructu (Construct 2). A teraz mu je už aj Construct málo a prechádza do Unity 3D.“ Títo žiaci „si chcú vyskúšať, čo všetko dokážu“, a to ich motivuje k ďalšej činnosti.

4.2 Realizácia vlastnej hry

Pri analýze kódov sa naprieč štyrmi rozhovormi objavovala rovnaká formulácia; že žiaci „tvorili niečo, čo bol ich sen“. „Chceli by naprogramovať niečo viac, ako len bloky. Celkovo ich zaujímajú tie hry a chceli by si vytvorili nejakú vlastnú“. Kategória *Realizácia vlastnej hry* by sa dala definovať slovom „tvorba“. Napriek tomu, že výskyt tohto kódu nebol najpočetnejší, podľa jedného z respondentov ide o faktor, ktorý má pre žiakov veľký význam: „Také dieťa som asi nemal, dieťa, ktoré by vedelo robiť vlastnú hru, že by sa späť vrátilo a celú hodinu robilo návody. Možno si spraví jednu úlohu, ale celkovo nie.“ Iný respondent zase poznamenal, že žiaci skôr stratia záujem, ak dlhodobo robia iba štruktúrované úlohy s návodmi, potrebujú zmenu činnosti, ale aj výzvu. A aj keď sa na prvý pohľad môže zdať, že ide o akýsi konečný cieľ, ktorý žiaci chcú dosiahnuť, v skutočnosti ide skôr o formu učenia; takú, kde majú žiaci možnosť robiť syntézu toho, čo sa naučili, s pravidlami hry, ktoré si sami zvolia, uplatňujúce postupy projektovej tvorby [17].

4.3 Interakcia s ostatnými

Interakcia s ostatnými bola jednou z kategórií, od ktorej sme očakávali, že sa preukáže ako dôležitá aj na základe autorkiných predchádzajúcich skúseností z učenia na kurze. Jedným z najčastejších javov bolo, že si žiaci navzájom testovali hry a prirodzene dávali a prijímali spätnú väzbu: „Pre decká je motivujúce, keď ich ostatné decká pochvália, a druhá vec je, že majú reálny obraz o svojom výtvore.“ Iný respondent spomenul: „počúvam, že sme to hrali s kamarátom a nevedeli sme to prejsť, tak sme to zmenili.“ Podobnú realitu opisovali aj iní respondenti, piati lektori tiež spomenuli, že takúto činnosť podporujú a zámerne pre ňu vytvárajú priestor na záver každého kurzu alebo keď žiak implementuje novú mechaniku do hry.

Viaceri respondenti tiež zhodnotili fakt, že kurzy často kombinovali skupiny žiakov s rôznymi schopnosťami, ako veľmi pozitívny. „Napríklad ich zaujme, že nejaký starší chalan ukazuje hru, čo vytvoril, tak tie deti sa občas prizerajú a sú zvedavé. A príde tá mladšia a spýta sa, ako to urobil, a starší príde a ukáže, ako to spravil.“ Iní respondenti poukázali tiež na podobný efekt prostredníctvom zdieľania projektov na spoločnej platforme: „Darmo ja budem rozprávať. Na to tam máme aj tie hry zverejnené, že si pozrú, čo niekto iný spravil, a ich to inšpiruje.“ Okrem inšpirácie to tiež prispieva k záujmu o napredovanie v programovaní. Jeden z respondentov zhrnul: „Keď tak nad tým rozmýšľam, veľa detí takto prešlo (zo Scratchu do Construct 2), že videli pokročilejších robiť niečo a tiež to chceli skúsiť.“

V neposlednom rade, podľa viacerých respondentov žiaci záujem často vzrastie v momente, keď si uvedomia, že môžu na projektoch pracovať spoločne: „Keď prišiel ten element, že s tromi kamarátmi robíme svoju hru, tak už to nerobili kvôli minciam.“

Túto kategóriu by sme mohli opísať slovom „reflexia“. V rámci komunity kurzu sú prirodzene vystavovaní fakt, že sa každý posúva a tvorí, čo sa preukázalo ako významná „hnacia sila“.

4.4 Gamifikačný systém

Ako sme uviedli v druhej kapitole, GameCraft je kurz, ktorý je spojený jednotným online systémom s gamifikačnými prvkami. Takmer každý z respondentov v nejakej časti svojej výpovede spomenul, že pre žiakov je významným aktivizačným faktorom to, že si plnením úloh môžu nazbierať dostatok mincí, aby si ich vymenili za hodnotnú odmenu. Dvaja z lektorov konkrétne poznamenali, že keď žiaci systém pochopia, snažia sa ho využiť:

„Deti to baví, keď majú motiváciu, že niečo za to získajú“. Splňa to teda očakávanú funkciu externej motivácie. V najčastejšom prípade išlo o to, že žiaci plnili úlohy, aby získali odmenu, a ďalší jav sa objavoval v skupinách pokročilejších programátorov, ktorých k činnosti nábádała túžba dokončiť všetky dostupné úlohy v systéme.

Zaujímavým javom bolo, že viacerí respondenti charakterizovali zmenu žiackej motivácie práve v spojení s gamifikačným systémom. „Stretnem sa s tým, ale tí starší to (mince) až tak neriešia. Nováčikovia to skôr riešia, s tým som sa stretol.“ Iný respondent zhodnotil: „Keď žiak chodí na kurz dlhšie, tak motivácia už je robiť vlastnú hru.“ Tieto a podobné výpovede nasvedčujú tomu, že gamifikačný systém kurzu môže fungovať ako oporný mechanizmus, ktorý ide do úzadia, keď žiaci získajú dost' schopností na to, aby tvorili vlastné hry. Sú však potrebné ďalšie údaje, aby sa mohol tento záver s určitosťou potvrdiť.

4.5 Identifikácia vzťahov

Pri postupnej analýze údajov sa začali ukazovať taktiež vzťahy medzi spomenutými faktormi; bolo možné vnímať, že zlepšovanie sa pre úspech, realizácia vlastnej hry a interakcia s ostatnými boli faktory, ktoré spolu tesne súviseli, zatiaľ čo, podľa dostupných údajov, gamifikačný systém vstupoval do vzťahu iba za určitých okolností. Ak žiaci prišli na kurz a jeden z prvých troch opísaných faktorov zvyšoval ich záujem, respondenti uvádzali, že význam gamifikačného systému bol vedľajší. A ak bola jedným z faktorov zvyšujúcich záujem možnosť využiť gamifikačný systém a získať odmenu, iba u malého počtu žiakov to tak ostalo dlhodobo. Keďže sme tento faktor s dostupnými údajmi nevedeli dostatočne vyhodnotiť, rozhodli sme sa ho do modelu nezahrnúť.

Vzťahy medzi zvyšnými tromi faktormi nepredstavujú cyklus. Javia sa byť obojsmerne spojené, a ide skôr o presúvanie sa medzi akýmisi fázami činnosti, kde záujem o programovanie nerastie iba v konkrétnych fázach, no aj pri presúvaní sa medzi nimi.



Obrázok 2: Model učenia sa tvorbou vlastnej hry postupným rozširovaným a prehlbovaným poznatkov z programovania

Vzťah medzi zlepšovaním sa pre úspech a realizáciou vlastnej hry sa ukázal ako prvý. Väčšina žiakov, aj keď nie všetci, začínajú na kurze práve tak, že plnia štruktúrované úlohy nasledovaním inštruktážnych návodov, čím si budujú a rozvíjajú vlastnú bázu schopností, ktoré neskôr vedia použiť pri tvorbe hry. Prirodzene, ako bolo spomenuté, nejde o jednosmerný posun od učenia k tvorbe; deje sa to aj opačne. Záujem o programovanie narastal u žiakov aj vtedy, keď si vymysleli novú mechaniku do vlastnej

hry, no museli prv zistiť, ako tá mechanika funguje a osvojiť si ju, kým ju mohli implementovať do svojej hry, čo vyžadovalo buď asistenciu od lektora alebo študovanie z dostupných návodov.

Nárast záujmu bol spozorovaný aj vtedy, keď žiaci dokončili vlastnú hru a chceli ju ukázať ostatným v skupine alebo ju zdieľať na spoločnej platforme. Prirodzene, v tomto smere išlo o vzrast záujmu na základe očakávania spätnej väzby na svoj projekt od lektora, no ešte častejšie od žiakov v skupine. Tento vzťah funguje aj opačným smerom, keď záujem o programovanie rástol pri možnosti spolupracovať s kamarátom na ich vlastnej spoločnej hre.

Zaujímavý vzťah bol tiež medzi sociálnym aspektom kurzu a nadobúdaním a rozvíjaním nových schopností. Žiaci záujem o programovanie vzrastá, keď vidia, kde sa môžu posunúť, ak sa budú snažiť, a naopak, v mnohých prípadoch práve pomoc od skúsenejších programátorov vedela povzbudiť menej skúseneho žiaka, aby na sebe ďalej pracoval.

5 ZÁVER

Zámerom tohto výskumu bolo získanie informácií o tom, ako prebieha učenie na kurze vývoja počítačových hier GameCraft. Považovali sme to za prvý krok, ktorý mal pomôcť lepšie pochopiť, akú úlohu zohráva tvorba počítačovej hry vo vyučovaní programovania, aby sme vedeli dôslednejšie postupovať pri dizertačnom výskume.

Pri analýze údajov z rozhovorov s lektormi bolo možné registrovať isté opakujúce sa typy činností, ktoré zvyšovali žiakov záujem o programovanie a tvorbu hry ako takú. Tieto faktory sme identifikovali ako zlepšovanie sa pre úspech, realizácia vlastnej hry, interakcia s ostatnými a gamifikačný systém. Hlbšou analýzou bolo možné objaviť vzťahy medzi tromi hlavnými faktormi, ktoré sa javili podobne významné, a rozhodnili sme, že záujem žiakov nevzrastal iba pri týchto troch typoch činností, ale tiež pri presúvaním sa medzi nimi. Presné miesto gamifikačného systému sa v tomto modeli nepodarilo s istotou určiť, no ukazuje sa skôr ako oporný mechanizmus, ktorý pomáhal udržať žiakov záujem, kým prerastie do jedného z troch prepojených faktorov.

Analýza tejto kategórie údajov pomohla identifikovať faktory, ktoré musíme zohľadniť pri navrhovaní pedagogickej intervencie. Pri tejto analýze bola tiež viackrát spomenutá dôležitosť návodov, najmä vo vzťahu medzi učením a tvorbou hier. V ďalšej časti analýzy rozhovorov, ktoré plánujeme v najbližšej budúcnosti realizovať, by sme sa preto chceli zamerať práve na celkový význam návodov v procese učenia, čo by malo pomôcť lepšie navrhnúť vlastné kvalitné materiály pre dizertačný výskum.

POĎAKOVANIE

Chceli by sme sa poďakovať Hemisfére ľavej o.z. a ich lektorom za umožnenie výskumu a tiež projektu VEGA 1/0602/20, vďaka ktorému mohli byť výsledky publikované.

BIBLIOGRAFICKÉ ODKAZY

- [1] STOFFOVÁ, V., HORVÁTH, R.: *Didactic computer games in teaching and learning process*. Else Bucurest, The 13th International Scientific Conference, eLearning and Software for Education, Bucharest, April 27-28, 2017, DOI: 10.12753/2066-026X-17-00.
- [2] STOFFOVÁ, V. *Educational Computer Games in programming Teaching and Learning..* In: New technologies and redesigning learning spaces : eLearning and Software for Education. Bucuresti : Carol I National Defence University, 2019. ISSN 2066-026X, CD-ROM, p. 39-45. WoS. DOI 10.12753/2066-026X-19-004.
- [3] BAYTAK, A., LAND, S. M. *An investigation of the artifacts and process of constructing computers games about environmental science in a fifth grade classroom*. In: Educational Technology Research and Development, 59(6), 765-782, 2011, DOI: 10.1007/s11423-010-9184-z.
- [4] WEINTROLP, D., WILENSKY, U. *Playing by Programming: Making Gameplay a Programming Activity*. In: Educational Technology, 2016. Dostupné na internete: <https://ccl.northwestern.edu/2016/playingbyprogramming.pdf>.
- [5] PEPPLER, K – KAFAI, Y. B. *What videogame making can teach us about literacy and learning: Alternative pathways into participatory culture*. In: Situated Play: Proceedings of the Third International Conference of the Digital Games Research Association, 369-376, 2007.
- [6] CZAKÓOVÁ, K., STOFFOVÁ, V. *Training Teachers of Computer Science for Teaching Algorithmization and Programming*. In: Proceedings of The 14th International Multi-Conference on Society, Cybernetics and Informatics (IMSCI 2020) Florida p. 231 -235, ISBN: 978-1-950492-46-6.
- [7] WERNER, J. et. al. *Can middle-schoolers use Storytelling Alice to make games?: Results of a pilot study*. In: Proceedings of the 4th international conference on foundations of digital games, 207-214, 2009, DOI: 10.1145/1536513.1536552.
- [8] MORENO-LEON, J. et. al. *Towards Data-Driven Learning Paths to Develop Computational Thinking with Scratch*. In: IEEE Transactions on Emerging Topics in Computing, 2017.
- [9] SPIELER, B. ET. AL. *The Learning Value of Game Design Activities: Association between Computational Thinking and Cognitive Skills*. 2020. DOI: 10.1145/3421590.3421607.
- [10] SALEN, K. *Gaming Literacies: A Game Design Study in Action*. In: Journal of Educational Multimedia and Hypermedia, 16(3), 301-322, 2007.
- [11] KARPIELOVÁ, M. *Ako na Construct 2*. 2020. Dostupné na internete: <https://akonaconstruct2.webnode.sk/>
- [12] *Obsah Kurzu*. Hemisféra, 2020. Dostupné na internete: <https://hemisfera.sk/gamecraft/obsah-kurzu> [citované 26. 1. 2020].
- [13] KAPP, K. M. *The gamification of learning and instruction: game-based methods and strategies for training and education*, San Francisco, CA: Pfeiffer.
- [14] KOŽLEJ, J. *Gamifikácia vo vyučovaní programovania a informatiky* [Diplomová práca]. Trnavská Univerzita v Trnave, 2020. 73 s.
- [15] STRAUSS, A., CORBINOVÁ, J. *Základy kvalitatívneho výskumu: Postupy a metódy techniky zakotvené teorie*. Brno: Sdružení Podané ruce, 1999. 105 s. ISBN 80-85834-60.X.
- [16] ŠVAŘÍČEK, R., ŠEĎOVÁ, K. *Kvalitatívni výzkum v pedagogických vědách*. Praha: Portál, 2007. ISBN 978-80-7367-313-0.
- [17] JONES, B. F. *Real-life problem solving: A Collaborative Approach to Interdisciplinary Learning*. In: Psychology in the Classroom Series [print]. Washington DC: American Psychological Association, 1997. 246 s. ISBN: 978-1-55798-294-0.

Tvorivosť v kontexte projektového vyučovania u budúcich učiteľov informatiky

Creativity in the Context of Project Teaching for Pre-service Informatics Teachers

Nika Klimová, Gabriela Lovászová

Katedra informatiky, Univerzita Konštantína Filozofa v Nitre

Tr. A. Hlinku 1

949 74 Nitra

Slovensko

nika.klimova@ukf.sk, glovaszova@ukf.sk

ABSTRACT

Creative-humanistic teaching is a modern concept of teaching that contributes to the democratization of the education by emphasizing the development of the student's personality. We deal with the issue of preparation of pre-service Informatics teachers with the project-based method following the creative-humanistic concept of teaching. Three case studies from the university seminar Projects in School Informatics with a focus on the identification of creative elements in project-based teaching are presented. In all cases, a high level of creativity was identified. The students showed specific features of creativity from the position of students and teachers.

Keywords

Creativity. Project-based teaching. Pre-service Informatics teachers.

ABSTRAKT

Tvorivo-humanistické vyučovanie je moderná koncepcia vyučovania, ktorá prispieva k demokratizácii školského vzdelávania zdôrazňovaním komplexného rozvoja osobnosti žiaka. V článku sa venujeme otázke prípravy budúcich učiteľov informatiky na uplatňovanie projektovej metódy v súlade s tvorivo-humanistickou koncepciou vyučovania. Prezentované sú tri prípadové štúdie z vysokoškolského seminára Projekty v školskej informatike so zameraním na identifikáciu prvkov tvorivosti v projektovom vyučovaní. Vo všetkých troch prípadoch bola identifikovaná vysoká úroveň tvorivosti úloh. Pri ich riešení študenti prejavili špecifické znaky tvorivosti z pozície žiaka aj učiteľa.

Kľúčové slová

Tvorivosť. Projektové vyučovanie. Budúci učelia informatiky.

1 ÚVOD

Moderné filozofie výchovy a vzdelávania zdôrazňujú komplexný rozvoj osobnosti žiaka – okrem získavania vedomostí a zručností aj personálny rozvoj, socializáciu, enkulturáciu. V slovenskom kontexte je predstaviteľom tohto trendu model tvorivo-humanistickej výchovy [1]. Na strane non-kognitívnych procesov kladie na vrchol humanizáciu osobnosti, človek vníma lásku a dobro ako základné životné hodnoty. Na strane kognitívnych procesov je na vrchole tvorivé myslenie. Tvorivý človek vníma svet okolo seba ako neustále sa meniaci otvorený systém, ktorý je možné vďaka tvorivosti zdokonaľovať.

Veľký potenciál na rozvoj tvorivého myslenia a humanizáciu vzdelávania má projektové vyučovanie. Jeho základom je americká

filozofia vzdelávania Johna Dewey z 30. rokov 20. storočia zdôrazňujúca demokratizáciu vyučovacieho procesu a praktické, autentické učenie sa zo skúseností [2]. Vzhľadom na to, že žiaci v ňom majú možnosť nazrieť do praktického života a riešiť problémy z neho vyplývajúce, je medzi žiakmi často obľúbené. Ešte viac na svojej atraktivite nadobudlo s rozvojom digitálnych technológií, ktoré poskytujú užitočné nástroje na tvorbu projektov.

Aby bola projektová metóda vo vyučovaní efektívne implementovaná, je dôležité sa ňou zaoberať už pri príprave budúcich učiteľov. V štúdiu predstavujeme niekoľko príkladov práce na projektoch v rámci vysokoškolského seminára Projekty v školskej informatike pre budúcich učiteľov informatiky na Univerzite Konštantína Filozofa v Nitre. Dôraz je kladený na tvorivosť v projektovom vyučovaní.

2 PROJEKTOVÉ VYUČOVANIE

Základným stavebným prvkom projektového vyučovania je projekt. Ide o sofistikovanú školskú úlohu, ktorá je zameraná na praktické využitie, vedie k tvorbe tvorivého riešenia alebo osobitného produktu a vyžaduje od žiaka autorský vklad [3]. Projektové vyučovanie by malo spĺňať tieto charakteristiky [4]:

- je prítomný problém, ktorého riešenie je riadiacim motívom aktivít žiaka,
- žiak ako výsledok vytvára konkrétny produkt (artefakt),
- žiak preberá zodpovednosť za splnenie úlohy,
- projekt často obsahuje medzipredmetové znaky,
- proces práce na projekte je slobodný – žiak si vyberá, ako bude zadaný problém riešiť,
- ak sa projektové vyučovanie uplatňuje aj v skupinovej práci, žiaci majú voľbu, ako sa budú rozhodovať a akým činnostiam sa budú venovať (často dochádza k del'be práce).

Úloha učiteľa ako vzdelávacej autority je v projektovom vyučovaní potláčaná do úzadia. Čapek na ilustráciu uvádza metaforu, že učiteľ je pri projektovom vyučovaní len spolujazdcom, ktorý nehovorí „šoférovi“ do riadenia a skôr len upozorňuje na niektoré zákazy vjazdu a prednosti v jazde [3]. To však neznamená, že jeho úloha nie je významná, mení sa len jeho pozícia z autority na partnera. Učiteľ vystupuje ako zadávateľ projektu, ktorý [4]:

- nie je zainteresovaný, akým spôsobom bude problém vyriešený, ale má záujem o výsledný produkt (artefakt),
- umožňuje žiakom podieľať sa na podrobnejšej špecifikácii zadania,

- stanovuje termín odovzdania výsledného produktu, čo vo vyučovaní znamená pevný časový harmonogram práce na projekte.

Okrem roly zadávateľa projektu učiteľ naďalej zostáva v reálnej roly učiteľa, ktorý aktívne podporuje prácu žiakov a je zodpovedný za plnenie výchovno-vzdelávacích cieľov vyučovania.

Projekty môžu byť navrhované učiteľmi, kedy ide o tzv. umelé projekty, ale aj žiakmi, teda spontánne žiacke projekty [5]. Takisto existuje aj kombinácia, kedy žiaci a učelia navrhujú projekt spoločne. V závislosti od cieľa môžu byť problémové, kedy cieľom je vyriešiť nejaký problém; konštruktívne, kedy sa artefakt vytvára alebo konštruje; hodnotiace, ktorých cieľom je skúmať a porovnávať; drilové, ktoré nacvičujú nejakú zručnosť. Okrem toho sa projekty môžu rozdeľovať na individuálne, skupinové či krátkodobé a dlhodobé, alebo školské, domáce a kombinované.

Otázkou projektového vyučovania a jeho princípmi sa zaoberá aj Kosová, ktorá zdôrazňuje, že [6]:

- je potrebné pri výbere témy projektu prihliadať na záujmy a potreby žiakov, pričom by mali vplývať na výber a špecifikáciu témy projektu,
- je vhodné, aby projekt súvisel aj so žiackou mimoškolskou skúsenosťou a ich zážitkami, pričom otvára školu širšiemu okoliu, spoločenstvu, ale aj osobnej situácii žiaka a riešeniu jeho problémov. Projekt by mal byť mostom medzi životom a školou a nemal by vytvárať len fiktívnu, zdanlivú a umelú realitu pre predpísané učivo,
- dobré projektové vyučovanie vychádza z vnútornej motivácie, záujmu a zainteresovanosti žiakov,
- projekty by mali prekračovať rámec jednotlivých vyučovacích predmetov a zabezpečiť interdisciplinárny prístup s komplexným pohľadom na skutočný svet,
- v projektoch by mala dominovať kolaborácia, teda spolupráca v skupinách,
- projekt by mal žiaka viesť ku konkrétnym výsledkom, na základe ktorých si žiaci osvojujú príslušné vedomosti, zručnosti, postoje či kvalifikáciu, pričom môže byť prítomná odmena.

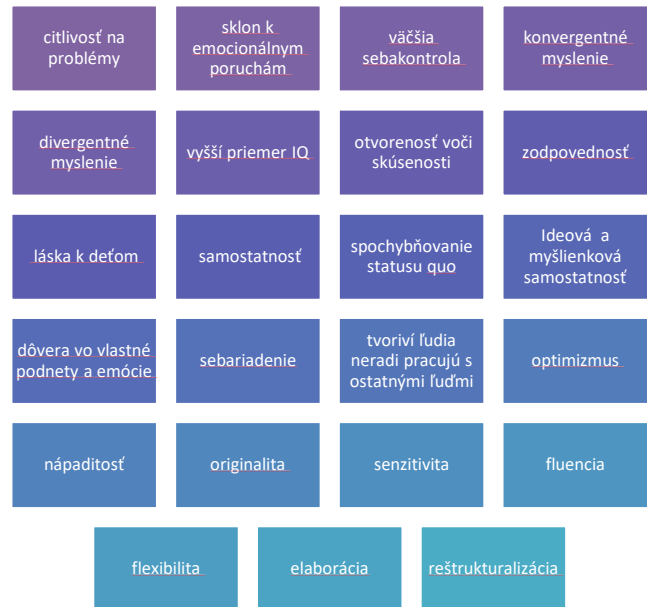
Projektové vyučovanie má svoje miesto aj v informatike. Príklady aplikácie projektového vyučovania možno nájsť v [7].

3 TVORIVOSŤ

Tvorivosť je odvodená z latinského slova creatio, čo znamená tvorbu. Existuje množstvo definícií, čo je to tvorivosť, my súhlasíme s tvrdením Portika, že ide o schopnosť tvoriť, ktorá existuje v potenciálnom stave u každého jedinca a v každom veku; aktivita, ktorá prináša zatiaľ neznáme a súčasne spoločensky hodnotné výtvory cenne nielen pre tvorcu, ale aj spoločnosť; duševná schopnosť, ktorá vychádza z poznávacích a motivačných procesov, ktoré v sebe zahŕňajú inšpiráciu, fantáziu, intuíciu; môže ísť o schopnosť vymyslieť niečo nové, postoj prijať niečo nové, napríklad zmenu, proces charakterizovaný prácou, systematickou myšlienkovou aktivitou tvorby nových riešení [8].

Osobnosť učiteľa by mala zahŕňať množstvo kladných vlastností, medzi ktoré patrí aj tvorivosť, pretože „učiteľ nemôže postupovať vo svojej práci podľa „hotových receptov“, „kopírovať“ návody a postupy iných učiteľov, ale má uplatňovať skúsenosti iných tvorivov (prihliadajúc na vek žiakov, na osobitosti triedy a školy), pričom má neprestajne prejavovať vlastnú tvorivosť

a vynachádzavosť“ [9]. Charakterové rysy tvorivej osobnosti (tvorivého učiteľa) sú znázornené na obr. 1.



Obrázok 1: Charakteristické črty tvorivej osobnosti [1]

Tvorivosť má svoje významné miesto vo vzdelávaní. Podľa revidovanej Bloomovej taxonómie vzdelávacích cieľov [10] je tvorivosť najvyššou úrovňou kognitívnych schopností. Zohráva dôležitú úlohu v rozvoji osobnosti žiaka: je prostriedkom na dosiahnutie individuálnej sebarealizácie, či schopnosti prispievať k prosperite spoločnosti.

Pedagogická tvorivosť predstavuje tvorivosť u učiteľov a ostatných pedagogických pracovníkov. Takáto tvorivosť pozitívne vplýva aj na tvorivosť žiakov [5]. Ak učiteľ dosahuje vysokú mieru tvorivého myslenia, veľmi pravdepodobne aj jeho žiaci budú dosahovať v tejto oblasti vysokú mieru. V opačnom prípade, kedy má učiteľ nízku mieru tvorivého myslenia, sa táto skutočnosť môže prejaviť aj u žiakov, ktorí nemajú vytvorené podmienky na to, aby sa mohli tvorivo rozvíjať. Ide o vlastnosť, kedy učiteľ nevedome inklinuje k takým psychologickým typom žiakov, k akým sám patrí. Napríklad, ak je učiteľ precízny a dosahuje nízku mieru tvorivého myslenia, vo svojom predmete vysoko hodnotí formálnu stránku práce žiakov (úhladnosť poznámok v zošite, či vzhľad alebo farebnosť projektov) a menej ocení, respektíve neocení originálne, ale nedokonalé riešenia. Psychologicky iné typy žiakov môžu vedieť produkovať originálne riešenia, aj sa o to budú pokúšať, ale nemajú vytrvalosť ich kvalite spracovať, a preto precíznym, málo tvorivým učiteľom nebudú dobre hodnotení. Vytrvalosť a precíznosť sú kladné vlastnosti, ktoré učiteľ svojím prístupom podporuje, avšak nedocenenie tvorivosti zo strany učiteľa vplýva na tvorivosť žiaka negatívne.

Učiteľské zamestnanie patrí medzi tvorivú činnosť. Je to z toho dôvodu, že vyučovacia hodina nikdy nebude taká istá. Budú v nej žiaci, ktorí sa inde neopakujú, podmienky vyučovania sa neustále menia napr. vplyvom nálady žiakov, predchádzajúcich vyučovacích hodín, vplyvom osobnosti učiteľa a jeho aktuálneho psychického a fyzického rozpoloženia a mnohých iných faktorov. Teda neexistuje žiadny algoritmus, ktorý dokáže spoľahlivo určiť priebeh vyučovacieho procesu [1]. Tvorivého učiteľa môžeme

spoznať podľa niekoľkých charakteristík, napr. naučí priamo na vyučovaní, povinné úlohy si plní skôr v porovnaní s ostatnými učiteľmi, či nepreťažuje žiakov domácou prípravou. Tak ako platí, že pri rozvíjaní tvorivého myslenia u žiakov je žiaduce, aby aj učiteľ bol tvorivý a dokázal zrealizovať podmienky vhodné pre tvorivú činnosť, aj pri rozvíjaní tvorivého myslenia u učiteľov by malo vedenie školy a jej riadiaci pracovníci zabezpečiť tvorivú klímu v škole [5]. To sa dá dosiahnuť prijímaním nových myšlienok, sústredením sa na zlepšovanie, oceňovaním tvorenia, hľadania či experimentovania, podporou sebavedomia učiteľov, originalitou, iniciatívou, aktivitou, srdečnosťou, dôrazom na spoluprácu sa celého kolektívu, nielen vedenia školy, nevyhnutná je pozitívna orientácia na prácu, pochyta a optimizmus. Keď sú tieto podmienky splnené, je na učiteľovi, aby pre žiakov vytvoril vhodné podmienky pre tvorivé myslenie.

4 PROJEKTOVÉ VYUČOVANIE V PRÍPRAVE BUDÚCICH UČITEĽOV INFORMATIKY

V študijnom programe budúcich učiteľov informatiky na Univerzite Konštantína Filozofa v Nitre je zaradený seminár Projekty v školskej informatike. Ide o povinne voliteľný predmet, ktorý zvyčajne absolvujú druháci v magisterskom ročníku v zimnom semestri, teda v poslednom roku ich prípravy. Takéto zakomponovanie predmetu má výhodu, že budúci učelia už absolvovali všeobecnú didaktiku aj obe didaktiky informatiky. Osvojili si pojmy zo všeobecnej didaktiky, vedia ich aplikovať v odborovej didaktike, získali praktické zručnosti s projektovaním vyučovania informatiky aj s jeho realizáciou na pedagogickej praxi. Predmet Projekty v školskej informatike je zameraný v zmysle dedukcie, teda od všeobecných poznatkov (o didaktike) cez predmetovú didaktiku (informatiky) až ku konkrétnej metóde vyučovacieho procesu – projektovému vyučovaniu. V tomto predmete sa študent podľa informačného listu naučí:

- uviesť príklady projektov s informatickým obsahom a uplatnením medzipredmetových vzťahov,
- pracovať na projektoch s použitím moderných digitálnych technológií a tvorivých softvérových prostredí,
- zhodnotiť a sformulovať prínos konkrétneho projektu pre dosahovanie cieľov vyučovania,
- aplikovať teoretické vedomosti o projektovom vyučovaní a praktické skúsenosti pri návrhu projektu pre žiakov základnej alebo strednej školy,
- sformulovať ciele, definovať opis výsledného produktu, navrhovať harmonogram, formulovať kritériá hodnotenia,
- hodnotiť kvalitu návrhu školského projektu z hľadísk: ciele, originalita, metodika, formálne spracovanie, vzorový produkt.

V zimnom semestri 2020/2021 sa na predmet Projekty v školskej informatike prihlásili štyria budúci učelia z piatich, pretože piaty študent tento predmet už absolvoval pred rokom. Keďže ide o povinne voliteľný predmet, už z pohľadu prihlasovania sa na predmet môžeme predpokladať, že budúci učelia považujú projektové vyučovanie za dôležité. Keďže s touto skupinkou sme sa zameriavalí na rozvoj kritického myslenia v uplynulom roku počas didaktiky informatiky a tvorivosť je jedna z dispozícií kriticky zmyšľajúceho človeka, tak sme sa rozhodli zamerať sa na

tvorivé myslenie (obr. 2). Podľa Dohňanskej [11] znakmi tvorivého myslenia sú:

fluencia – schopnosť pohotovo reagovať (dôraz na kvantitu)	o Vymenuj čo najviac objektov, ktoré by si dokázal vymodelovať v Tinkercad.
Flexibilita – zmena stratégie riešenia, kde ide o rôznorodosť	o Navrhni čo najrozmanitejšie spôsoby, ako v Scratchi....
Redefinovanie – náhradné riešenie, netradičného spájania vedomostí	o Navrhni, ako zmerať výšku schodu, ak máš k dispozícii ultrazvuk.
Originalita – jedinečnosť nápadu, zvláštnosť, prevkapanie	o Vymysli originálny informatický vtip.
Citlivosť na problémy – postrehnúť podstatu javov	o Nájdi chybu v programe.
Elaborácia – skompletizovanie celku	o Dopln, čo chýba v grafe.

Obrázok 2: Znakmi tvorivého myslenia [11]

V nasledujúcich prípadových štúdiách uvádzame príklady troch aktivít zo seminára Projekty v školskej informatike, v ktorých študenti pracovali na projektových zadaniach. V prípadoch si všímame:

- Aké sociálne vzťahy a interakcie vznikajú počas projektového vyučovania?
- Aká je úroveň tvorivosti úloh?
- Aké znaky tvorivého myslenia sa prejavujú pri práci na projekte?

4.1 Prípád A: Programovanie vedeckého humoru

Projektová aktivita na seminári bola zapojením sa do verejne publikovaného projektu Programujeme vedecký humor organizovaného na Univerzite Pavla Jozefa Šafárika v Košiciach [12] v rámci iniciatívy CodeWeek 2020.

Ciele z hľadiska seminára PSKI pre študentov učiteľstva: získať skúsenosti s prácou na projekte z pozície žiaka, rozvíjať tvorivosť, kritické myslenie a zmysel pre humor.

Zadanie projektu: Vytvorte projekt v prostredí Scratch s odborným informatickým námetom a prvkami humoru.

Opis výsledného produktu: Projekt publikovaný v štúdiu Programujeme vedecký humor v cloude prostredia Scratch s vlastnosťami:

- odborný informatický námet,
- pozitívny, slušný, korektný humor,
- estetické spracovanie formou multimedialnej prezentácie,
- publikovanie v cloude s anotáciou projektu v časti Instructions a uvedením autora a odkazmi na použité zdroje a inšpirácie v časti Notes and Credits.

Priebeh a výsledky: Počas online seminára bolo študentom predstavené zadanie projektu a na inšpiráciu ukážky, ktoré už boli v rámci projektu vytvorené. Diskutovalo sa o nevyhnutnosti porozumenia odbornej stránky vtipu pre vnímanie humornej pointy. Práca na vlastnom projekte prebiehala vo dvojiciach

(rozdelenie na chlapcov a dievčatá). Každá dvojica hľadala najskôr námet na infromatický vtíp, inšpirovali sa zbierkami vtípor na internete. Po výbere námetu začali pracovať na realizácii v Scratchi tak, že jeden pracoval v Scratchi a druhý mu pomáhal slovné odbornými, tvorivými, kritickými príspevkami. Dvojice boli rozdelené na chlapcov a dievčatá.

Dievčatá vytvorili minimalistický multimediálny projekt s odborným námetom týkajúcim sa pojmu rekurzia (obr. 3). Projekt obsahuje dve postavy, ktoré vedú dialóg realizovaný bublinami s textom a riadeným časovaním:

A: „Klop klop klop!“

B: „Kto je tam?“

A: „Rekurzia.“

B: „Kto je tam?“

A: „Rekurzia.“

Zadanie nebolo splnené úplne, na stránke projektu v cloude chýbajú informácie o projekte.



Obrázok 3: Projekt študentiek

Projekt chlapcov je takisto minimalistický multimediálny projekt s dvomi postavami, ktoré vedú dialóg riadený časovaním. Pochopenie pointy vtípu je založené na vedomostiach o hexadecimálnom kódovaní farieb čierna a biela s využitím modelu RGB (obr. 4). Dialóg papagája a zebry:

Papagáj: „Ako sa máš?“

Zebra: „Mám sa ... F F F F F F ... O O O O O Oh .. ale fajn“

Papagáj: „Ale vieš... veci nie sú vždy len čierne a biele...“

Chlapci publikovali projekt aj s požadovanými informáciami.



Obrázok 4: Projekt študentov

Analýza výsledkov, reflexie študentov:

Dvojice si vzájomne odprezentovali projekt, pričom dievčatá nerozumeli vtípu chlapcov. Chlapci museli vysvetliť hexadecimálny farebný kód.

Nasledovala reflexia s cieľom identifikovať znaky tvorivosti počas práce na projekte. Obe dvojice sa zhodli, že najčastejšie a najdlhšie bola prítomná fluencia – mali množstvo nápadov, pričom sa vyjadrili, že „celá chvíľka bola veľká fluencia“. Zaujímavá situácia vznikla pri redefinovaní, kedy dievčatá druhú postavu, ktorá mala byť za dverami, zmenšili a následne ukryli v kľúčovej diere. Pri diskusií o originalite sa chlapci najskôr vyjadrili, že ich práca nemala znaky originality, avšak po pripomenutí toho, že počas tvorby povedali: „To isto nikto nemá!“ objavili prvky originality vo svojom projekte: vytvorili originálne meno pre svoj tím, pre lepšie porozumenie projektu zvolili papagája pre jeho farebnosť a zebra pre jej čierno-bielu farbu. Študenti ocenili reflexiu o tvorivých znakoch ako príležitosť na hlbšie zamyslenie sa nad prežitou skúsenosťou a objavenie zmyslu, ktorý bez reflexie nevideli. Ocenili tiež prítomnosť súťaženia (chlapci vs dievčatá), vytváranie „vecí“ (artefaktu), kolaboráciu, pretože „keby som to mala robiť sama, nebola by som taká tvorivá“.

4.2 Prípady B: Online fotoalbum

Ciele z hľadiska seminára PSKI pre študentov učiteľstva: získať skúsenosti s prácou na spontánnom žiackom projekte z pozície žiaka, hodnotiť priebeh a výsledky z pozície žiaka aj učiteľa, identifikovať metodické postupy, spracovať námet na projektové vyučovanie, diskutovať o návrhu projektového vyučovania.

Zadanie projektu: Nájdite tému projektu, na ktorom budete kolaboratívne pracovať. Následne jednotlivo vypracujte návrh zadania takéhoto projektu, kde bude zahrnutá anotácia, opis výsledného produktu, ciele vyučovania, časový harmonogram a kritériá hodnotenia. Na základe učiteľovho zadania vypracujte na ďalšom seminári projekt kolaboratívnou formou práce a riešenie odprezentujte.

Priebeh a výsledky: Na vymyslenie témy projektu bola zvolená brainstormingová metóda 421. Číslo 4 predstavuje počet účastníkov skupiny, kde každý si v Jamboarde napísal námet na projekt počas 2 minút, následne jeho nápad „koloval“ k ďalšiemu spolužiakovi, ktorý ho počas ďalších 2 minút doplnil a takto sa počas jedného kola vystriedali v každom nápade každý študent. Následne sa vyberal najlepší nápad prostredníctvom hodnotenia jednotlivých nápadov (modré lístky) a vybral sa námet s najvyšším súčtom – fotoalbum našich piatich rokov na univerzite. Na domácu úlohu si študenti vytvorili zadanie na tento projekt, čo slúžilo ako podkladový materiál vyučujúcemu na zadanie projektu. Uvádzame časti zadania:

Anotácia: Online fotoalbum je alternatívou ku klasickému fotoalbumu. Môže byť vytvorený v online prostredí. Výhodou fotoalbumu je, že ho môžeme mať uložený na cloude a v prípade potreby ho nemusíme mať fyzicky so sebou, stačí sa len prihlásiť do cloudu. Takisto je nesporná ďalšia výhoda oproti klasickému fotoalbumu, a to implementácia multimediálnych prvkov, akými sú zvuk, video či animácia. Žiaci sa naučia vytvoriť online fotoalbum, ktorý bude prístupný v cloude pod heslom.

Opis výsledného produktu: Projekt publikovaný v cloude prostredia online MS PowerPoint, ktorý obsahuje:

- snímky: minimálne 15 snímok, použitý morphing slov/znakov/objektov pri všetkých snímkach,

- zdieľanie cez odkaz pod heslom,
- grafiku: min. 10 fotografií, min. 1 animácia, min. 1 video, min. 1 ikona,
- zvuk: hudba,
- text: bez pravopisných chýb, zobrazenie na min. 7 snímkach, min. 1 prvok SmartArt
- zabezpečenie: heslo,
- voliteľné: použitie šablóny.

Pri hodnotení bola použitá metóda 3S [7]:

- splnenie zadania – publikovaný projekt v cloude s prvkami podľa opisu výsledného produktu.
- správnosť výsledkov a procesov – Bol vytvorený online fotoalbum správne? Je grafika spracovaná kvalitne pomocou vhodných nástrojov? Sú zvuky/hudba implementované vhodné? Reflexívne hodnotenie vlastnej práce.
- súhrnnosť – tvorivý námet, prepracovaný scenár, technická presnosť spracovania, množstvo prvkov.

Pri kolaboratívnej práci vytvárania online fotoalbumu sa študenti rozhodli pre nástroj MS PowerPoint, kde využili preddefinovanú šablónu, rozdelili si prácu na snímky, na ktorých pracovali samostatne. Online fotoalbum dokončovali aj na domácu úlohu.

Analýza výsledkov, reflexie študentov: Prezentácia s hodnotením trvala viac ako hodinu. Podrobná špecifikácia výsledného produktu umožňovala objektívnu kontrolu splnenia. Pri jednotlivých častiach zadania sa veľa diskutovalo. Niektoré prvky študenti z nepozornosti zabudli do riešenia zakomponovať, čo malo za následok miestami skleslú náladu. V diskusiách sa zistilo, že niektoré požiadavky zo zadania možno chápať viacerými spôsobmi (napr. čo je animácia), diskutovalo sa, či sa rôzne alternatívne chápania majú považovať za chybu. Diskusia smerovala k zisteniu, že uvedomenie si nedokonalosti je zdrojom toho, že sa môžeme niečo naučiť. Z písomnej reflexie vyplynulo, že najväčším problémom bola organizácia a rozdelenie úloh. Ako prínos v oblasti vedomostí označili študenti online využitie PowerPointu, zdieľanie dokumentu pod heslom či morfing. Do budúcnosti sa naučili, že treba pozorne čítať zadanie a z tvorivých prvkov využívali najmä fluenciu, originalitu a citlivosť na problémy.

4.3 Prípad C: Projekt projektového vyučovania

Ciele z hľadiska seminára PSKI pre študentov učiteľstva: získať skúsenosti s prácou na projekte z pozície učiteľa: spracovať námet na projektové vyučovanie, diskutovať o návrhu projektového vyučovania, hodnotiť priebeh a výsledky z pozície žiaka aj učiteľa, identifikovať metodické postupy, navrhovať alternatívne metodické riešenia.

Zadanie projektu: Vytvoriť zadanie (anotácia, opis, ciele, harmonogram, hodnotenie) na projektové vyučovanie, kde zadávateľ je v role učiteľa. Následne si študent vymení zadanie so spolužiakom a vypracuje spolužiakovu zadanie v role žiaka. Pri prezentácii projektov hodnotí zadávateľ (rola učiteľa) splnenie z pohľadu zadania projektu a žiak prezentuje svoje vypracovanie projektu z pozície žiaka. Pri prezentácii je každý študent v role učiteľa a v role žiaka.

Priebeh a výsledky: Študenti pripravili zaujímavé zadania projektov, pričom dodržali štruktúru a snažili sa o čo najpresnejšiu formuláciu opisu výsledného produktu. Námety projektov:

- Vytvoriť koláž z fotografií, ktoré zobrazujú písmená abecedy z rôznych minimálne 15 predmetov.
- Vytvoriť vlastný videopriebeh (rozprávka, povesť, text piesne).
- Navrhnuť najvhodnejšie počítačové vybavenie pre dané účely (napr. pre prácu youtubera).
- Vytvoriť propagačný materiál/prospekt o obci (rodnej obci, bydlisku).

Po vypracovaní zadání z pozície učiteľa si študenti zadania vymenili a pracovali na nich do nasledujúcej hodiny v pozícii žiaka. Vypracované zadania prezentovali.

Analýza výsledkov, reflexie študentov: Zaujímavé situácie vznikali pri prezentácii, kde si študenti skutočne dávali pozor na jednotlivé prvky, ktoré mali splniť a „zabúdanie“ sa už neobjavilo. Vyskytlo sa však to, že „učiteľ“ povedal „žiakovi“, že nesplnil celkom požiadavky, ale diskusiou sa dospelo k záveru, že nie „žiak“ urobil chybu, ale „učiteľ“ nedostatočne naformuloval požiadavky. Takisto sa zakomponovala nová úloha, a to oznámkovať projekt (elaborácia). Mnohí „učitelia“ sa tejto otázke vyhýbali dovedy, kým nebola explicitne daná, pričom v tomto momente mali tendenciu namiesto známky začať kritizovať seba ako učiteľa, čo mohli oni lepšie urobiť. Zároveň sa dostali k tomu, že kritériá mohli viac rozpracovať, pretože v takom prípade je hodnotenie objektívnejšie a žiak ho môže lepšie prijať. Zaujímavým spracovaním bola abecedná koláž (originalita), ktorá veľmi milo prekvapila „učiteľku“, ktorá následne povedala, že nečakala takéto spracovanie s takouto myšlienkou (obr. 5).



Obrázok 5: Výsledný projekt Abecedná koláž

5 DISKUSIA

Na prvom seminári študenti vypracovali Kelloggov test tvorivosti. Výsledky testu boli nadpriemerné u všetkých participujúcich študentov (nadpriemerne tvorivá až veľmi tvorivá osobnosť). Naším primárnym cieľom preto nebolo rozvíjanie tvorivosti, ale metakognícia – rozmýšľanie o vlastnej tvorivosti a jej úlohe vo vzdelávacom procese. Študenti mali všeobecné teoretické poznatky o tvorivosti, pretože sa to učili na didaktike. Praktické identifikovanie tvorivých znakov v konkrétnych činnostiach však bolo pre nich nové a kategorizácia im lepšie pomohla zorientovať

sa v tom, čo je tvorivé a čo nie. Vedomé rozvíjanie a uplatňovanie prvkov tvorivosti u budúcich učiteľov má byť predpokladom ich budúceho podporovania tvorivosti u žiakov.

V prípadových štúdiách sme sa zamerali na sledovanie sociálnych interakcií a vzťahov a tvorivosti v projektovom vyučovaní.

Sociálne vzťahy a interakcie. V prezentovaných prípadoch sa uplatňovali tri rôzne sociálne formy vyučovania. V prípade A išlo o prácu vo dvojiciach. Študenti pracovali v úlohe žiakov na zadaní učiteľa. V reflexii referovali, že spolupráca vo dvojici (zdieľanie nápadov a rozhovory o realizácii projektu) významne podporovala ich tvorivosť. Pozitívny vplyv mal aj motív súťaženia dvojíc chlapcov a dievčat. V prípade B pracovali študenti opäť v úlohe žiakov vo väčšej (4-člennej) skupine. Využili učiteľom ponúknutú techniku brainstormingu na výber témy. Výsledkom bol výber témy vyvolávajúcej pozitívne emócie zo spoločného študentského života. Projekt ďalej realizovali formou del'by práce a individuálnym riešením čiastkových úloh. Tu sa prejavili nedostatky v manažovaní projektu, chýbalo vzájomné kritické hodnotenie projektu ako celku, ktoré mohlo odhaliť nepozornosť jednotlivcov pri vypracovaní svojich častí projektu. V prípade C pracovali študenti predovšetkým v úlohe učiteľa, ktorý vypracúva zadanie projektu a hodnotí výsledok. Aby bolo čo hodnotiť, v úlohe žiakov si študenti navzájom vypracovali svoje zadaná. Pri hodnotení projektov študenti lepšie zvládali kritické hodnotenie vlastnej práce (učiteľského zadania) ako práce spolužiakov (žiackeho riešenia).

Úroveň tvorivosti úloh. Na hodnotenie úrovne tvorivosti úloh použijeme taxonómiu tvorivých úloh podľa Gehlbacha a Zelinu, ktorá je založená na konvergentnosti a divergentnosti úloh. Opirajú sa o tri premenné [5]:

- definovanie problému,
- riešenie problému,
- výstup/výsledok.

Každá premenná má dve možnosti – otvorená (O - divergentná) alebo zatvorená (Z - konvergentná). Na základe zhodnotenia všetkých premenných sa stanovuje miera tvorivej úlohy. V prípade A sú všetky tri premenné otvorené (O). V porovnaní s ostatnými dvomi prípadmi môžeme definovanie problému považovať za najmenej otvorené – v zadaní je daná téma (vedecký humor) a forma (program v Scratchi). Ohraničenie je však natoľko všeobecné, že v rámci daného ohraničenia je otvorené. V prípade B mali študenti sami vymyslieť námet na projekt, jeho zadanie však špecifikoval učiteľ, čím sa možnosti riešenia výrazne obmedzili. Spomedzi troch prípadov je prípad B najviac uzavretý v premennej riešenie problému. Prípad C opisuje projektovú úlohu otvorenú vo všetkých troch premenných.

Znaky tvorivosti. V prípadoch A a B, v ktorých študenti pracovali kolaboratívne vo dvojiciach alebo v skupine zo znakov tvorivosti dominovala fluencia (schopnosť generovať veľa nápadov). Zdieľanie nápadov podporuje tvorbu nových nápadov. V prípadoch B a C, kde pracovali študenti aj samostatne, sme ako dominantné identifikovali citlivosť na problémy a elaboráciu.

6 ZÁVER

V troch prípadových štúdiách sme predstavili tri rôzne projekty realizované s budúcimi učiteľmi informatiky na vysokoškolskom seminári Projekty v školskej informatike. Projekty sa lišili z hľadiska sociálnej formy v tom, v akej úlohe v nich študenti vystupovali (v úlohe žiaka alebo učiteľa) a v akej miere sa

využívala spolupráca (kolaboratívny alebo individuálny projekt). Z kvalitatívnej analýzy prípadov sa ukazuje, že sociálne vzťahy a interakcie v projektovom vyučovaní ovplyvňujú to, aké znaky tvorivosti dominujú pri práci na projekte. Všetky prípady projektového vyučovania však vykazovali vysokú úroveň tvorivosti úloh bez ohľadu na to, či ide o kolaboratívny alebo individuálny projekt. Na základe prezentovaných prípadov môžeme konštatovať, že skúsenosti z tvorby projektov z pozície žiaka aj učiteľa sú pre prípravu budúcich učiteľov informatiky veľmi prínosné. Reflektovanie týchto skúseností prispieva k lepšiemu porozumeniu tvorivosti a metód na jej cieľavedomé uplatňovanie a rozvíjanie.

POĎAKOVANIE

Tento článok bol finančne podporený Univerzitnou grantovou agentúrou a projektom Implementácia nových trendov v informatike do výučby algoritmického myslenia a programovania v predmete informatika v sekundárnom vzdelávaní (018UMB-4/2020).

BIBLIOGRAFICKÉ ODKAZY

- [1] ZELINA, M. *Stratégie a metódy rozvoja osobnosti dieťaťa*. Iris, 2011. 978-80-892-5660-0.
- [2] DEWEY, J. *Experience and Education*. New York : Touchstone, 1938, 96 p. ISBN-13: 978-1-4165-8727-9.
- [3] ČAPEK, R. *Moderní didaktika*. Praha: Grada, 2017, 608 s. ISBN 978-80-247-3450-7.
- [4] BLUMENFELD, P. C. et al. *Motivating Project-Based Learning : Sustaining the Doing, Supporting the Learning*. In *Educational Psychologist*, 1991, vol. 26, no. 3&4, pp. 369–398.
- [5] TUREK, I. *Didaktika*. Bratislava: Wolters Kluwer. 2014. ISBN 978-80-8168-004-5.
- [6] KOSOŤÁ, B. *Projektové vyučovanie*. In *Pedagogické rozhľady*. 1995/1996, roč. 4, č. 3.
- [7] KALAŠ, I. et al. *Ďalšie vzdelávanie učiteľov základných škôl a stredných škôl v predmete informatika: Digitálne technológie a zásahy do vyučovania*. Bratislava : ŠPÚ. 40 s. ISBN 978-80-89225-62-0.
- [8] LOVÁSZOVÁ, G. et al. *Stratégie kritického a tvorivého myslenia v príprave učiteľov informatiky*. Nitra : UKF. 2019. ISBN 978-80-558-1499-5.
- [9] ĎURIC, L. et al. *Psychológia pre učiteľov*. Bratislava : SPN, 1973.
- [10] ARMSTRONG, P. *Bloom's Taxonomy*. Center for Teaching, Vanderbilt, 2019.
- [11] OSBORN, A.: *Applied imagination*. N.Y. : Ch. Scribner's sons, 1963.
- [12] ŠNAJDER, E. *#CodeWeek 2020 –Programujeme vedecký humor* [online]. cit. 2021-01-31. 2020. Dostupné na internete: <https://humor.ics.upjs.sk/>

Mobilná aplikácia s umelou inteligenciou pre deti

Mobile application with artificial intelligence for children

Jaroslav Knežník
Univerzita Mateja Bela
Tajovského 40
97401 Banská Bystrica
Slovenská Republika
jkneznik@student.umb.sk

Jarmila Škrinárová
Univerzita Mateja Bela
Tajovského 40
97401 Banská Bystrica
Slovenská Republika
jarmila.skrinarova@umb.sk

Patrik Voštinár
Univerzita Mateja Bela
Tajovského 40
97401 Banská Bystrica
Slovenská Republika
patrik.vostinar@umb.sk

ABSTRACT

The goal of the paper is to present a methodology for creating artificial intelligence models for image recognition. These models can be used in programming environments suitable for teaching programming to children. We describe the "Search." methodology, by which we create a model of artificial intelligence for image recognition. Subsequently, we focus on the description of the methodology of creating a mobile application in the App Inventor environment, in which the model of artificial intelligence is used.

Keywords

Programming for children. Teaching programming. Artificial intelligence. Artificial intelligence for kids. Programming with artificial intelligence. AppInventor.

ABSTRAKT

Cieľom tohto článku je prezentovať metodiku tvorby modelov umelej inteligencie na rozpoznávanie obrázkov. Tieto modely je potom možné používať v programovacích prostrediach, vhodných na výučbu programovania pre deti. V príspevku opisujeme metodiku s názvom „Hľadaj.“, pomocou ktorej vytvoríme model umelej inteligencie na rozpoznávanie obrázkov. Následne sa zameriavame na opis metodiky tvorby aplikácie v prostredí App Inventor, v ktorom je použitý model umelej inteligencie.

Kľúčové slová

Programovanie pre deti. Vyučovanie programovania. Umelá inteligencia. Umelá inteligencia pre deti. Programovanie s umelou inteligenciou. AppInventor.

1 ÚVOD

V súčasnosti veľmi často počujeme pojem: „umelá inteligencia“. Aplikácie a technológie, na báze umelej inteligencie, používame denne, bez toho, aby sme si to uvedomovali. Typickým príkladom je rozpoznávanie odtlačku prsta alebo scan tváre, ktorý používame pri prístupe do nášho telefónu. Ďalším príkladom je nasadenie „softvérových robotov“ na vyhľadávanie obsahu webových stránok, ktoré implementovala napríklad spoločnosť Google. Vďaka tejto technológii máme vo vyhľadávači zobrazené odkazy na webové stránky za pár sekúnd.

Čo je inteligencia? Definícií pojmu inteligencia je veľa. Autori Shane Legg a Marcus Hutter v práci „A Collection of Definitions of Intelligence“, v roku 2007, uviedli 71 definícií inteligencie. Pričom 18 definícií predstavuje definície zo slovníkov a encyklopédií, 35 definícií od vedcov z oblasti psychológie a 18 od vedcov z oblasti umelej inteligencie. Jedna z nich definuje ľudskú inteligenciu ako: „Schopnosť využívať pamäť, vedomosti, skúsenosti, porozumenie, myslenie, predstavivosť a úsudok s cieľom vyriešiť problémy a prispôbiť sa novým situáciám“ [1]. Z definície je zrejmé, že ľudská inteligencia predstavuje

kompozíciu schopností, ktoré sa postupne rozvíjajú a navzájom sa ovplyvňujú. Tiež je zrejmé, že ani súčasný moderný svet so všetkými poznatkami a technológiami nie je pripravený vyvinúť takú komplexnú umelú inteligenciu ako je ľudská.

Umelá inteligencia sa zaoberá vývojom algoritmov a technológií, ktoré modelujú určité prvky inteligentného správania sa. Tieto prvky správania sa sú odpozorované od biologických systémov a sú podstatne zjednodušené.

Podľa [3] je umelá inteligencia: „...schopnosť systému primerane konať v neurčitom prostredí, ... je to činnosť, ktorá zvyšuje pravdepodobnosť úspechu na dosiahnutie čiastkových cieľov správania sa tak, aby systém splnil konečný cieľ.“ [2,3]

Medzi základné úlohy umelej inteligencie patrí:

- Rozpoznávanie obrazu, zvuku, textu.
- Klasifikácia, t.j. zatriedenie prvkov alebo objektov do určitých skupín.
- Predikcia, ktorá na základe predchádzajúcich znalostí predpovedá výstupy v ďalších krokoch.

Cieľom tohto článku je uviesť príklady, metodiku a primerané aktivity, ktoré môžu učitelia informatiky použiť pri vyučovaní programovania. Metodika, na základe praktického príkladu, rozvíja praktické zručnosti v programovaní a súčasne rozvíja porozumenie práce s umelou inteligenciou, zameranou na rozpoznávanie obrazu. Cieľom je, aby po použití metodiky, žiak chápal čo je potrebné urobiť, aby umelá inteligencia rozpoznala obraz a dokázal si toto poznanie zovšeobecniť na celú triedu systémov, určených na rozpoznávanie pomocou umelej inteligencie.

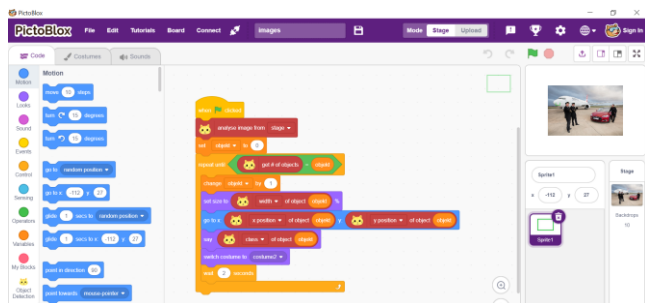
V kapitole 2 uvádzame niekoľko príkladov dostupných softvérových nástrojov a metodík, ktoré vytvárajú modely umelej inteligencie. Tieto modely je potom možné používať v programovacích prostrediach, vhodných na vyučovanie programovania pre deti. V kapitole 3 opisujeme metodiku tvorby príkladov s umelou inteligenciou. V tejto kapitole sme predstavili metodiku „Hľadaj“, pomocou ktorej vytvoríme model umelej inteligencie na rozpoznávanie obrázkov. V kapitole 4 sa zameriavame na opis metodiky tvorby programu v online prostredí App Inventor, v ktorom je použitý model umelej inteligencie vytvorený v kapitole 3.

2 UMEĽÁ INTELIGENCIA PRE DETI

V súčasnosti (najmä v zahraničí) patrí pojem umelá inteligencia na školách k čoraz viac skloňovaným témam. Je dôležité pracovať s umelou inteligenciou aj na slovenských školách. Široký pojem „umelá inteligencia“ je možné žiakom predkladať radšej formou čiastkových úloh a na príkladoch ich naučiť pracovať s umelou inteligenciou. Dobrým príkladom je rozpoznávanie obrazu. V súčasnosti existuje viacero aplikácií, ktoré využívajú princípy

umelej inteligencie a ich vývojové prostredia sú podobné tým, ktoré sa už dnes využívajú na základných a stredných školách.

Jednou z takýchto aplikácií je PictoBlox¹. PictoBlox obsahuje blokové prostredie, ktoré je veľmi podobné Scratchu a umožňuje nám rozoznávať objekty, gestá, tváre, hudobné tóny a mnoho iného. Na obrázku 1 je zobrazené programovacie prostredie aplikácie PictoBlox.



Obrázok 1: Ukážka blokového prostredia Pictobloxu

Na obrázku 2 môžeme vidieť výsledok rozpoznania obrazu. Vľavo, na obrázku 2, aplikácia rozpoznala, že dievčatko nemá rúško a postavička hovorí: „prosím, nos rúško, aby si sa ochránila“. Vpravo, na obrázku 2, aplikácia rozpoznala, že dievčatko má rúško a postavička hovorí: „ďakujem, že nosíš rúško“.



Obrázok 2: Výstup z programu na rozpoznanie obrázku v Pictoblox [4]

MIT App Inventor² ponúka sériu 6 metodík vhodných pre predmet informatika na nižšom a vyššom stupni sekundárneho vzdelávania. Sú zamerané na spoznávanie základných konceptov a postupov typických pre aplikácie umelej inteligencie. Dôraz sa kladie na experimentovanie, objavovanie a tvorivý prístup pri programovaní vlastných mobilných aplikácií. Okrem podrobnej osnovy vyučovacej hodiny sú k dispozícii aj rôzne podporné materiály (prezentácie pre učiteľov, videá, sprievodcovia tvorbou mobilných aplikácií, vzorové riešenia) [5].

Ďalšie zaujímavé metodiky a príklady pre využitie umelej inteligencie na školách ponúkajú nástroje na webstránke Machine learning for kids³. Nástroj umožňuje, na základe našich vstupných dát, natrénovať model. Takto natrénovaný model použijeme v našom programe. Použitý model rozhodne, či sa objekt alebo tvar, ktorý bol na vstupných dátach zhoduje s tým, ktorý chceme rozlišovať. Ak sú vstupné dáta obrázky, potom model rozpozná, či sa na obrázku nachádza naučený objekt. Metodiky a nástroje, ktoré urobia učiteľom a študentom pri vyučovaní radosť vytvoril pán Dale Lane [6].

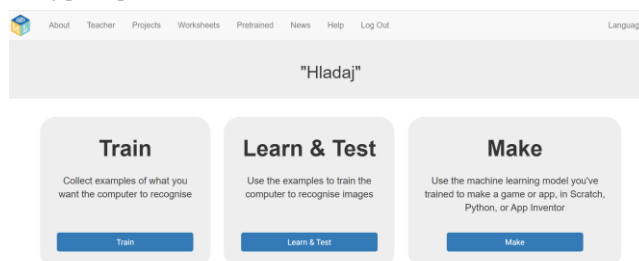
3 AKTIVITA HĽADAJ V „MACHINE LEARNING FOR KIDS“

V tomto článku opíšeme príklad a aktivitu „Hľadaj“, ktorú sme vytvorili pomocou nástroja na stránke Machine learning for kids. Najprv vytvoríme nový projekt, ten v sebe bude zahŕňať naše vstupné dáta a taktiež aj model, ktorý je potrebné na základe týchto dát natrénovať. Pri vytváraní projektu je možné zvoliť si z možností text, obrázky, čísla a zvuky. My sme zvolili „obrázky“. To znamená, že budeme rozpoznať obrázky.

Keď už máme projekt vytvorený zobrazia sa nám 3 kroky v postupe, ktoré je potrebné dodržať (pozri Obrázok 3.):

- Trénuj (angl. Train).
- Uč sa a testuj (angl. Learn & Test).
- Vlož (Make).

Tieto možnosti hovoria o tom, akým spôsobom sa má v projekte ďalej postupovať.



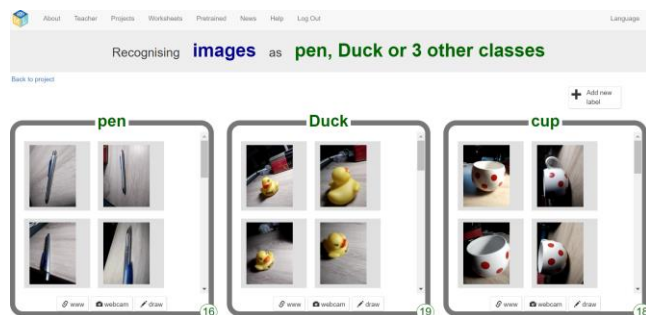
Obrázok 3: Kroky postupu

Časť *Train* je prvá časť, do ktorej je potrebné vstúpiť po vytvorení projektu a zároveň aj najdôležitejšia časť projektu. V tejto časti musíme poskytnúť nášmu projektu dostatočné množstvo vstupných dát, v našom prípade obrázkov, na základe ktorých sa natrénuje model. Dáta sú vložené do kategórií, ktoré sú posudzované osobitne. Model bude po natrénovaní schopný rozlišovať medzi týmito kategóriami. Počet vstupných dát, pre každú kategóriu musí byť väčší ako desať, aby bolo možné model úspešne natrénovať. Je však dobré poskytnúť modelu väčšie množstvo vstupných dát, pretože od množstva vstupných dát sa odvíja aj presnosť, s akou je model schopný rozoznať čo sa na obrázku nachádza. Vhodné je začať tak, že sa zvolené objekty budú od seba výrazne odlišovať. My sme sa rozhodli, že budeme rozlišovať objekty pero, kačička a šálka (pozri Obrázok 4). Fotografia získame kliknutím na ikonu „webcam“ v spodnej časti každej kategórie. My sme každý objekt odfoťili viackrát, konkrétne kačičku 19-krát, šálku 18-krát a pero 16-krát. Pre objekty, ktoré sú si viac podobné je potrebné vytvoriť a vložiť na tréovanie viac vstupných súborov (fotografií).

¹ <https://thestempedia.com/product/pictoblox/>

² <https://appinventor.mit.edu/>

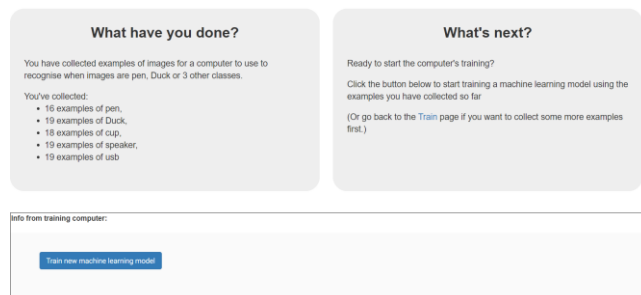
³ <https://machinelearningforkids.co.uk/>



Obrázok 4: Kategórie vstupných dát

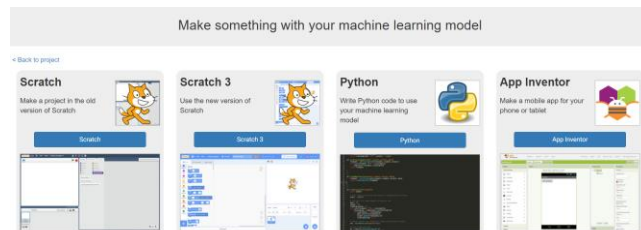
Ak už máme dostatočný počet obrázkov (fotografií) pre šálku, kačičku a pero môžeme sa presunúť do časti Learn & Test. My sme sa však rozhodli nášho projektu pridať ďalšie 2 objekty. (USB kľúč a reproduktor). A až potom sme sa presunuli do časti Learn & Test.

V časti Learn & Test sa model naučí rozpoznávať obrázky. Systém vypíše čo z celého procesu sme už urobili a čo je potrebné spraviť ďalej (pozri Obrázok 5.). Klikneme na tlačidlo Train new machine learning model a systém automaticky natrénuje model. Trvá to spravidla pár minút.



Obrázok 5: Natrénovanie modelu

Poslednou časťou postupu je vloženie modelu do aplikácie (pozri Obrázok 6.), tu je potrebné implementovať vytvorený model do jedného z dostupných programovacích prostredí. Na výber je z možností Scratch, Python a App Inventor. V našom prípade sme vybrali App Inventor. Podľa inštrukcií pridáme vytvorený model ako rozšírenie do vytvoreného programu v App Inventore. Na tento krok sa bližšie pozrieme v kapitole 4.



Obrázok 6 Programovacie jazyky (prostredia), do ktorých je možné vložiť model

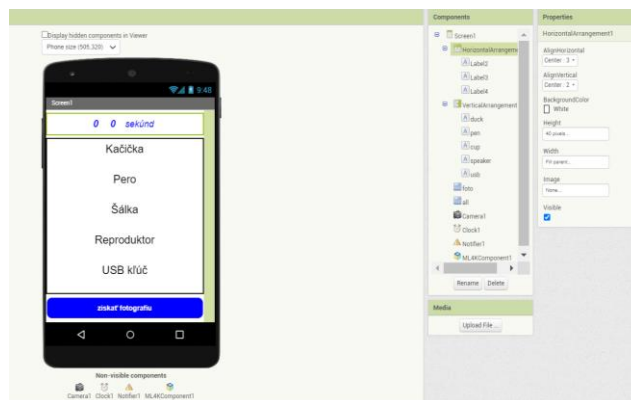
4 TVORBA WEBOVEJ APLIKÁCIE POMOCOU APPINVENTOR – TRETÍ KROK V METODIKE HĽADAJ

V predchádzajúcej časti sme opísali prvé 2 kroky metodiky „Hľadač“. V tejto kapitole opíšeme postup tvorby mobilnej aplikácie. Aplikácia bude využívať model vytvorený pomocou predchádzajúcej kapitoly. Model bude rozoznávať či sa na

fotografii získanej pomocou našej aplikácie nachádza kačička, pero, šálka, USB kľúč alebo reproduktor. Fotografie získame pomocou fotoaparátu priamo cez aplikáciu. V poslednej časti Make po výbere nástroja App Inventor dostaneme vygenerovanú URL adresu. Na tejto adrese sa nachádza naučený model, adresu vložíme do nášho novovytvoreného projektu v App Inventore ako rozšírenie. Toto rozšírenie tu nájdeme pod skratkou „ML4KComponent“. V Prostredí AppInventoru sa náš projekt skladá z grafickej časti (časť označovaná ako dizajnér) a z časti bloky, ktorá obsahuje programovacie prostredie.

4.1 Postup tvorby vzhľadu mobilnej aplikácie

V prostredí dizajnéra vytvárame vzhľad mobilnej aplikácie. Dizajnér obsahuje políčka, ktoré sa budú vyfarbovať, ak sa daný objekt nájde a súčasne sa bude zaznamenávať čas, nameraný pomocou stopiek. Pri vytváraní dizajnéra budeme pridávať komponenty z palety v ľavej časti obrazovky. Najprv vytvoríme hlavnú časť, v tejto časti sa budú nachádzať názvy našich kategórií (pozri Obrázok 7.).



Obrázok 7: Tvorba vzhľadu mobilnej aplikácie – dizajnér

Z časti „Extensions“ z palety komponentov pridáme model „ML4KComponent1“ a komponent „Vertical Arrangement“ (zvislé usporiadanie) z časti „Layout“ (rozloženie) a vložíme ho na obrazovku. Komponent „label“, ten vložíme do komponenty „Vertical Arrangement“.

V časti „Properties“ (vlastnosti), zmeníme text, jeho veľkosť a umiestnenie pomocou komponentu „label“. V časti „Components“ premenujeme „Label1“ na „cup“. Tento krok zopakujeme pre všetkých 5 kategórií obrázkov.

Následne vytvoríme tlačidlo tak, že vyberieme z palety „button“ (tlačidlo), vložíme ho pod „Vertical Arrangement“ a upravíme parametre tlačidla.

Teraz potrebujeme vytvoriť ukazovateľ stopiek, preto z palety pridáme: 1x „Horizontal Arrangement“ z časti „Layout“, 3x komponent „label“ a upravíme parametre každého „label“ komponentu

Nakoniec z palety pridáme aj neviditeľné komponenty „Camera“, „Clock“ a „Notifier“ a dizajnér máme hotový (pozri Obrázok 7).

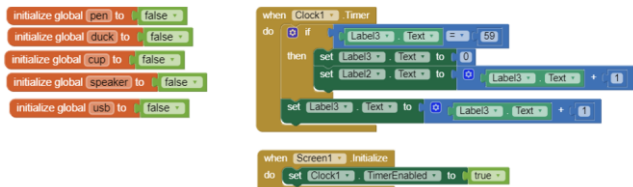
4.2 Postup tvorby programu s modelom z umelej inteligencie

V časti bloky naprogramujeme funkcionality jednotlivých komponentov, ktoré sme vytvorili v časti dizajnér. Cieľom je, aby aplikácia pracovala takto:

- Po spustení aplikácie sa spustia stopky, tie budú zobrazovať čas v hornej časti obrazovky v minútach a sekundách.
- Po stlačení tlačidla „získať fotografiu“ sa spustí fotoaparát,
- po získaní fotografie sa fotografia odošle na rozpoznávanie vloženému modelu,
- Model určí, ktorá z vecí (kačička, pero,...) sa nachádza na fotografii, a vyfarbí pozadie názvu danej veci zelenou farbou v mobilnej aplikácii.
- Po odfotení všetkých určených vecí sa zobrazí notifikačná hláška spolu s dosiahnutým časom.

Postup tvorby programu:

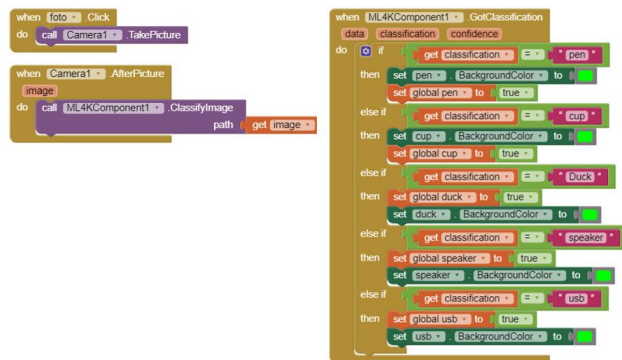
Najprv inicializujeme globálne premenné, ktoré si vytvoríme na hodnoty „false“ pre všetky kategórie obrázkov (pozri Obrázok 8 vľavo).



Obrázok 8: Inicializácia premenných (vľavo), práca stopiek (vpravo)

Pri spustení rozpoznávania sa spustia stopky, čas ktorý stopky zaznamenávajú sa zobrazuje. Postačí nám ukazovateľ minút a sekúnd (pozri Obrázok 8.).

Po stlačení tlačidla „foto“ sa na telefóne spustí fotoaparát, fotografia sa následne odošle do modelu. Z blokov nášho modelu vyberieme „Classify image“, tento blok zavolá funkciu kategórie, ktorá pošle fotografiu do nášho modelu (pozri Obrázok 9.).



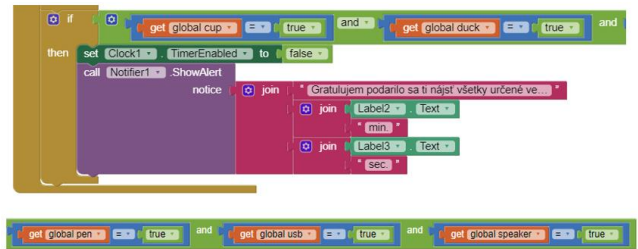
Obrázok 9 Funkcia fotoaparát (vľavo), rozpoznávanie a výstup z modelu UI (vpravo)

Použijeme model umelej inteligencie *ML4KComponent1*, ktorý sme vytvorili v kapitole 3. Model rozpozná, do ktorej kategórie fotografia patrí a vráti nám naspäť názov kategórie.

Pomocou podmienky „IF-ELSE“ zistíme názov kategórie. Pri kategórii, pre ktorú je podmienka splnená, zmeníme hodnotu príslušnej premennej na „true“ (pozri Obrázok 9). Tým sa farba pozadia príslušného názvu kategórie obrázku na mobilnej aplikácii na zelenú (pozri Obrázok 11.).

Na koniec tohto cyklu pridáme podmienku tak, aby sme zistili, či sú všetky premenné nastavené na hodnotu „true“ (či boli správne rozpoznané všetky veci). Ak sme našli všetky určené veci, zastavia

sa stopky a zavolá sa okno „notifier“ s výpisom: „Gratulujem, podarilo sa ti nájsť všetky určené veci“.



Obrázok 10: Podmienka ukončenia rozpoznania všetkých objektov

Na obrázku 11 je zobrazená obrazovka našej vytvorenej aplikácie.



Obrázok 11: Ukážka bežiaciej aplikácie s umelou inteligenciou

5 ZÁVER

Základ metodiky, ktorá je opísaná v tomto článku, pôvodne tvorila jedna z priebežných úloh v predmete Didaktika informatiky, ktorý je vyučovaný na Katedre informatiky, FPV UMB v Banskej Bystrici. Študenti druhého ročníka magisterského štúdia v programe Učiteľstvo informatiky v kombinácii dostali za úlohu vytvoriť programátorské aktivity, ktoré využívajú prvky umelej inteligencie a zároveň sú určené pre deti základných škôl. Študentom bolo predstavených niekoľko nástrojov a príkladov z oblasti umelej inteligencie pre deti. Niekoľko z nich sme uviedli v kapitole 2. Z pohľadu učiteľov môžeme konštatovať, že študenti úspešne splnili úlohu a vytvorili programy a aktivity, pričom využili nástroj machine learning for kids. Študenti, po splnení úloh v rámci spätnej väzby, uviedli:

- „...úloha skvelo prepája prvky umelej inteligencie a softvéru (napr. Scratch a App Inventor), ktorý žiaci dobre poznajú...“
- „Vybral som si App Inventor a vytvoril jednoduchú mobilnú aplikáciu. Táto aplikácia rozpoznáva určité veci zachytené na fotografii, pričom komunikuje s modelom umelej inteligencie a na základe odpovede, ktorú dostane zobrazuje

čo sa nachádza na fotografii. Domnievam sa, že tento nástroj má obrovský potenciál a je jedným z dôležitých krokov, ktoré umožňujú umelej inteligencii dostať sa už k žiakom základných škôl.“

- „Pred touto aktivitou som sa s umelou inteligenciou nestretol, no zaoberal som sa detským programovaním a preto sa domnievam, že pravé aktivity tohto druhu dokážu najmä u žiakov základných škôl vzbudiť veľký záujem. Pre mňa osobne bola práca s nástrojom machine learning for kids veľmi zaujímavá a myslím, že rovnaký efekt by to malo aj pre žiakov, ktorých zaujíma všetko nové a neprebádané. V neposlednom rade to prinesie oživenie programovania a hodín informatiky, nakoľko možnosti využitia tohto nástroja sú obmedzené len fantáziou žiakov.“

K tejto aktivite sme vypracovali metodiku, ktorú študenti pôvodne plánovali overiť na základnej škole, kde absolujú pedagogickú prax alebo na informatickom krúžku, ten navštevujú deti zo základných škôl v Banskej Bystrici. No, žiaľ kvôli nepredvídaným okolnostiam pandémie sa nám nepodarilo navštíviť školu a overiť túto metodiku. Ostáva nám len dúfať, že sa situácia v dohľadnej dobe zlepší a my sa opäť plnohodnotne vrátíme do našich životov.

POĎAKOVANIE

Príspevok je podporený grantovým projektom Implementácia nových trendov v informatike do výučby algoritmického myslenia a programovania v predmete informatika v sekundárnom vzdelávaní, projekt KEGA 018UMB-4/2020.

BIBLIOGRAFICKÉ ODKAZY

- [1] LEGG, S., HUTTER, M. A collection of Definitions of Intelligence. In *Frontiers in Artificial Intelligence and Applications*. 2007, vol.157, p. 17-24.
- [2] LEGG, S., HUTTER, M. Universal Intelligence: A Definition of Machine Intelligence. *Minds and machines*. 2007, vol. 17, no. 4, p. 391-44. ISSN: 0924-6495. DOI= 10.1007/s11023-007-9079-x.
- [3] ALBUS, J., S. Albus. Outline for a theory of intelligence. *IEEE Transaction of Systems, Man and Cybernetics*. 1991, vol. 21, no. 3, p. 473–509.
- [4] PICTOBLOX. *Thestempedia.com* [online]. 2021 Pictoblox. [cit. 2021-01-08]. Dostupné na: <https://thestempedia.com/shop/online-courses/artificial-intelligence-for-kids/>
- [5] MIT App Inventor. Artificial Intelligence with MIT App Inventor. Dostupné na internete: <https://appinventor.mit.edu/explore/ai-with-mit-app-inventor>
- [6] LANE, D. *Bringing AI into classroom* [online] [cit. 2021-01-27]. Dostupné na internete <https://dalelane.co.uk/blog/?m=202002>

Identifikácia gradácie informatických konštruktov pri vyučovaní edukačnej robotiky na 1. stupni ZŠ

Identification construct gradation in robotics educational at primary school

Andrea Hrušecká

Katedra didaktiky matematiky, fyziky a informatiky,
FMFI UK
Mlynská dolina
842 48 Bratislava
Slovensko

andrea.hrusecka@fmph.uniba.sk

Karolína Miková

Katedra didaktiky matematiky, fyziky a informatiky,
FMFI UK
Mlynská dolina
842 48 Bratislava
Slovensko

karolina.mikova@fmph.uniba.sk

ABSTRACT

In this paper we focus on identifying a scale that would represent the cognitive levels of difficulty in teaching educational robotics at primary school. We carried out the research at two different schools and during the closure of schools at home. We worked with a series of activities that we iteratively developed, edited and then analyzed. Our results suggest a four-level scale, with each level containing 4 different areas. By combining these areas, we can create a gradation of activity as required by the cognitive level of educated students. These are preliminary results of qualitative research, which are located at the level of the cognitive domain and will need to be linked to the psychomotor domain in the future.

Keywords

Educational robotics. Difficulty gradation. Cognitive operations. Blue-Bot. Primary education.

ABSTRAKT

V tomto článku sa venujeme identifikovaniu škály, ktorá by reprezentovala kognitívne stupne obťažnosti pri vyučovaní edukačnej robotiky na 1. stupni ZŠ. Výskum sme realizovali na dvoch rôznych školách a počas zatvorenia škôl v domácom prostredí. Pracovali sme so sériou aktivít, ktoré sme iteratívne vyvíjali, upravovali a následne analyzovali. Naše výsledky naznačujú štvorúrovňovú škálu, pričom každá úroveň obsahuje 4 rôzne oblasti. Kombináciou týchto oblastí vieme vytvárať gradovanie aktivity tak, ako požaduje kognitívna úroveň vzdelávaných žiakov. Ide o predbežné výsledky kvalitatívneho výskumu, ktoré sa nachádzajú na úrovni kognitívnej domény a v budúcnosti budú potrebovať prepojenie s psychomotorickou doménou.

Kľúčové slová

Edukačná robotika. Gradácia náročnosti. Kognitívne operácie. Blue-Bot. Prvý stupeň ZŠ.

1 ÚVOD

Edukačná robotika si nachádza už dobré desaťročie svoje miesto v kvalitnom a modernom vzdelávaní. Jej implementácia sa pomaly presúva do čoraz nižších stupňov vzdelávania. Obľúbenými edukačnými stavebnicami sú LEGO Mindstorms EV3 či NXT, Arduino a v poslednej dobe už aj Ozobot či micro:bit. V rámci projektu *Vzdelávanie pedagogických zamestnancov materských škôl ako súčasť celoživotného vzdelávania* [1] boli v rokoch 2007 až 2013 dodané do materských škôl notebooky, multifunkčné

zariadenia, digitálne fotoaparáty, televízory, či elektronické didaktické pomôcky. V rámci nich aj asi 4000 robotických hračiek Bee-Bot, tzv. "včely" [2]. Stretli sme sa s viacerými prípadmi, kedy tieto robotické hračky ostali v skrinách, alebo s nimi boli realizované aktivity zamerané na výučbu takých tém, kde vyučovanie programátorských konštruktov bol len ako vedľajší produkt [3].

Medzičasom vznikla nová verzia "včely" – Blue-Bot, ktorá rozšírila funkcionality Bee-Botov. Viac o nich v kapitole 3. Tieto robotické hračky zaujali nie len u nás na Slovensku, aj v zahraničí, kde vzniklo viacero metodických materiálov uľahčujúcich ich použitie v školskom prostredí [4, 5].

V rámci systematického budovania predmetu informatika na Slovensku sme sa rozhodli neignorovať edukačnú robotiku na 1. stupni a podrobnejšie sme preskúmali jej potenciál. Využili sme na to práve edukačného robota Blue-Bot (podrobnejšie v 3. kapitole), pričom sme sa zamerali na gradáciu informatických konštruktov, ktoré sa pomocou neho dajú vyučovať.

Na lepšie vzájomné porozumenie chceme hneď v úvode vysvetliť niekoľko pojmov, ako im rozumieme:

Programátorský koncept – je napr. opakovanie, sekvencia príkazov, otočenie a pod.

Operácie – nad týmito konceptmi sú napr. čítanie, zostavenie, trasovanie, vykonanie programu a pod.

Programovacie konštrukty – sú operácie nad jedným, alebo viacerými konceptmi.

Stav robota – rozumieme pod tým jeho natočenie a pozíciu (na podložke, resp. sieti).

Podložka – je štvorcová sieť v rozmeroch $m*n$, kde jeden štvorček pri použití Blue-Bota má 15 cm.

Políčko – je jeden štvorček na podložke, ktoré môže, ale nemusí obsahovať nejaký tvar (napr. úľ, kvietok, domček, číslo a pod.).

Reader – (TacTile Code Reader) je cez bluetooth spárovaná modrá doska, do ktorej môžu deti vkladať príkazy v podobe malých plastových dielikov. S deťmi ho voláme "modrý panel".

2 METODOLÓGIA

Uvedomujeme si, že veľký vplyv na kvalitné porozumenie má, okrem skúseností a zručností učiteľa, aj softvérové prostredie, programovací jazyk, vizualizácia príkazov, atď. [6] Všetky

uvedené aspekty ovplyvňujú vzdelávací proces a aby sme im lepšie porozumeli, zvolili sme kvalitatívny prístup na zodpovedanie našich výskumných otázok. Ako výskumný nástroj sme zvolili edukačného robota Blue-Bot¹ spolu so sériou aktivít zameraných na programátorské konštrukty, ktoré sme preňho vytvorili. K tomuto nástroju sme sa priklonili z viacerých dôvodov. Na Slovensku sú pomerne rozšírené Bee-Boty už v materských školách, aj preto sme sa rozhodli nadviazať na ne prácou s Blue-Botmi v prvých ročníkoch ZŠ. Tieto Blue-Boty sú cenovo dostupné, užívateľsky nenáročné a pritom ponúkajú paletu zaujímavých programátorských konštruktov a umožňujú žiakom rozvoj mnohých zručností ako napríklad komunikácia, kooperácia, verbalizácia myšlienok a podobne [7, 8, 9].

Naším hlavným výskumným cieľom bolo identifikovať škálu, ktorá by odrážala primeranú kognitívnu náročnosť operácií pri aktivitách pre žiakov prvých ročníkov ZŠ, aby sme tak mohli prispieť k lepšiemu poznaniu náročnosti programátorských konštruktov. Rozhodli sme sa preto položiť si nasledovnú výskumnú otázku:

Ako vyzerajú úrovně kognitívnej náročnosti pri oboznamovaní žiakov 1. stupňa ZŠ so základmi programovania (za pomoci Blue-Botov)?

Najskôr bolo potrebné sa oboznámiť s náročnosťou jednotlivých úloh (programátorských), ktoré sa dajú realizovať s Blue-Botmi. Počas výskumu sme museli preto navrhnuť a overiť viacero úloh, aby sme vedeli zistiť, ako môže vyzeráť vhodná gradácia úloh pre žiakov primárneho vzdelávania s Blue-Botmi, ktorá pre nich vytvára primeranú výzvu a ostáva stále pútavou. Výsledkom je séria 20-tich vyučovacích hodín, ktorá však nie je výstupom nášho výskumu, ale využili sme ju ako súčasť nástroja na overovanie. Keďže ide o rozsiahlu prácu, ktorá by postačovala na celý článok, nebudeme sa jej viac venovať v tomto príspevku. Na základe povahy výskumu sme sa rozhodli pre výskum vývojom [10], ktorý nám umožňuje v iteratívnych cykloch sa približovať k správne usporiadaniu úloh na škále a tak zjemňovať granularitu náročnosti a používaných pojmov. Údaje sme zbierali pomocou terénnych zápisov, videozáznamov, fotodokumentácie a rozhovormi s učiteľmi a žiakmi.

Náš výskum prebiehal v troch rozličných prostrediach. Na základnej škole v hlavnom meste, kde žiaci mali skúsenosť s Bee-Botmi a to v triedach 2. ročníka (15 až 20 žiakov) a 4. ročníka (20 až 24 žiakov). Potom v priľahlej mestskej štvrti na základnej škole v blízkosti veľkého mesta, kde žiaci, ktorí sa zúčastnili výskumu, nemali s takými robotickými hračkami skúsenosti. Tam prebiehal výskum v 1. ročníku (15 až 18 žiakov). A počas uzavretia škôl z dôvodu pandémie koronavírusu, sme overovali jednotlivé aktivity s deťmi v malom počte, prevažne v domácom prostredí. Aktivity sme sa deťom snažili predložiť rovnakým spôsobom, akoby pracovali v škole samostatne. Niekoľkokrát sa nám podarilo vytvoriť aj dvojice, ak išlo o deti v približne rovnakom veku. Zväčša však išlo o samostatnú prácu detí na zadanie.

Podarilo sa nám tak po čiastkách overiť sériu 20 vyučovacích hodín, ktoré obsahujú viaceré aktivity. V kapitole 4 predstavíme kognitívne zastrešenie týchto aktivít. Na základe iterovaných výsledkov z pozorovaní sme navrhli stupne na škále, ktoré vhodne odzrkadľujú kognitívnu postupnosť operácií pri oboznamovaní sa s programovacími konštrukciami na ovládanie edukačného robota Blue-Bot.

¹ Distribútorom pre slovenské školy je napr. PMS Delta s.r.o (<https://pmsdelta.sk/robotika/>)

3 CHARAKTERISTIKA EDUKAČNÉHO ROBOTA

V tejto časti by sme radi priblížili edukačného robota Blue-Bot, s ktorým sme realizovali výskum a jeho predchodcu, robotickú včelu Bee-Bot.

3.1 Bee-Bot a Blue-Bot

Edukačný robot Blue-Bot (Obrázok 1 – vľavo) je malá podlahová programovateľná robotická hračka o rozmere asi ľudskej dlane, ktorá sa hýbe po zemi. Je to nová verzia populárnej robotickej včely Bee-Bot (Obrázok 1 – vpravo), ktorá bola na veľtrhu BETT 2006 ocenená ako najpôsobiljší hardvér pre deti materských a základných škôl. Využíva relatívne riadenie (v štvorcovej sieti), rovnako ako korytnačka v jazyku Logo. [11]



Obrázok 1 - Blue-Bot a Bee-Bot

S Blue-Botom môžeme pracovať rovnako ako s Bee-Botom, teda riadiť ho pomocou tlačidiel na ňom. Po zadaní jednotlivých príkazov (dopredu, cúvni, vpravo, vľavo) a stlačení tlačidla GO sa spustí vykonávanie zadaných príkazov. Po zadaní ďalších príkazov sa predchádzajúce automaticky nevymažú, ale sa pridávajú k už zapamätanej postupnosti. Bee-Botovi môžeme v jednej sekvencii zadať 40 príkazov. Blue-Bot si ich vie zapamätať viac ako 200. Textové pomenovania tlačidiel CLEAR a PAUSE boli u Blue-Bota nahradené ikonami, takže sú zrozumiteľnejšie aj pre deti, ktoré nevedia čítať. Nevýhodou tohto ovládania môže byť, že si deti nepamätajú postupnosť, ktorú robotovi zadali, ale aj to, že zabúdajú pred novou postupnosťou starú postupnosť vymazať.

Blue-Bota však môžeme ovládať aj inými spôsobmi:

- pomocou tabletu alebo počítača (akoby diaľkovým ovládačom),
- pomocou readra (Obrázok 2).



Obrázok 2 - TacTile Code Reader aj s ukázkou programu

Tablet (počítač) aj reader sa s Blue-Botom spájajú pomocou bluetooth. Ak je Blue-Bot so zariadením spojený (spárovaný), zasvieti na modro.

Pri práci s tabletom alebo počítačom je potrebné mať nainštalovanú aplikáciu, pomocou ktorej môžeme riadiť Blue-Bota a to buď priamo – zadáme jeden príkaz a Blue-Bot ho automaticky vykoná, ale aj naplánovaním postupnosti príkazov, ktoré vidíme na

obrazovke, a ktoré potom (po presunutí virtuálneho robota na nejaké políčko a kliknutí na tlačidlo GO v aplikácii) Blue-Bot vykoná.

Pomocou modrého panela plánujeme postupnosť príkazov, ktorú potom pošleme Blue-Botovi na vykonanie.

3.2 Relatívne a absolútne riadenie

V programovaní, ktoré realizujeme s deťmi na počítači, je prirodzenejšie, keď s absolútnym riadením pracujeme skôr ako s relatívnym.

Agentu (postavička, ktorú ovládame) v prostrediach s absolútnym riadením ovládame buď šípkami na klávesnici alebo tlačidlami s rovnakým významom, kde ↑ znamená presun agenta o jedno políčko smerom hore, ↓ presun agenta o jedno políčko dolu, → presun o jedno políčko vpravo a ← o jedno políčko vľavo od aktuálnej pozície. Takýmto spôsobom riadime aj robota Emila v druhom svete v Robot Emil 3 [12] (od priameho riadenia klikaním na políčka v prvom svete sa v druhom svete prešlo na riadenie pomocou tlačidiel so šípkami, Obrázok 3). Pri tomto type riadenia neriešime smer natočenia agenta. Častokrát sa agent ani neotáča.



Obrázok 3 - Absolútne riadenie robota Emila

Aj pri relatívnom riadení môžeme agenta ovládať šípkami (častejšie však už príkazmi), ktoré majú pri takomto riadení iný význam. ↑ znamená posun dopredu o jedno políčko v smere natočenia agenta, ↓ posun vzad o jedno políčko v danom smere, → otočenie sa na mieste o 90° vpravo a ← otočenie sa na mieste o 90° vľavo bez následného posunutia. Teda na presun o jedno políčko vpravo od aktuálnej pozície potrebujeme zvyčajne viac príkazov ako jeden, ak práve nie je agent otočený v správnom smere. Vo virtuálnom prostredí je takéto riadenie pre deti menej prirodzené, lebo si súčasne musia uvedomovať viac skutočností – pozíciu, otočenie, kam sa treba a ktorým smerom posunúť... Robota Emila tiež riadime aj relatívne, ale až v Robot Emil 4 (Obrázok 4).

Pomocou robotických hračiek Blue-Bot (ktoré sa riadia relatívne) však s deťmi môžeme prirodzene pracovať aj s relatívnym riadením. Nepracujeme tu vo virtuálnom prostredí, ale v reálnom. Deti tak môžu nasledovať robota (postaviť sa zaňho a napodobňovať jeho pohyby). Deti sa tak s ním stotožnia a ľahšie potom vedú, ako majú ďalej pokračovať.



Obrázok 4 - Relatívne riadenie robota Emila

4 VÝSLEDKY VÝSKUMU

Pri vytváraní škály sme vychádzali z našich dlhoročných skúseností, ktoré máme na katedre s vyučovaním informatiky v primárnom vzdelávaní a s vyučovaním za pomoci robotickej hračky Bee-Bot v materských školách [11,13]. Vytvorená škála reflektuje aktivity, pri vývoji ktorých sme sa zamerali na postupné gradovanie náročnosti informatických konštruktov, prirodzene rešpektujúc kognitívnu vyspelosť detí daného veku. Ako sme písali v časti metodológia, samotné aktivity nie sú cieľom nášho výskumu, ale len nástrojom, preto sa im nebudeme venovať podrobnejšie, ale predstavíme kognitívne zastrešenie týchto aktivít.

Nami vytvorená škála má 4 úrovne, v ktorých sa zameriavame na gradovanie štyroch rôznych oblastí.

- Spôsob práce** žiakov s edukačným robotom.
- Náročnosť podložky**, ktorá graduje na základe rozmerov ale hlavne prítomnosťou objektov na jej políčkach.
- Stav edukačného robota**, ktorý je určený určením pozície a smerovým natočením. Najčastejšie sa jedná o určenie počiatočnej alebo koncovkej pozície.
- Programovacie konštrukty**. Keďže robot nemá veľa príkazov, ktorými by sa dal ovládať, ako majú napr. textové alebo detské programovacie jazyky, podstatu týchto konštruktov vidíme v náročnosti programov, ktoré majú žiaci vytvárať; v obmedzeniach, ktoré im určujú plánovanie cesty (napr. zákaz používania daného príkazu alebo nepriechodné miesta na podložke, ...).

1. Úroveň

Ovládanie: Riadenie robota cez reader (S jednou kartičkou – priame riadenie. Ak dáme viac kartičiek – riadenie programovaním.)

Náročnosť podložky: každá pozícia na podložke je jednoznačne určená objektom, respektíve obrázkom.

Stav robota: počiatočný smer nie je riešený žiakmi, pretože je robot vždy rovnako natočený. V rôznych úlohách je jeho pozícia samozrejme rôzna, aj keď daná. Tieto pravidlá ostávajú na tejto úrovni stále rovnaké.

Programovacie konštrukty:

- čítanie a interpretovanie krátkych programov
- zakreslenie trasy podľa daného programu
- riadenie robota z miesta A do B bez obmedzení
- riadenie robota z miesta A do B s podmienkou prechodu danou (medzi)pozíciou
- identifikácia možných koncových pozícií do vzdialenosti 3 príkazov

- reverzné trasovanie už prejdenej cesty
- riadenie robota bez príkazu Dopredu

2. Úroveň

Ovládanie: Riadenie robota pomocou jeho tlačidiel (program žiak nevidí).

Náročnosť podložky: Niektoré pozície na podložke sú voľné a teda nie sú jednoznačne identifikovateľné.

Stav robota: Žiaci sa oboznamujú s určovaním smeru a natočením robota pomocou smerovej ružice (hore, dole, vpravo, vľavo).

Programovacie konštrukty:

- pridávanie obmedzení pri prechode cez podložku:
 - zakázaný prechod niektorými hranami
 - zakázaný prechod niektorými pozíciami
 - povinnosť dodržať poradie navštívených pozícií (objektov)
- identifikácia možných koncových pozícií do vzdialenosti 3 a 4 príkazov
- kombinácia zákazov:
 - prechod políčkom + 1 príkaz
 - prechod políčkom + 2 príkazy

3. Úroveň

Ovládanie: Plánovanie robota pomocou papierových kartičiek, ktoré menia z implicitného na explicitné. Podobá sa to riadeniu cez reader, len programy môžu byť dlhšie.

Náročnosť podložky: Pozície na podložke sú označené abstraktnejšie. Buď písmenami alebo číslami.

Stav robota: Natočenie robota je definované pomocou výrazov: “smerom K”, “smerom Od”.

Programovacie konštrukty:

- spúšťanie a trasovanie rovnakého programu z rôznych pozícií a s rôznym natočením
- plánovanie dlhších programov (pomocou papierových kartičiek)
- zákaz použitia najprv 1 (ľubovoľného) príkazu, potom 2 príkazov
- doplnenie záverečných príkazov programu – podľa zadania
- určenie počiatočného natočenia – mám daný program a cieľ
- tvorba zadání s podmienkami

4. Úroveň

Ovládanie: Kombinovanie predošlých úrovní. Naplánovanie cesty (zakreslením cesty do obrázku podložky, alebo šípkami) a následná realizácia. Plánovanie paralelného správania dvoch robotov

Náročnosť podložky: pracuje sa s podložkou bez označenia políčok alebo bez podložky, teda bez priestorového obmedzenia.

Stav robota: žiaci musia sami určiť podľa zadania.

Programovacie konštrukty:

- hľadanie a oprava chyby v danom programe
- plánovanie správania dvoch robotov pri paralelnom vykonávaní programov s podmienkou: nezraziť sa a prechod povolenými pozíciami
- identifikácia postupnosti príkazov podľa vizualizácie
- tvorba postupnosti príkazov pre pohyb dvoch robotov podľa stredovej a osovej súmernosti

5 ZÁVER A DISKUSIA

V článku sme predstavili predbežné výsledky kvalitatívneho výskumu, ktorého cieľom bolo: *Identifikovať úrovne kognitívnej*

náročnosti a ich obsah pri oboznamovaní sa žiakov 1. stupňa ZŠ so základmi programovania (za pomoci Blue-Botov). Identifikovali sme štyri úrovne, ktoré reprezentujú gradovanie kognitívnej náročnosti pri vzdelávacích aktivitách. Tieto úrovne škály nám vedía v budúcnosti, okrem iného, pomôcť pri vytváraní primeraných aktivít, či pri posudzovaní ich náročnosti a pod. Každá úroveň škály zahŕňa popis náročnosti podložky, stavu robota, ovládania robota žiakmi a programovacie konštrukty. Aj keď sa môže zdať, že úrovne na škále sa dotýkajú aj psychomotorických činností, zamerali sme sa v tomto článku hlavne na gradovanie kognitívnych úkonov pri práci s robotom. Žiaci by pri práci s robotom mali prejsť všetkými úrovňami Bloomovej taxonómie, samozrejme primerane ich veku. To znamená napr. od najnižších úrovní, kedy vedía identifikovať alebo vlastnými slovami vysvetliť zmysel štartovacieho políčka, cez aplikovanie pravidiel pri vytváraní cesty, až po plánovanie tanečného paralelného pohybu viacerých robotov. Uvedomujeme si však, že ďalší výskum bude požadovať hlbšie preskúmanie jednotlivých operácií s programovacími konceptmi, aby sme ich vedeli presnejšie priradiť do určitých stupňov kognitívnej taxonómie.

Za slabé miesta výskumu vnímame, že sa nedalo pri overovaní niektorých aktivít pracovať s celou triedou, lebo sa do škôl nechodilo. Problematické bolo aj vytváranie podložiek, keďže ide o nadrozmerne papiere (A2 až A1), na ktoré v škole tlačiareň, resp. ploter nemáme. Jednou z možností bola tlač na A4, následné vystrihovanie a lepenie na tvrdé kartóny, čo však zabralo enormné množstvo času. Neprijemnou situáciou bolo aj prerábanie gradácie úloh z dôvodu vylúčenia tabletu na ovládanie robota. Pre náš iteratívny vývoj to znamenalo pridanie niekoľkých kôl iterovania na viac. Dôvodom vylúčenia bola zmena aplikácie, ktorá už našim požiadavkám nevyhovovala. Pôvodnú aplikáciu, ktorá obsahovala iba tlačidlá na ovládanie robota, nahradila aplikácia, v ktorej sa tlačidlá radikálne zmenšili a veľkú časť plochy zaberá robot, ktorý sa otáčal podľa zadávaných tlačidiel. Avšak deti sa často s reálnym robotom stotožnia – nasledujú ho, otáčajú sa presne ako on a v takomto prípade vizualizácia na tablete nekorešponduje s reálnou situáciou. V konečnom dôsledku to vytváralo u žiakov viac miskoncepcií ako prehĺbenie porozumenia. Aj z tohto dôvodu sme zaradili najprv aktivity s readrom. Niektorí môžu namietať, že by bolo pre žiakov jednoduchšie pracovať najprv s tlačidlami na robotovi a až potom s readrom. Avšak pri stláčaní tlačidiel, ší žiaci vytváraný program musia uchovávať v pamäti a pri každom novom príkaze ten pôvodný z pamäti robota vymazať, čo so skúsenosťami vieme, im robí problémy. Preto je lepšia vizualizácia príkazov cez reader, kde pracujú na začiatku s úlohami, kde programy majú dĺžku len jedného príkazu.

Veríme, že nami vytvorená škála gradácie prispeje ku kvalitnejším poznatkom didaktiky informatiky. Medzi neplánované overenia nám poslúžili viaceré príklady, ktoré sme zaznamenali počas overenia, dva z nich uvádzame. Pri 1. úrovni žiaci absolútne plynulo prechádzajú od priameho riadenia k plánovaniu. Najprv položili jednu kartičku do readera a nechali ho vykonať, následne druhú a nechali ho vykonať atď. a samovoľne začali pokladať viacero karičiek vedľa seba, čím začali plánovať program. Ako druhý príklad uvádzame situáciu z 3. úrovne, kedy žiaci začali vytvárať dlhšie programy, pričom občas spravili chybu a začali ju spolu hľadať v pláne z kartičiek a opravovať a teda nastala samovoľná propedeutika k debugovaniu, ktoré sa nachádza o úroveň vyššie.

Ako sme spomenuli v úvode, ide o prvotné výsledky dlhodobejšieho výskumu, v ktorom budeme musieť pokračovať ešte ďalšími overovaniami. Prispieť by k tomu malo aj znovu otvorenie škôl.

hračky. Bratislava : Metodicko-pedagogické centrum, 2013.
41 s. ISBN 978-80-8052-582-8

POĎAKOVANIE

Radi by sme sa poďakovali Ivanovi Kalašovi, ktorý je spoluautorom aktivít, bez ktorých by tento výskum nebol možný. Materiál s aktivitami sa práve pripravuje pre tlač. Taktiež veľké ďakujem patrí aj všetkým deťom, pani učiteľkám, Daniele Bezákovej a Romanovi Hrušeckému, ktorí sa podieľali na overovaní aktivít. Poďakovať sa chceme aj neziskovej organizácii Indícia, ktorá celý proces zastrešovala a projektu VEGA 1/0602/20, vďaka ktorému môžeme pracovať na hlbšom porozumení tohto obsahu a publikovať získané výsledky.

BIBLIOGRAFICKÉ ODKAZY

- [1] Projekt: Vzdelávanie pedagogických zamestnancov materských škôl ako súčasť reformy vzdelávania <https://archiv.mpc-edu.sk/sk/projekty/mat> (2007-13)
- [2] Včela Bee-Bot <https://www.tts-international.com/bee-bot-programmable-floor-robot/1015268.html>
- [3] Neinformatick=e aktivity so včelou BeeBot https://www.infracz.cz/data_13/soubory/365.pdf
- [4] LYDON, A.: Let's Go with Bee-Bot. Using your Bee-Bot across the curriculum. TTS Group. Ltd. 2007.
- [5] MANĚNOVÁ, M., PEKÁROVÁ, S., Algoritmizace s využitím robotických hraček pro děti do 8 let Univerzita Hradec Králové, 2020. s. 72 ISBN 978-80-7435-775-6. Dostupné tiež na: <https://imysleni.cz/ucebnice/rozvoj-informatickeho-mysleni-s-vyuzitim-robotickych-hracek-v-materske-skole-a-na-1-stupni-zs>
- [6] KELLEHER, C., PAUSCH, R. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. ACM Computing Surveys (CSUR) 2005, 37(2), s. 83-137.
- [7] BERS, M. U.: *Designing digital experiences for positive youth development: From playpen to playground*. Oxford University Press, 2012.
- [8] KHANLARI, A. Effects of robotics on 21st century skills. European Scientific Journal, 9(27). 2013.
- [9] EGUCHI, A.: Educational robotics for promoting 21st century skills. In *Journal of Automation Mobile Robotics and Intelligent Systems*. 2014, s. 5-11.
- [10] VAN DEN AKKER, J., GRAVEMEIJER, K., MCKENNEY, S., & NIEVEEN, N. (Eds.). Educational design research. Routledge, 2006.
- [11] PEKÁROVÁ, J.: Using a Programmable Toy at Preschool Age: Why and How? In *Workshop Proceedings of SIMPAR 2008*. Venice, 2008, s. 112-121. ISBN 978-88-95872-01-8
- [12] KALAŠ, I., BLAHO, A., MORAVČÍK, M.: Exploring Control in Early Computing Education. In: *Sergei N. Pozdniakov and Valentina Dagiene (eds.) Informatics in Schools. Fundamentals of Computer Science and Software Engineering*. Springer, LNCS 11169, s. 3-16. ISBN 978-3-030-02750-6.
- [13] PEKÁROVÁ, J., ZÁHOREC, J., HRUŠECKÝ, R.: *Digitálne technológie v materskej škole 3 multimédiá*. Digitálne

Analýza edukačných robotických hračiek dostupných na Slovensku

Analysis of educational robotic kits available in Slovakia

Karolína Miková

FMFI UK
Mlynská dolina
842 48 Bratislava
Slovensko

Karolina.Mikova@fmph.uniba.sk

Lucia Budinská

FMFI UK
Mlynská dolina
842 48 Bratislava
Slovensko

Lucia.Budinska@fmph.uniba.sk

Barbora Stenová

FMFI UK
Mlynská dolina
842 48 Bratislava
Slovensko

stenova7@uniba.sk

ABSTRACT

In this article we want to present the results of our analysis, which focuses on the analysis of educational robotic kits available in Slovakia, with a focus on the potential it can have for teaching computing in elementary and middle schools (6-15 years old pupils). The result of our research is categorisation of both software and hardware criterion. On one hand, software environment with specific programming concepts used in robotic control are described, on other hand there are also described one specific feature in educational robotics which is interactivity. This is a feature that distinguishes classic programming in text languages, cards or other common programming languages from robot programming and makes it more attractive. These categories allow us to better understand the potential of a given kit. These are the initial results of this research, which we want to extend over time to other kits and their software.

Keywords

Elementary school. Robotic kits. Educational potential. Programming concepts. Interactivity.

ABSTRAKT

V tomto článku chceme priblížiť výsledky našej analýzy, ktorá sa zameriava na analýzu edukačných robotických hračiek dostupných na Slovensku, s dôrazom na odhalenie potenciálu, ktorý môžu mať pre vyučovanie informatiky na základných školách (čiže 6-15 ročných žiakov). Výsledkom nášho výskumu sú kategórie opisujúce na jednej strane softvérové prostredia, konkrétne programovacie koncepty na ovládanie vybraných robotických stavebníc, a na strane druhej veľmi charakteristickú vlastnosť typickú pre edukačnú robotu - interaktivitu. Ide o vlastnosť, ktorá odlišuje klasické programovanie v textových, kartičkových, či iných bežných programovacích jazykoch od programovania robotov a robí ho atraktívnejšie. Tieto kategórie nám umožňujú lepšie porozumieť potenciálu, ktorý má daná stavebnica. Ide o počiatočné výsledky tohto výskumu, ktorý chceme rozšíriť časom aj pre ďalšie stavebnice a ich softvér.

Kľúčové slová

Základná škola. Robotické stavebnice. Edukačný potenciál. Programovacie koncepty. Interaktivita.

1 ÚVOD

Edukačná robotika sa postupne dostáva na veľkú časť slovenských škôl. Istý podiel na tom má aj dostupnosť robotických hračiek a stavebníc na slovenskom trhu. To však nemôže byť jediný parameter, podľa ktorého by sa mali vyberať stavebnice do škôl. Samozrejme, je veľkou výhodou, ak je stavebnica bežne predávaná

na Slovensku a je možné riešiť jej reklamácie, či prípadne dokupovanie rovnakých verzií bez väčších problémov a dlhých čakacích dôb, čo nákup zo zahraničia nie vždy spĺňa. Dôležitá je aj cena, najmä preto, že pri použití v triede je nutné vlastniť viac kusov robotov či stavebníc.

Na druhej strane sú parametre, ktoré nemajú jasné ohodnotenie. Medzi tie vieme zaradiť vhodnosť stavebnice pre konkrétny vek a programovateľnosť stavebnice, aby okrem motivačného efektu pomáhala žiakom rozvíjať svoje algoritmické myslenie a programátorské zručnosti. Keďže pri edukačnej robotike žiaci okrem softvéru, v ktorom programujú robota, pracujú aj fyzicky priamo so stavebnicou, je to nemenej dôležitý aspekt, ktorý potrebujeme brať do úvahy.

V tomto článku sa bližšie pozrieme na niekoľko robotických hračiek vhodných pre mladší školský vek dostupných na slovenskom trhu a predstavíme parametre, podľa ktorých môžu učitelia vybrať robotickú stavebnicu vyhovujúcu ich podmienkam a zámerom.

2 METÓDY VÝSKUMU

Cieľom nášho výskumu bolo detailne preskúmať potenciál dostupných edukačných robotických hračiek z pohľadu programovania a interaktivity. Zamerali sme sa na hračky vhodné pre nižší školský vek, čiže pre prvý a začiatok druhého stupňa základných škôl. Kritériá, podľa ktorých sme vybrali analyzované stavebnice, sú opísané v nasledujúcej kapitole.

Keďže sme analyzovali možnosti stavebníc a zároveň sme sa zameriavali aj na didaktické vlastnosti softvérových prostredí určených na ich programovanie, rozhodli sme sa v tejto fáze pre teoretický výskum. Využili sme analýzu produktov [1] - a to hardvéru, aj softvéru. Pri analyzovaní softvérov jednotlivých robotických hračiek sme použili analytickú indukciu [2]. Pri tvorbe kategórií to znamenalo, že sme vedeli buď koncepty/vlastnosti... zaradiť do už vytvorených kategórií, alebo sme boli nútení tieto kategórie redefinovať.

Výsledkom nášho výskumu je kategorizácia programovacích konceptov a interaktivity týchto produktov, ktorá vznikla postupným analyzovaním. Pre prehľadnejšie zobrazenie sú vzniknuté kategórie uvádzané spolu s produktami v tabuľkách.

3 PREHLAD VYBRANÝCH STAVEBNÍC

Pre výber produktov sme mali niekoľko základných kritérií, ktoré predpokladáme, by mali aj učitelia základných škôl pri výbere robotických hračiek. (1) Museli byť dostupné na zakúpenie a servis na Slovensku, (2) mať vizuálne programovacie prostredie vhodné pre žiakov ZŠ podľa aktuálneho iŠVP [3] alebo byť priamo

programovateľné bez použitia dodatočného softvéru, (3) samotná manipulácia so stavebnicou, či zapájanie jednotlivých komponentov, by nemali vyžadovať veľa času, aby sa väčšia časť hodiny v škole mohla venovať programovaniu. Zároveň tak učiteľ môže s jednou sadou stavebníc nechať pracovať rôzne triedy žiakov bez toho, aby to negatívne ovplyvňovalo ich prácu.

Do tohto prvého výberu sme zaradili edukačné roboty Ozobot Bit a Evo [4], Botley [5], stavebnicu LEGO WeDo 2.0 [6] a Micro:bit [7]. Tie bližšie opíšeme v nasledujúcich častiach.

Na slovenských školách sú pomerne rozšírené aj stavebnice LEGO Mindstorms Ev3 a jeho staršia verzia LEGO NXT. Tieto stavebnice však nespĺňajú tretie kritérium, keďže stavba modelov z nich je časovo náročná (už len stavba základného modelu trvá približne 60 min.). Zároveň, hoci tieto stavebnice sa dajú programovať pomocou vizuálneho jazyka, ich programovanie je náročnejšie a skôr vhodné pre žiakov, ktorí už s robotikou majú skúsenosti, respektíve už majú skúsenosti s programovaním. Priestor na využitie takýchto stavebníc vidíme skôr v mimoškolskej činnosti [8, 9], na súťažiach [10,11] a to nie len doma, ale aj v zahraničí [12,13].

3.1 Ozobot

Ozobot (na Obrázku 1) je robot tvaru podobného polguli s výškou 2,5 cm. Neponúka možnosť stavania modelov, je však možné ho upraviť využitím takzvaných skinov, čiže obalov, ktoré vedú zmeniť Ozobotovi farbu či tému. Je dostupný v dvoch verziách - Bit a Evo. Vo verzii Bit obsahuje optický senzor a senzor farieb, jedno LED svetielko a samozrejmosťou je pohyb (spolu s predprogramovaným pohybom po čiare). Verzia Evo ešte navyše obsahuje ďalších 5 LED svetielok, reproduktor a 4 senzory vzdialenosti a Bluetooth. Tomu zodpovedá aj cena, kým Ozobot Bit stojí okolo 70 €, Ozobot Evo 130 €. Pre školy sú určené aj zvýhodnené sady obsahujúce viac kusov.

Oba Ozoboty sa vedú pohybovať po čiarach nakreslených fixkou, dokážu rozpoznať aj farebné kódy, a tiež je možné ich programovať v prostredí OzoBlockly, ktoré aj v ďalšej kapitole práce analyzujeme. Toto prostredie je založené na programovacích blokoch Google Blockly a ponúka 5 úrovní náročnosti od najjednoduchších po najnáročnejšie. V jednoduchších úrovniach sú použité iba ikonické kartičky, vo vyšších úrovniach sú na blokoch texty. OzoBlockly neponúka slovenčinu, je možné ho ale prepnúť do češtiny, či ponechať angličtinu. Programy robot načítava tzv. blikaním, vo verzii Evo je možný prenos cez Bluetooth, rovnako tak aj priame riadenie robota cez spárované zariadenie. Nabíjajú sa cez USB kábel a nabíjanie trvá cca 1 hodinu, rovnaký čas vydržia aj v prevádzke.



Obrázok 1: Ozobot Evo

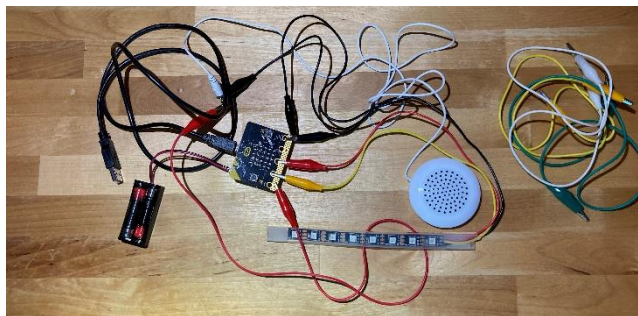
Pohyb robota sa programuje presne - určením vzdialenosti (resp. v nižších úrovniach určením kroku) a rýchlosti, otáčanie sa nastavuje pomocou uhla (vo vyšších úrovniach zadaním presného čísla).

3.2 Micro:Bit

Micro:bit je mikropočítač vo veľkosti kreditnej karty, obsahuje 2 tlačidlá, 25 LED svetielok (v mriežke o rozmeroch 5×5), pohybový senzor (kompas a senzor naklonenia), vstupno-výstupné piny na pripojenie ďalších zariadení, vstup na USB a vstup na pripojenie externej batérie. V základnej verzii bez ďalších zariadení je tak použiteľný na programovanie s využitím tlačidiel ako vstupu a LED mriežky ako výstupu. Pripojením ďalších zariadení (ako napr. reproduktor, LED pásik, ďalšie tlačidlá či kontroléry, ale aj konštrukcie s kolieskami) je vhodný na všestranné použitie aj mimo informatiky. Cena samotného Micro:bitu je cca 15 €.

Micro:bit sa dá programovať pomocou blokového prostredia Make Code, ktoré analyzujeme v nasledujúcej kapitole, následne sa dá využiť aj programovanie pomocou jazyka Scratch a pre pokročilých je možné Micro:bit programovať v jazyku Python. Prostredie Make Code je lokalizované do slovenčiny, čo je pozitívom, keďže programovacie bloky obsahujú textové príkazy. Program sa do Micro:bitu sťahuje pomocou USB pripojenia alebo cez Bluetooth zo spárovaného zariadenia.

Na používanie je potrebné mať k Micro:bitu pripojené externé baterky alebo ho nechať cez USB kábel zapojený do iného zariadenia.



Obrázok 2: Micro:bit s príslušenstvom

3.3 Botley

Botley je malý robot kvádrovitého tvaru s dvoma kolesami po bokoch a "očami" vpredu. Medzi očami sa nachádza senzor na detegovanie predmetov, na vrchnej strane sa nachádzajú štyri LED svetielka, ktoré znázorňujú vykonávajúce sa príkazy. Obsahuje aj senzor na sledovanie čiar a odnímateľné ruky pre robota. Balenie tiež obsahuje dieliky na stavbu prekážok. Cena balenia je cca 40 €.

Botley sa programuje pomocou diaľkového ovládača so štyrmi tlačidlami na pohyb, štyrmi doplnkovými tlačidlami (slučka, vymazanie programu, nastavenie hlasitosti, detekcia predmetov) a veľkým tlačidlom na prenos programu. Tým, že takéto programovanie neumožňuje žiakov vidieť zápis programu, v balení sa nachádzajú aj kartičky s príkazmi, z ktorých si žiaci môžu skladať vykonávanú postupnosť a vytvárať tak zápis svojich programov. Robot sa hýbe o preddefinovanú vzdialenosť dopredu/dozadu, ktorú nejde zmeniť, otáča sa vždy o 90 stupňov. Na jeho fungovanie sú potrebné batérie jednak priamo v robote, jednak v ovládači.



Obrázok 3: Botley

3.4 LEGO WeDo 2.0

LEGO Education ponúka širokú paletu robotických stavebníc pre školy. Stavebnica LEGO WeDo 2.0. je určená pre 1. stupeň ZŠ. Oproti prvej verzii pribudla možnosť ovládať model pomocou Bluetooth pripojenia, čím sa programovanie a stavanie stáva nezávislým od počítačov. Stavebnica obsahuje základný smart hub, ktorým sa spúšťa a zastavuje program. Do neho sa dá pripojiť motor a senzory – v balení sa nachádzajú senzor pohybu a senzor náklonu. Samozrejmosťou je viac ako 200 dielikov LEGA a iných súčiastok (napr. kolesá) z ktorých je možné tvoriť predpripravené modely, ale aj modely podľa vlastnej fantázie žiakov. Robot potrebuje na svoju prevádzku baterky.

V aplikácii, v ktorej sa robot programuje, sa nachádza viacero návodov na postavenie rôznych modelov. Návody sú obrázkové a obsahujú aj videá a sprievodné texty, ktoré sú však v angličtine. Softvér je lokalizovaný do viacerých jazykov, slovenčina a čeština to však nie sú.

Programovacie prostredie je ikonické, vhodné pre žiakov na prvom stupni. Pri programovaní sa programuje rýchlosť a otáčanie motora, čo znamená, že žiaci musia dbať aj na správne konštrukčné zostrojenie modelu, a zároveň to umožňuje širšie možnosti programovania pohybu (nielen celého modelu, ale aj jeho časti, napr. závera). Základná súprava stojí okolo 200 €.



Obrázok 4: LEGO WeDo 2.0

4 Výsledky

Po identifikácii vhodných edukačných robotov, sme sa zamerali na cieľ nášho výskumu, na analýzu softvérových prostredí, resp. na príkazy, ktorými sú roboty programovateľné. Keďže cieľom nášho výskumu bolo detailne preskúmať potenciál dostupných

edukačných robotických hračiek z pohľadu programovania a interaktivity, zamerali sme sa hlavne na programovacie konštrukty, ktorými softvérové prostredia disponujú, a interaktivitu, ktorá je tak charakteristická pre robotiku. Na pomedzí týchto dvoch oblastí nám vznikla tretia samostatná oblasť týkajúca sa rozširovania sa robota. Ide o príkazy, ktorými sa ovláda pohyb robota. Ide o veľmi autentickú súvislosť medzi konkrétnym modelom a príkazom na pohyb. Preto sme sa rozhodli túto oblasť odčleniť a prezentovať samostatne.

4.1 Interaktivita

Interaktivita je základnou črtou, ktorá odlišuje programovanie robota od bežného programovania. Jednoducho povedané, zjednodušuje komunikáciu medzi ľuďmi a robotmi. Pri vyučovaní ju vieme žiakom pripodobniť k počúvaniu, rozprávaniu a správaniu človeka. Pri analyzovaní príkazov, ktoré reprezentovali interakciu robota s okolím, nám výsledky ukázali, že sa jedná o tri skupiny interakcií.

4.1.1 Vstupná

je spojená s prevodom informácií z okolitého prostredia na vstupy, ktoré môžu byť vyhodnocované alebo spracované rôznym spôsobom. Najčastejšie išlo o rôzne senzory ako senzor vzdialenosti, náklonu a tlačidlá. Najviac ich mal Micro:bit.

4.1.2 Výstupné

sú rôzne reakcie, ktoré robot vie vykonať. Vieme ich zaradiť do troch oblastí: pohyb, zvuk a svetlo. Keďže pohybu sa budeme venovať samostatne, predstavíme medzi výstupnou interaktivitou zvuk a svetlo. V Tabuľke 1 vidíme typy svetla, ktorými roboty disponujú, buď ide o zabudované LED svetlá umiestnené podľa konštrukcie (na vrchu, alebo vpredu) alebo doplnkové svetlo, ktoré je možné robotovi pripnúť. Potom sú to rôzne nastavovacie príkazy a svetelné efekty.

Tabuľka 1: Výstupná Interakcia - Práca so svetlom

	Typ svetla			Nastavenie farby				
	Horné svetlo	Predné svetlo	Doplnkové svetlo	Výberom	Pomocou RGB parametrov	Náhodná farba	Nastavenie jasú	Svetelné efekty
Bit	✓			✓	✓	✓	✓	✓
Evo	✓	✓		✓	✓	✓		✓
WeDo		✓	✓	✓		✓		
Micro:Bit		✓	✓		✓	✓	✓	✓
Botley	✓							

Pri zvukoch vidíme, že príkazov už je menej. Jedným z vysvetlení môže byť, že ide o zložitejšie, a teda aj cenu zvyšujúci prvok, ktorý sa výrobcovia rozhodli nezapojiť. Micro:bit to má riešené zas prídavným zariadením. V Tabuľke 2 vidíme, že napr. Ozobot Bit nevie vydávať zvuky vôbec, na rozdiel od jeho nástupcu, ktorý má prácu so zvukom pomerne rozsiahlu, samozrejme v závislosti od úrovne, v ktorej programujeme. LEGO WeDo 2.0 má databázu asi 20-tich zvukov, ale vie si nahráť aj ľubovoľný zvuk, ktorý vie podľa programu prehrať.

Tabuľka 2: Výstupná Interakcia Práca so zvukom

	Tón			Emócie	Povedz číslo, smer, farbu	Nastavenie tempa	Nastavenie hlasitosti
	Stupnica	MIDI	Zvukový súbor				
Bit							
Evo	✓	✓	✓	✓	✓		
WeDo		✓	✓	✓	✓		
Micro:Bit	✓	✓	✓			✓	✓
Botley				✓			✓

4.1.3 Vstupno-výstupné

sme definovali ako zabudované funkcie, na základe ktorých robot po prijatí vstupnej informácie zo senzora zareaguje preddefinovaným spôsobom. Týchto funkcií nebolo až tak veľa. Všetky roboty okrem WeDo a Micro:bitu vedeli sledovať čiaru. Ozobot Evo ešte navyše vedel zasvietiť takou farbou, aká bola práve pod ním.

4.2 Programovacie koncepty

Keďže pri našej analýze vychádzame len z dát, ktoré máme k dispozícii, predkladáme v rámci výsledkov nášho výskumu len také programovacie koncepty, ktoré boli prítomné v softvérových prostrediach. Uvedomujeme si, že existujú aj ďalšie, ktoré neuvádzame, tie sa však nenachádzali v dátach. Ohľadne Ozobotov uvádzame všetkých 5 úrovní, ktoré ponúka softvérové prostredie nastaviť, keďže práve výber týchto konceptov určuje gradovanie náročnosti úrovni.

Tabuľka 3: Programovacie konštrukty

	Ozobot Bit					Ozobot Evo					WeDo 2.0	Micro:bit	Botley			
	1	2	3	4	5	1	2	3	4	5						
Cyklus	nekonečný cyklus													✓	✓	
	s pevným počtom opakovaní													✓	✓	✓
	s riadiacou premennou														✓	
	podmienený													✓	✓	
	cez pole														✓	
Vetvenie	jednoduché													✓	✓	✓
	zložené													✓	✓	
Premenné	vlastná premenná (celé číslo)														✓	✓
	vlastná premenná (reťazec)													✓	✓	
	pole														✓	
Parametre	číslo													✓	✓	
	text													✓	✓	
	vstup zo senzoru													✓	✓	✓
Vlastné procedúry	bez parametrov													✓	✓	
	s parametrami													✓	✓	✓
	s návratovou hodnotou													✓	✓	
Matematické výrazy	" + - * / "													✓	✓	✓
	zložitejšie													✓	✓	

4.3 Pohyb

Samostatne by sme sa chceli venovať programovacím konceptom, ktoré sú typické pre robotiku. Definujú spôsob hýbania sa robotov, ktorý je na rozdiel od objektov v programovacích jazykoch či virtuálnych prostrediach, reálny. Pri programovacích konceptoch sme pracovali s 5-timi úrovňami programov, pri pohybe je to jedno, preto úrovne neuvádzame.

V prvom rade sme určovali absolútny (krok v smere) a relatívny (iba otočenie a krok vpred a vzad) pohyb robota. Teda nejaký základný spôsob pohybu. Pri komplikovanejších robotoch bolo možné hýbať robotom oboma pohybmi. Pre jednoduchších robotov bola dĺžka kroku konštantná (ako Botley), pri zložitejších robotoch nastaviteľná (WeDo). Okrem dĺžky kroku majú niektoré roboty aj možnosť nekonečného pohybu, či nastavenie rýchlosti svojho pohybu. V tomto vidíme zásadné rozdiely pre druhy aktivít, ktoré môžu vzniknúť pri takomto hýbaní robota.

V Tabuľke 4 môžeme vidieť, že Micro:bit nedisponuje primárnymi príkazmi na ovládanie pohybu. Napriek tomu umožňuje ovládať množstvo príslušenstva, ktoré môžu byť aj pohybovým zariadením. Vieme, že existuje aj model s kolieskami, ale keďže to nie je základná zabudovaná funkcia, nebrali sme ju pri analýze do úvahy.

Tabuľka 4: Ovládanie pohybu robotov

	Ozobot Bit	Ozobot Evo	LEGO WeDo 2.0	Micro:Bit	Botley
Absolútny pohyb			✓		
Relatívny pohyb	✓	✓			✓
Nastaviteľná dĺžka kroku	✓	✓			
Nastaviteľná rýchlosť	✓	✓	✓		
Nastaviteľný čas chodu motora			✓		
Konštantný krok					✓
Nekonečný pohyb			✓		

5 ZÁVER

V prvej fáze analýzy sme sa bližšie venovali piatim programovateľným hračkám, v ktorých vidíme edukačný potenciál pre žiakov na ZŠ. Analýzu sme zamerali na programovacie konštrukty v prostrediach, v ktorých sú roboty programovateľné, čo nám dáva bližší obraz o šírke programátorských zručností, ktoré žiaci vedia pri práci s danou hračkou získať. Ďalej sme analyzovali interakciu robotov so svetlom a zvukom, čo pridáva motivačný efekt a zároveň slúži na pochopenie práce senzorov. V neposlednom rade sme analyzovali aj spôsob, akým sa programuje pohyb robota, keďže to tiež určuje náročnosť a tým aj vhodnosť na použitie v jednotlivých ročníkoch. Ďalšie roboty a stavebnice plánujeme pridávať v nasledujúcich fázach analýzy. Už teraz však vidíme, že do takto vytvorených kategórií vieme zaradiť ďalšie robotické stavebnice ako Blue-bot a Bee-bot, LEGO Prime Spike, M-Bot, Dash-dot, Thymio, Sphero, Edison či Matalab robot.

Myslíme si, že naše výsledky môžu uľahčiť učiteľom výber stavebníc, ktoré majú pre svoju výučbu do školy kúpiť. Uvedomujeme si však aj slabé stránky našich výsledkov. Stavebnice sa veľmi rýchlo vyvíjajú, a to, čo platí dnes, nemusí pri stavebniciach platiť o pol roka. Už teraz napr. vieme o Botley 2.0, ktorý má viacero nových funkcionalít. Často sa stáva, že novšie verzie robotov nespolupracujú so staršími verziami softvéru, alebo sa úplne inak ovládajú, a preto naše tabuľky možno nebudú už o pár mesiacov správne. Druhá vec, ktorú nezohľadňujeme, ale je určite dôležitá pre učiteľa pri rozhodnutí, ktorú robotickú hračku využiť, je dostupnosť metodických materiálov. Vieme, že pre LEGO WeDo [14], Ozobot-y [15] aj Micro:bit-y [16] existuje viacero metodických materiálov aj v slovenskom jazyku, pre stavebnicu Botley sme zatiaľ našli len zahraničné [17, 18].

Veríme však, že takáto analýza bude nápomocná nie len pre učiteľov, ale aj didaktikov, ktorí sa venujú včeleňovaniu edukačnej

robotiky do nižších ročníkov základných škôl, a postupne vzniknú ďalšie metodické materiály.

POĎAKOVANIE

Na tomto mieste by sme chceli poďakovať projektu VEGA 1/0602/20, vďaka ktorému mohol byť realizovaný nákup analyzovaných stavebníc a publikované výsledky výskumu.

BIBLIOGRAFICKÉ ODKAZY

- [1] GAVORA, P. et al. *Elektronická učebnica pedagogického výskumu*. [online]. Bratislava : Univerzita Komenského, 2010. Dostupné na: <http://www.e-metodologia.fedu.uniba.sk/> ISBN 978-80-223-2951-4.
- [2] ŠVAŘÍČEK, R. ŠEĎOVÁ, K. et al. *Kvalitatívny výskum v pedagogických viedach*. Praha: Portál, 2014. 384 s. ISBN 978-80-262-0644-6.
- [3] ŠTÁTNY PEDAGOGICKÝ ÚSTAV. *Inovovaný Štátny vzdelávací program*. <https://www.statpedu.sk/svp/inovovany-statny-vzdelavaci-program/>
- [4] OZOBOT & EVOLLVE. Ozobot. <https://ozobot.com/>
- [5] LEARNING RESOURCES. Botley. <https://www.learningresources.com/media/botley/>
- [6] THE LEGO GROUP. LEGO Education WeDo 2.0 Core set. <https://education.lego.com/en-us/products/lego-education-wedo-2-0-core-set/45300#wedo-20>
- [7] MICRO:BIT EDUCATIONAL FOUNDATION. BBC Micro:bit. <https://microbit.org/>
- [8] LEE, P. T., LOW, C. W. Implementing a Computational Thinking Curriculum with Robotic Coding Activities through Non-formal Learning. In *CoolThink@ JC*, 2020. s. 150.
- [9] ETEOKLEOUS, N., NEOPHYTOU, R., KOLANI, E., & CHRISTODOULOU, C. The case of the Robotics Academy. In *21st Century Skills Developed through a Non-formal Educational Setting*. 2020, Frederick University. 10(1B), s. 162-170.
- [10] STOFFOVÁ, V., ZBORAN, M. Hravá forma stavby a programovania robotov na základnej škole. In *Trendy ve vzdelávaní*. 2018, s. 39.
- [11] BALOGH R., PETROVIČ P. Robot League – A Unique On-Line Robotics Competition. In Merdan M., Lepuschitz W., Koppensteiner G., Balogh R., Obdržálek D. (eds) *Robotics in Education*. RiE 2019. *Advances in Intelligent Systems and Computing*, vol 1023. Springer, Cham.
- [12] PANWAR, A., CHAUHAN A., ARYA K. Analyzing Learning Outcomes for a Massive Online Competition through a Project-Based Learning Engagement. In *IEEE Global Engineering Education Conference (EDUCON)*, Porto, Portugal, 2020, s. 1246-1251.
- [13] CHRISTOFOROU E.G. et al. Educational Robotics Competitions and Involved Methodological Aspects. In Merdan M., Lepuschitz W., Koppensteiner G., Balogh R., Obdržálek D. (eds) *Robotics in Education*. RiE 2019. *Advances in Intelligent Systems and Computing*, vol 1023. Springer, Cham. https://doi.org/10.1007/978-3-030-26945-6_27
- [14] VESELOVSKÁ, M., MAYEROVÁ, K. LEGO WeDo curriculum for lower secondary school. In *International Conference on Robotics and Education RiE*. 2017. s. 53-64. Springer, Cham.

- [15] UČÍME S ROBOTY. *Ozobot*.
<https://ucimesroboty.cz/index.php/category/ozobot/>
- [16] SPY O.Z. *Učíme s hardvérom*.
<https://www.ucimeshardverom.sk/>
- [17] VICTORIA K. *Our Hands On Review with the Botley Coding Robot*. [online] Teach Your Kids Code, 2018 [cit. 2021-01-31] <https://teachyourkidscode.com/botley-coding-robot/>
- [18] LEARNING RESOURCES. *Coding with Botley Printables*.
<https://blog.learningresources.com/coding-botley-printables/>

Dištančné vzdelávanie GNU/Linux zručností pomocou modulu Multimediálnej čítanky

Distance learning of GNU/Linux skills using a module of Multimedia Reader

Marek Nagy

Comenius University in Bratislava
Mlynská dolina F1
842 48 Bratislava
Slovakia

mnagy@ii.fmph.uniba.sk

ABSTRACT

The following article describes the implementation of online distance courses, which are aimed at acquiring skills in the text console of the GNU/Linux operating system. The courses are included in the bachelor's study of applied informatics at the university. They are taught using a server with a web application Multimedia Reader, which has been extended with specific needs. Course participants log in to a server using a web browser, where they communicate with each other, watch lectures and demonstrations of work in the terminal. They acquire practical skills on a student Linux server set up for teaching purposes. They launch the claAgent application in the console, which mediates communication with the web server. The application gradually presents tasks to students and automatically checks the correctness of the solution. The teacher on the web server monitors the contents of all students' terminals in real time and, if necessary, can guide the student via text messages written directly into the console.

Keywords

GNU/Linux. Instruction. Command line.

ABSTRAKT

V nasledujúcom príspevku je opísaná realizácia dištančných online kurzov, ktoré sú zamerané na získavanie zručností v textovej konzole operačného systému GNU/Linux. Kurzy sú zaradené v bakalárskom štúdiu aplikovanej informatiky na univerzite. Pri ich výučbe sa využíva server s webovou aplikáciou Multimediálna čítanka, ktorá bola rozšírená o špecifické potreby. Účastníci kurzu sa pomocou webového prehliadača prihlásia na server, kde vzájomne komunikujú, sledujú prednášky a ukážky práce v termináli. Praktické zručnosti nadobúdajú na študentskom linuxovom serveri zriadenom pre potreby výučby. V konzole si spustia aplikáciu claAgent, ktorá sprostredkúva komunikáciu s webovým serverom. Aplikácia študentom postupne predkladá úlohy a automaticky kontroluje správnosť riešenia. Učiteľ na webovom serveri sleduje obsahy terminálov všetkých študentov v reálnom čase a v prípade potreby vie študenta usmerniť cez textové správy zapísané priamo do konzoly.

Kľúčové slová

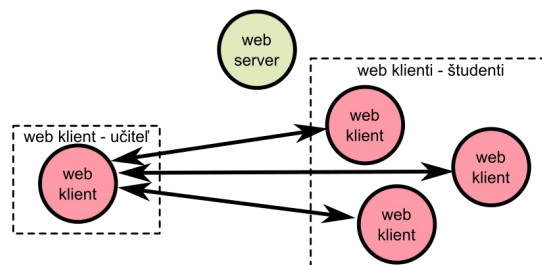
GNU/Linux. Vyučovanie. Príkazový riadok.

1 ÚVOD

Dištančné vzdelávanie v súčasnom období nabera veľký význam. V dôsledku protiepidemických opatrení v súvislosti so šírením ochorenia COVID-19 sa vyučovací proces presunul do online

priestoru (na internet). Nevyhlo sa to ani kurzom zručností s operačným systémom GNU/Linux, ktoré sa realizujú v bakalárskom študijnom programe na univerzite. Sú dobrovoľné (výberové) a prebiehajú v časovej dotácii 2 hodín týždenne. Kurz Linux1 zahŕňa základné zručnosti práce v príkazovom riadku [1]. Prebieha v letnom semestri a zasiahnutý bol prvou vlnou pandémie COVID-19. Pokračovaním kurzu v zimnom semestri je kurz Linux2, ktorý zahŕňa viac zručností s administráciou a konfiguráciou operačného systému. Používa sa pri tom príkazový riadok.

V predchádzajúcom prezenčnom období sa pri kurzoch využívalo prostredie Moodle [2], kde sa zverejňovali prezentácie a materiály. Cvičenia vtedy prebiehali interaktívne v učebni pri počítačoch a študentom pomáhala s úlohami skupina cvičiacich. Po uzavretí univerzity a prechode na dištančné vzdelávanie bolo potrebné realizovať nový prístup hlavne k praktickým cvičeniam. Skúsenosti z prvej vlny zatvorených škôl ukázali, že jednoduché odprezentovanie pomocou videokonferencie a následné samostatné riešenie úloh doma bez pomoci nie je postačujúce. Z technického hľadiska by individuálne asynchrónne riešenie úloh z cvičení nebol problém. Prístup ku GNU/Linux terminálu bol vytvorený na študentskom linuxovom serveri. Problémom však bola interakcia. Pomerne veľká časť študentov nedokázala napredovať samostatne. Pri výskyte drobného problému sa komunikovalo príliš ťažkopádne cez textový/audio chat. Riešenie cez súbežné zdieľanie väčšieho množstva obrazoviek študentov z pohľadu učiteľa a jeho technických parametrov pripojenia k internetu nebolo vhodné.



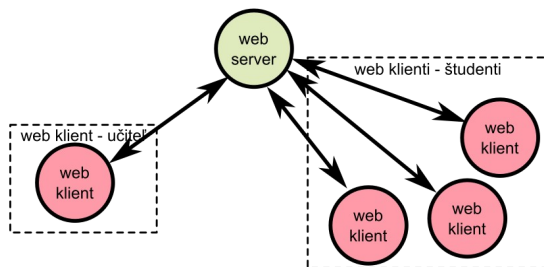
Obrázok 1: Pomocou WebRTC môžu byť študenti napojení priamo na webový prehliadač učiteľa. Tak je medzi nimi vytvorená priama komunikácia.

Priebeh videokomunikácie medzi klientmi webového servera sa momentálne môže riešiť tak, ako je znázornené na obrázku 1. Väčšina webových prehliadačov podporuje WebRTC rozhranie [3]. Vďaka nemu možno vytvárať priame komunikačné spojenia medzi klientmi, čím sa redukujú nároky na server, ktorý pomáha

iba pri nadviazaní spojenia. Takto sa znižuje aj časové oneskorenia v komunikácii klientov na nutné minimum. Samotné WebRTC umožňuje vytvárať toky multimediálnych dát, či už z webovej kamery, mikrofónu, záznamu okna aplikácie alebo celej pracovnej plochy. Tieto toky dát možno v súčasnej dobe smerovať, ako bolo spomínané vyššie, priamo od klienta ku klientovi. Z pohľadu vývojárov webových portálov sa to ľahko implementuje a vďačne využíva.

Problém však nastáva vtedy, keď sa realizuje typický scenár univerzitnej prednášky. Učiteľ posielal tok dát zo svojej webovej kamery a zdieľaného okna aplikácie klientom – študentom, ktorých možno rátať v stovkách. Keď videoprenos vyžaduje od každého pripojeného študenta kapacitu pripojenia cca 500 kbps, celkové nároky na internetové pripojenie učiteľa by sa vyšplhali ku 100 Mbps pre cca 200 pripojených študentov. Ak však učiteľ využíva bežné mobilné pripojenie s priepustnosťou rádovo 1 Mbps, je takýto scenár prednášky technicky nerealizovateľný. Rovnako treba spomenúť, že učiteľ potrebuje aj pomerne výkonný počítač na manažment takého objemu dát. Obdobnú situáciu možno identifikovať aj pri realizovaní univerzitných praktických cvičení. Aby mohol učiteľ sledovať, čo študenti robia, každý študent by mal nazdieľať okno precvičovanej aplikácie. Tým by posielali učiteľovi enormné množstvo dát, čo tiež kladie nároky na kapacitu pripojenia.

Spomínané „kapacitné“ problémy sa často riešia premysleným znižovaním rozmerov a kvality prijímaných tokov video a audio dát. Veď v konečnom dôsledku si klient vie videá umiestniť len na konečnú pracovnú plochu počítača. Rizikom degradácie kvality videoprenosu je znížená čitateľnosť prípadných prenášaných textov napríklad v textovom termináli. Lepším riešením je návrat späť ku schéme komunikácie prostredníctvom servera (viď obrázok 2). Je to starší prístup a jeho hlavná výhoda je v rovnocennej požiadavke na technické nároky pripojenia učiteľa a študentov. Nevýhodou je vysoké zaťaženie servera, čo nevyhovuje hlavne veľkým poskytovateľom služieb. V praxi však často možno sledovať kombináciu oboch prístupov ku komunikácii.



Obrázok 2: Komunikácia môže prebiehať aj skrze server.

Ak sa využíva centralizovaný prístup cez server (viď obrázok 2), môžu byť, napríklad, videá od študentov menšieho rozmeru a na serveri sa „poskladajú“ do jedného videa bežného rozlíšenia. To sa následne odosiela všetkým klientom. Pri zvuku je to obdobné. Prichádzajúce nahrávky z mikrofónov sa na serveri zložia a výsledný zvuk je rozposlaný klientom. Najviac úspory prenosu možno dosiahnuť pri zdieľanom prezentovaní statických dokumentov (napr. pdf). Dokument sa celý presunie na server a odtiaľ ku klientom. Potom stačí už len inštruovať, ktorá strana dokumentu sa má zobrazovať.

Veľkým problémom aj tak zostáva situácia, keď by počas cvičení mali študenti zdieľať okno terminálu. Aby bol text čitateľný, nie je možné zredukovať rozlíšenie a aj dynamika „obrazu“ nedovolí

redukciu prenášaných dát. A keďže všetky zdieľané okná musia „skončiť“ u učiteľa, v konečnom dôsledku sa prenos nakumuluje a je konfrontovaný s priepustnosťou internetového spojenia učiteľa.

Nakoľko sa pre potreby cvičení začal riešiť synchronný interaktívny prístup, prostredie Moodle [2] už prestalo vyhovovať. Iné (komerčné) platformy sa až príliš sústreďujú iba na videokonferenčnú interaktivitu a preto padla voľba na Multimediálnu čítanku [4]. Má už 15-ročnú tradíciu a je úspešne využívaná na účel počítačom podporovanej výučby na primárnom stupni vzdelávania. Naviac sú v nej zahrnuté skúsenosti s manažmentom prenosu audio dát, ktoré vznikajú a spracovávajú sa pri nahrávaní čítajúcich detí. Čítanka bola rozšírená o špeciálny modul Command Line Assistant (CLA), pre potreby cvičení oboch spomínaných kurzov Linux1 a Linux2.

2 MULTIMEDIÁLNA ČÍTANKA

Webová aplikácia Multimediálna čítanka [4] vznikla ešte v roku 2005. Postupne sa vyvíjala do podoby webového portálu a slúži pre učiteľov a deti primárneho vzdelávania. V poslednom období bola aplikácia štruktúrovaná do rámcovej podoby, kde možno identifikovať moduly, ktoré možno pridávať a vylepšovať. Typickým modulom je správa používateľov a ich zoskupovanie. Na skupinu sa možno v aktuálnej verzii pozeráť ako na triedu žiakov, alebo i ako na účastníkov kurzu. Pre skupinu môže učiteľ plánovať a zverejňovať aktivity, ktorých sa účastníci synchronne alebo asynchrónne zúčastňujú. Činnosť žiakov v aktivite je možné hodnotiť bodmi, ktoré sa zobrazujú v tabuľke. Nové aktivity ako moduly možno do čítanky podľa potreby dopĺňať a vznikajú aj formou univerzitných bakalárskych prác.

Pôvodne boli aktivity smerované na nácvik čítania detí v primárnom vzdelávaní. Existuje k nim prepojenie na vlastný rozpoznávač reči, ktorý sa využíva na kontrolu čítania [5]. Čítanka sa v čase pred pandémiou využívala priamo pri vyučovaní v počítačovej učebni a tak možno v tej situácii skôr hovoriť o počítačovo asistovanej výučbe. Učiteľovi pomáhala zefektívniť a spestriť vyučovanie. Už v tých časoch boli pokusy dištančného využitia formou domácich úloh na nácvik hlasného čítania. Narážalo to však na problémy s prístupom detí k domácejmu počítaču.

Vlastné webové prostredie čítanky s bohatou množinou nástrojov podnietilo myšlienku využiť ho aj pri výučbe na univerzite. Úvahy boli smerované k tomu, ako pomôcť študentom priamo pri výučbe v učebni. Najprv bola do čítanky vytvorená aktivita, ktorá podporovala realizáciu prednášky. Jednoduché zdieľanie pdf prezentácie a ukážok z terminálu pomohlo úplne zredukovať potrebu dataprojektora, ktorý sa predtým používal. Študenti všetko sledovali na svojom počítači v učebni. V čase prvej vlny pandémie táto aktivita veľmi pomohla. Bolo však nutné urýchlene realizovať rozšírenie čítanky o zvukový komunikačný modul, aby si študenti mohli dištančnú prezentáciu aj vypočuť. Súbežne s tým bol pridaný aj jednoduchý textový chat. Momentálne nechýba ani možnosť zazdieľať študentom vybrané okno aplikácie učiteľa. Pre potreby odovzdávania úloh sa vyriešilo aj nahrávanie a manažovanie súborov. V podstate sa jedná o základnú funkcionálnosť, ktorú ponúkajú i iné (komerčné) videokonferenčné systémy.

Ešte pri výučbe čítania detí v učebni sa veľmi osvedčil odpovedací modul. Učiteľ povedal otázku a deti odosieli odpovede, ktoré učiteľ vyhodnotil. Preto bol tento modul transformovaný na dištančné využitie. Je univerzálny a tak je možné ho využiť aj v aktivite prednášky pre študentov. Slúži

na udržiavanie pozornosti. Učiteľ sa kedykoľvek niečo spýta a študenti odosielaajú odpoveď. Je to lepšia možnosť ako použiť textový chat, kde sa komunikácia pri väčšom množstve študentov mieša a stráca. Pri separátnom odpovedacom module učiteľ prehľadne vidí kto, či a ako odpovedal. Ľahšie a rýchlejšie sa mu tak hodnotí správnosť odpovedí. V rozhodovaní mu pomôžu aj kumulatívne súčty.

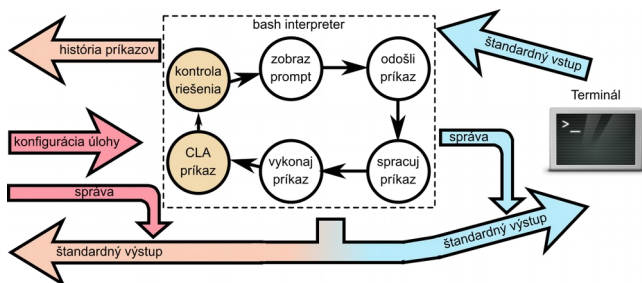
Počas druhej vlny pandémie COVID-19 bola aktivita čítanky rozšírená aj na podporu realizácie synchrónnych cvičení kurzu, ktoré sú zamerané na zručnosti práce v textovej konzole (modul CLA). Pomocou externej aplikácie claAgent môže učiteľ v reálnom čase sledovať, ako študenti pracujú v termináli a na základe toho zvoliť intervenciu. Neprenášajú sa obrazy okien terminálov, ale iba ich textový obsah, čím sa prenosy dát redukovujú.

3 COMMAND LINE ASSISTANT

Ako bolo spomínané, Multimediálnu čítanku je možné rozširovať o ďalšie moduly a aktivity. Väčšina administratívnych vecí, ako napríklad manažment používateľov, je už v nej vyriešená a tak nie je potrebné sa na ne sústrediť. Takto vznikol aj modul CLA, ktorý pomáha pri výučbe GNU/Linux zručností študentov na univerzite. CLA modul, okrem webovej implementácie, má aj podpornú externú aplikáciu claAgent. Aplikácia slúži na sprostredkovanie komunikácie medzi serverom a terminálom. Možno hovoriť, že je to taká zjednodušená verzia webového klienta, ktorú si študent spustí v textovej konzole.

3.1 Aplikácia claAgent

Získavanie informácií z terminálov študentov je realizované pomocou externej aplikácie claAgent. Ide o binárnu aplikáciu, ktorá je prekompilovaná podľa cieľovej architektúry. Spustí sa priamo v termináli. Nakoľko beží s právami používateľa, je lepšie na riešenie úlohy využiť cvičný študentský server. V prípade, ak sa precvičujú administrátorské zručnosti, osvedčilo sa spustiť virtuálny počítač, prihlásiť sa naň ako administrátor (root) a následne spustiť claAgent. Na obrázku 3 je znázornená schematická architektúra claAgent.



Obrázok 3: Architektúra aplikácie claAgent. Do bash cyklu je pridaná kontrola riešenia úlohy a vykonanie špecifických príkazov claAgent. Zoznam ukončených príkazov sa odosiela na server. Štandardný výstup je duplikovaný a v reálnom čase sa odosiela na server.

Do základného vykonávacieho bash cyklu [1] je včlenená kontrola požadovaného stavu riešenia. Kontrola je v podstate bash skript, ktorý sa spustí a podľa úspešnosti (exit kódu) signalizuje, či riešenie vyhovuje alebo nie. Textové zadanie úlohy, testovací skript a pomocné súbory prídu do claAgentu zo servera (označené na obrázku 3 ako konfigurácia úlohy). Text úlohy si môže študent kedykoľvek dať vypísať do terminálu. Úspešnosť riešenia sa

testuje po každom odoslanom príkaze v rámci prechodu vykonávacím bash cyklom.

Aby server vedel identifikovať používateľa, je potrebné sa do claAgentu prihlásiť. Pre potreby manažovania je v bash cykle zaradené rozpoznanie interných príkazov claAgentu. Napríklad vylisťovanie všetkých zadaní, z ktorých si študent vyberie jedno a následne mu server z neho posíla jednotlivé úlohy na riešenie. Momentálne je zadanie realizované ako usporiadaná sekvencia úloh, ktoré na seba nadväzujú.

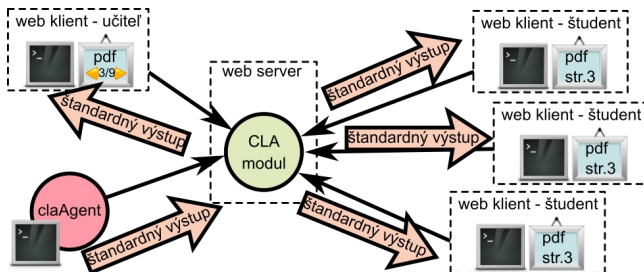
História spúšťaných príkazov v termináli sa zasiela na server a tak je možné neskôr realizovať prípadnú kontrolu a analýzu postupu študenta. Aby bolo možné sledovať aktuálnu činnosť v termináli v reálnom čase, znakový prúd smerujúci na zobrazenie do terminálu sa duplikuje a súčasne sa odosiela aj na server. Takto, v podstate, vidieť presný obraz terminálu študenta so zachovaním nízkych nárokov na prenosovú kapacitu internetového pripojenia. Okrem toho je tento spôsob prínosný aj pre učiteľa, nakoľko si vie v každom okamihu prelistovať celý záznam práce študenta. Rýchlo si tak vie pripomenúť, čo práve študent robil a práve robí. Kompetentnejšie a rýchlejšie vie následne so študentom komunikovať.

Veľmi často sa stáva, že študenti riešia úlohy zo zadania v termináli pomocou claAgentu a neprihlásia sa na webový server. Takto sa však zamedzí komunikácii medzi učiteľom a študentom. Kvôli tomu bolo do CLA modulu pridané rozšírenie, že učiteľ má možnosť poslať textovú správu, ktorá sa priamo zobrazí v termináli. Študenti písali učiteľovi zo začiatku odpovede priamo do príkazového riadku, čo však bolo rušivé, lebo bash samozrejme vypisoval chyby a miešalo sa to s riešením úlohy. Na tento účel sa zaviedol nový príkaz claAgentu, ktorý umožnil študentom odoslať vlastnú správu, ktorá sa odoslala aj na server. Týmto sa vytvorila jednoduchá chatovacia funkcionálnosť priamo v textovej konzole. Sekvencia správ komunikácie je zachytávaná na serveri a tak učiteľ dostane avízo o správe, aj keď nemá zobrazený obsah terminálu študenta. Význam to má najmä pri monitorovaní väčšieho počtu pripojených študentov.

3.2 Modul CLA

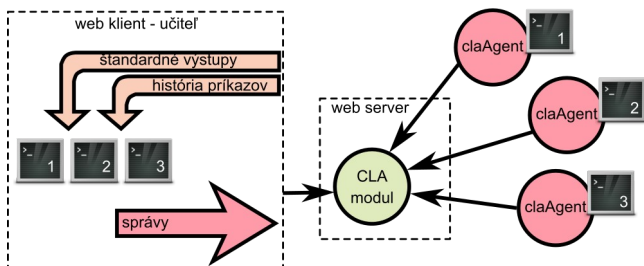
Rozšírenie webovej aplikácie je realizované modulom CLA. Najskôr sa v rámci modulu zrealizovalo zdieľanie pdf dokumentu na účely prezentácie slajdov. Učiteľ prepína slajdy, ktoré sa zobrazujú všetkým prihláseným študentom. Táto funkcionálnosť bola riešená ešte pred spustením dištančného vyučovania, aby prednáška aj následné cvičenie v počítačovej učebni mohlo prebiehať spoločne. Prezentovanie cez dataprojektor bolo nepraktické a študenti ani príliš nevideli na slajdy, hlavne v prípade, ak je učebňa rozsiahlejšia. Jednoduché zdieľanie pdf dokumentu problém vyriešilo. Podobne bolo potrebné riešiť aj ukážky v termináli. Keď sa premietal terminál cez dataprojektor, bola slabá čitateľnosť. V tomto čase vznikol aj prototyp claAgentu, ktorý spočiatku slúžil iba učiteľovi na prezentáciu. Učiteľ prezentuje „živé“ príklady zručnosti zo svojho terminálu so spusteným claAgentom. Informácie z claAgentu sa prenášajú na server a znakový prúd je presmerovaný do prezentačných terminálov, ktoré si pootvárajú študenti počas prednášky. Schematicky to možno vidieť na obrázku 4. To, že informácia sa do prezentačného terminálu prenáša v textovej podobe, umožňuje aj jednoduché kopírovanie ukážkových príkazov. Neskôr sa claAgent vylepšil aj o funkcionálnosť autonómnej kontroly úloh. Samotný výklad je prenášaný vlastným audio kanálom, ktorý je vylepšovaný a používaný ako súčasť celého servera čítanky. Pri vývoji bolo potrebné brať ohľad aj na pôvodné poslanie

čítanky a vhodne prepojiť audiokomunikáciu s nahrávaním a kontrolovaním správnosti čítania.



Obrázok 4: Zdieľanie prezentačných slajdov a „živých“ ukážok z terminálu učiteľa počas prednášky. Výklad je prenášaný audiokomunikačným modulom servera.

Na obrázku 5 možno vidieť orientačnú architektúru priebehu synchronného cvičenia. Študenti spustia claAgenta vo svojich termináloch. Učiteľ si vie zobrazit' zoznam pripojených claAgentov. V záznamoch vidí posledne spustený príkaz a posledne odoslanú správu študenta. V prípade potreby si zobrazí celý obsah študentského terminálu. Pri menšom počte študentov nie je problém zobrazit' si všetky terminály, avšak, zo skúseností, je ideálne mať ich na notebooku otvorených odhadom do sedem. Pri väčšom počte je vhodné pripojiť externý duálny monitor. Vyskúšala sa aj možnosť, keď synchronné cvičenie realizujú viacerí učitelia. V tom prípade sa však niekedy stávalo, že viacerí učitelia píšu správu rovnakému študentovi, čo je z pohľadu priebehu cvičení neefektívne. Do zoznamu všetkých claAgentov bola preto pripojená aj informácia o tom, ktorý učiteľ má otvorené, ktoré terminály a ktorý z nich je aktuálne vybraný t.j. má „pozornosť“ učiteľa. Technicky je možné správu poslať iba do takto vybraného terminálu. Vďaka tomuto mechanizmu sa učitelia mimovoľne nezokupujú iba pri jednom študentovi.



Obrázok 5: Základná architektúra priebehu synchronného cvičenia. Webový prehliadač učiteľa zobrazuje v reálnom čase obsahy terminálov a umožňuje do nich odosielať študentom správy.

Pri zmene „pozornosti“ učiteľa sa osvedčila možnosť, rýchlo si pozrieť celý záznam práce študenta v termináli. Je to možné, nakoľko znakový prúd (obsah terminálu) je dočasne uchovávaný. Odpadne tak zbytočná komunikácia, aby sa učiteľ dostal „do obrazu“.

Na obrázku 6 je ukážka monitorovacieho okna, ktoré sa zobrazuje učiteľovi a jeho pomocníkom. Základné delenie študentov je na tých, ktorí majú otvorený terminál t.j. spusteného claAgentu a ostatných, ktorí sú odpojení. Ku všetkým je možné zobrazit' si históriu spúšťaných príkazov. Terminál momentálne pripojených študentov možno sledovať v reálnom čase a komunikovať s nimi cez posielanie správ. Okno monitora je členené na stĺpce. V prvom je meno študenta, druhý zobrazuje doteraz najlepšie

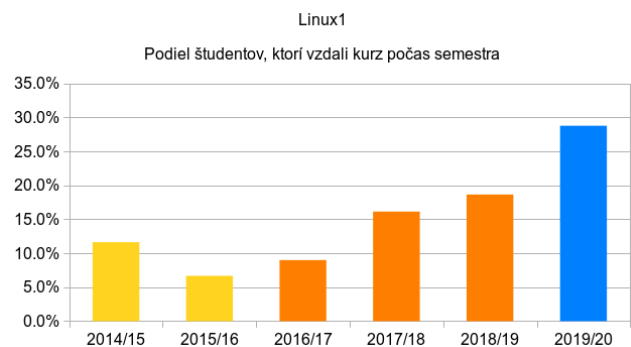
dosiahnuté skóre zo všetkých pokusov. Nasleduje stĺpec, kde vidieť, ktorú úlohu práve študent rieši. Farebné štvorce signalizujú, že niekto z cvičiacich alebo učiteľ má otvorený terminál, kde pozoruje a komunikuje so študentom. Cez titulok je dostupná informácia aj kto konkrétne. Ďalej nasleduje stĺpec s časom, ktorý odpočítava posledne odoslanú správu. Správu vidieť v ďalšom stĺpci. Rozlišuje sa, či správa prišla od študenta, alebo od učiteľa. Farebne to signalizuje aj časovač. Podľa tohto údaju sa možno rozhodnúť, kde začať intervenciu. Dynamiku napredovania študenta ilustrujú posledné dva stĺpce, kde je vypísaný posledne spustený príkaz v termináli aj s odpočítaním uplynutého času. Učiteľ sa môže rozhodnúť pozrieť na študenta, ktorý „stagnuje“.



Obrázok 6: Ukážka súhrnného monitorovacieho okna, ktoré sleduje učiteľ počas synchronných online cvičení.

4 PRIEBEH VÝUČBY

Počas prvej vlny pandémie COVID-19 v akademickom roku 2019/20 sa 13-týždňový letný semester začal ešte v normálnom prezenčnom režime. Na dištančnú formu sa prešlo až v 4. týždni. V tomto období modul CLA existoval iba v obmedzenej podobe. Využíval sa hlavne na prezentáciu prednášky a ukážky práce v textovom termináli ako náhrada prezentovania cez dataprojektor. Po zrušení prezenčného vyučovania sa do aplikácie čítanky urýchlene doplnila možnosť audiokomunikácie, aby nebolo nutné využívať viaceré platformy a pohodlne plnohodnotne odprezentovať prednášku. Audio modul sa priebežne vylepšoval. Cvičenia však v tom čase zostávali ako „domáca“ úloha.



Obrázok 7: Počty študentov, ktorí vzdali kurz Linux1 z letného semestra. Automaticky kontrolované úlohy začínajú od roka 2016/17 a posledný rok prebiehal dištančne.

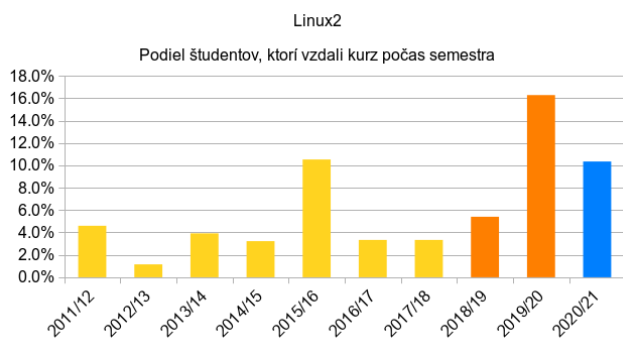
Riešenie úlohy na cvičeniach kurzov pozostáva z vykonania sledu príkazov v príkazovom riadku. Už v minulosti sa proces kontroly správnosti riešenia zautomatizoval [6] a tak počas dištančného vzdelávania sa zdalo, že stačí, ak študenti prejdú cez úlohu v termináli samostatne len za pomoci tejto automatickej „asistencie“. Tento prístup k cvičeniam bol však bez intervencie učiteľa pre väčšinu študentov nevládnuteľný. Niektoré problémy sa podarilo riešiť emailovou komunikáciou, audiokomunikáciou alebo textovým chatom. Zvolený prístup a sťažaná komunikácia sa prejavila aj na redukcii počtu študentov kurzu. Svedčí o tom

29 % študentov, ktorí kurz v priebehu semestra vzdali (viď graf na obrázku 7). Do počtu sú rátnati študenti, ktorí začali, ale nedokončili kurz. Neuvažujú sa študenti, ktorí mali kurz zapísaný vo svojom študijnom pláne, ale ho vôbec nenavštevovali.

Automatická kontrola úloh [6] bola experimentálne zavedená od roku 2016/17, ale všetky cvičenia pokrývala až od roku 2017/18. Z grafu na obrázku 7 je zrejmé, že tento prístup vykazoval potrebu zvýšenej komunikácie medzi učiteľom a študentom aj počas prezenčného riešenia úlohy na cvičení v počítačovej učebni. Vtedy závisel od počtu študentov v učebni a od počtu cvičiacich, ktorí sa im venovali. Automatická kontrola riešenia úloh je veľmi nekompromisná a aj drobný nedostatok môže zablockovať riešenie nasledujúcich (graduujúcich) úloh, čo sa prejaví na frustrácii študenta. Toto sa ukázalo ešte viac počas obmedzenej intervencie učiteľa v priebehu prvej vlny dištančného vzdelávania. Na obrázku 7 možno vidieť vysoký nárast „frustrovaných“ študentov v poslednom roku.

Počas druhej vlny pandémie COVID-19 v akademickom roku 2020/21 sa v zimnom semestri začal využívať modul CLA v plnej podobe aj na synchronných cvičeniach. Prvé dva týždne boli ešte v prezenčnej podobe, čo sa využilo na oboznámenie sa s novým prostredím a claAgentom. Následne už riešenie úloh prebiehalo dištančne prostredníctvom claAgentu a tak mohla byť intervencia synchronná. Aby bolo možné sústrediť sa najmä na ťažkosti pri riešení, zadanie úloh bolo sprístupnené dopredu na „domácu“ samostatnú úlohu a následne prebehlo synchronne dištančné cvičenie s „menej skúsenými“ študentmi. V zimnom semestri prebieha pokračovanie kurzu (Linux2), ktoré je viac zamerané na správu systému. Preto mali študenti na študentskom serveri spustený svoj virtuálny stroj, kde boli prihlásení ako administrátori.

Do tohto kurzu prenikli automaticky kontrolované úlohy [6] až neskôr v roku 2018/19. Naplno sa však využívali od roku 2019/20. Na obrázku 8 možno vidieť graf študentov, ktorí vzdali kurz. Zahnuté je širšie časové obdobie, ku ktorému sa zachovali dáta. Podobne ako pri kurze Linux1 možno vidieť, že automatická kontrola úloh bez dobrej intervencie viedla v roku 2019/20 k zvýšenému predčasnému odchodu študentov. V poslednom akademickom roku sa však tento podiel študentov podarilo zredukovať i napriek tomu, že cvičenia prebiehali dištančne. To dáva predpoklad, že prístup pomocou claAgentu umožňuje efektívnejšiu intervenciu učiteľa, ako je počas prezenčných cvičení v počítačovej učebni.



Obrázok 8: Počty študentov, ktorí vzdali kurz Linux2 v priebehu zimného semestra. Automaticky kontrolované úlohy začínajú rokom 2018/19. Posledný rok je dištančnou formou a s novým interaktívnym prístupom cez claAgentu.

5 ZÁVER

Dištančný prístup k realizácii nadväzujúcich kurzov Linux1 a Linux2 dopomohol ku zásadnej zmene v pôvodnom prístupe riešenia a automatickej kontrole úloh na cvičeniach v počítačovej učebni [6]. Už počas prezenčných cvičení dochádzalo vplyvom automatického prístupu ku kontrole úloh k nárastu úbytku študentov v priebehu semestra. Možno to pripísať k nedostatočnej intervencii učiteľa počas riešenia úloh. Spomalenie úbytku nastalo po zvýšení počtu vypomáhajúcich cvičiacich z radov starších študentov. No po presune do online priestoru úbytok predstavoval až rekordných 29 % študentov v kurze Linux1 (viď obrázok 7).

Nový prístup k dištančným synchronným cvičeniam prostredníctvom realizovaného claAgentu dopomohol zvýšiť a zefektívniť intervenciu učiteľa, čo prispelo k lepšiemu napredovaniu študentov pri riešení úloh. Novou metódou je možné včas identifikovať situácie vedúce ku frustrácii študentov a aktívne zasiahnuť už v zárodku. Učiteľ môže sledovať v podstate všetky terminály študentov, čo pri klasickom prezenčnom prístupe v učebni nie je možné. Tým, že je k dispozícii aj história práce študentov, je možné kompetentnejšie reagovať na situáciu pri presúvaní pozornosti a intervencie zo študenta na študenta. V prezenčnom prístupe v učebni to bez vyrušenia študenta z práce nešlo. Zároveň sa podarilo odstrániť fenomén „cudzích rúk na klávesnici“, ktorý bolo možné pozorovať na cvičeniach v učebni. Niektorí pomáhajúci cvičiaci mali tendenciu úkon v termináli vykonať namiesto študenta, čo potiera najnovšie pedagogické poznatky. Samotnému študentovi to nepomáha a svedčí to skôr o strate trpezlivosti cvičiaceho. Aplikácia claAgent umožňuje iba komunikáciu. Spúšťanie príkazov je len na samotnom študentovi, čím sa pokúšenie riešiť úlohu namiesto študenta eliminuje. Modul CLA sa snaží riešiť aj problém neefektívneho zoskupovania cvičiacich pri jednom študentovi. Monitorovaním práce „dištančných“ cvičiacich je možné predchádzať „konfliktom“. Realizovaný nový prístup k cvičeniam pomohol zredukovať úbytok študentov kurzu Linux2 na 10 % (viď obrázok 8).

POĎAKOVANIE

Rád by som poďakoval všetkým študentom a kolegom za trpezlivosť a pomoc.

BIBLIOGRAFICKÉ ODKAZY

- [1] Bash - GNU Project - Free Software Foundation [online]. [cit. 2021-02-21]. Dostupné na: <https://www.gnu.org/software/bash/>
- [2] Moodle - Open-source learning platform | Moodle.org [online]. [cit. 2021-02-21]. Dostupné na: <https://moodle.org/>
- [3] WebRTC : Real-time communication for the web [online]. [cit. 2021-02-21]. Dostupné na: <https://webrtc.org/>
- [4] NAGY, M. Multimediálna čítanka [Multimedia Reader] [online]. [cit. 2021-02-21]. Dostupné na: <https://www.mmcitanka.sk>
- [5] NAGY, M. Umelá inteligencia pomáha deťom s osvojovaním čítania. In *Kognícia a umelý život 2019 : zborník*. Bratislava : Univerzita Komenského v Bratislave, 2019. ISBN 978-80-223-4720-4, s.79-84.
- [6] NAGY, M. Using a text adventure game approach to train GNU/Linux command line skills. In *DidInfo 2019 : zborník*. Banská Bystrica : Univerzita Mateja Bela, Fakulta prírodných vied, 2019. ISBN 978-80-557-1533-9, s. 100-103.

Vybrané materiály využívané pri výučbe programovania na ZŠ Spojová

Selected Materials Used for Teaching of Programming On ZŠ Spojová

Mgr. Pavol Polonec

Základná škola, Spojová 14, Banská Bystrica
Žabokreky 374
038 40 Žabokreky
Slovenská republika
polonec@zsspojbb.sk

Mgr. Soňa Úradníková

Základná škola, Spojová 14, Banská Bystrica
Sitnianska 6599/33
974 11 Banská Bystrica
Slovenská republika
uradnikova@zsspojbb.sk

ABSTRACT

Teaching computer science in Slovakia is difficult in regards of materials and schoolbooks available. Therefore, computer science teachers need to create their own materials for teaching. Cards for unassisted learning are an example of such material.

Cards we made are short guides for practicing basic programming skills in various environments, such as Scratch, LEGO Classroom for LEGO robots, and Probots. Our cards serve as an introduction to programming languages and induce students to solve various programming problems (for example to create a simple game in Scratch Environment). Working with these cards develops logical thinking, develop reading literacy and improve students' ability to work in a team.

Keywords

Computer Science, Programming, Teaching Material. Primary and Secondary School

ABSTRAKT

Vyučovanie informatiky je v mnohých prípadoch náročné na prípravu, nakoľko sa učiteľ nemôže oprieť o učebnice, ktoré na trhu chýbajú. Preto sa každý učiteľ informatiky snaží vytvoriť si vlastné portfólio materiálov a pomôcok na vyučovanie. Jednou z pomôcok môžu byť aj kartičky, ktoré používajú žiaci pri samostatnej práci. Kartičky, ktoré sme vytvorili obsahujú návody na prácu v Scratchi, prácu s Probotom a programovanie LEGO robotov v prostredí LEGO Classroom. Tieto kartičky predstavujú žiakom prostredie, v ktorom programujú alebo ich vedú postupne k tomu, aby vyriešili zadanie a naprogramovali napríklad jednoduchú hru v Scratchi. Prácou s týmito kartičkami žiaci rozvíjajú svoje logické myslenie, podporujú čitateľskú gramotnosť a zlepšujú sa pri práci v skupinách.

Kľúčové slová

Informatika. Programovanie. Vyučovacia pomôcka. Základná škola.

1 ÚVOD

Algoritmické myslenie je neodlučiteľnou súčasťou práce každého programátora. Nie je však potrebné a užitočné len v praxi programátorov či informatikov – takéto myslenie určite pomôže každému pri riešení bežných úloh či problémov.

Algoritmus je spôsob riešenia problému pomocou presne zadanej postupnosti krokov. Pod algoritmickým myslením rozumieme taký spôsob myslenia a uvažovania, ktorý vedie k tvorbe algoritmu. Tento spôsob uvažovania je podmienený schopnosťou:

- analyzovať problém,
- presne určiť problém,
- identifikovať základné kroky, ktoré vedú k riešeniu problému,
- vytvoriť správny algoritmus na základe týchto krokov,
- zobrať do úvahy všetky okolnosti, ktoré môžu ovplyvniť riešenie problému,
- vylepšiť fungovanie algoritmu.

Pre správne algoritmické uvažovanie je, prirodzene, veľmi potrebné samostatné a tvorivé myslenie [1].

Ako teda učiť algoritmické myslenie? Ide o neľahkú úlohu rovnako, ako pri učení kreativity. V praxi je potrebné riešiť čo najväčšie množstvo problémov. Zároveň žiakom veľmi pomáha vhodná vizualizácia problému [1]. Najmä na základnej škole musíme opatrne vyberať problémy, ktoré sú pre žiakov zrozumiteľné a motivujúce. Základom dobre zhotoveného materiálu je vhodný výber problému.

2 PRÍKLADY

Cieľom tohoto článku je predstaviť našu prácu a inšpirovať kolegov - či už našimi nápadiami alebo k tvorbe vlastných materiálov. Iste si všimnete, že aj kvalita grafického spracovania od materiálu k materiálu stúpa. Preto je evidentné, že pedagóg nepotrebuje na tvorbu materiálu veľkú nápaditosť či talent. Je to zručnosť, ktorá sa dá naučiť.

2.1 PROBOT – PROGRAMOVANIE ÁUT

„Pro-Bot v tvare autíčka... vie kresliť čiary po podložke, prostredníctvom pera, uchyteneho do zabudovaného držáka, a má aj LCD displej“ [2, s. 106]. My na prvom stupni nevyužívame funkciu kreslenia iba pohybu po danej trase. Fotografia robota a jeho príslušenstva sa nachádza na Obrázku 1.


Zatiaľ sme pracovali len s tlačidlami vpred, vzad, vpravo a vľavo so žiakmi tretieho a štvrtého ročníka. Každý z týchto príkazov má prednastavenú hodnotu. Príkazy *vpred* a *vzad* auto posunú o približne 30 cm, *vpravo* a *vľavo* ho otočia o 90 stupňov. Okrem toho sa dajú upraviť parametrom, a to tak, že za príkaz napíšeme číslo.


V učebni informatiky sa nachádza stôl s rozmermi približne 200 x 400 cm. Na tomto stole vytvoríme cestu – bludisko pomocou ústrižkov kartónu s jasne označeným štartom a koncom. Úlohou žiakov je vo dvojici alebo trojici naprogramovať auto tak, aby prešlo zo štartu na cieľ.





Obrázok 1: Probot (obrázok autorov)


PROGRAMOVANIE S PROBOTOM


 Šípkami naprogramuj kam má auto ísť

Číslo za príkazom určí ako ďaleko auto pôjde alebo o koľko sa otočí 

 Polož auto na štart a program spusti tlačidlom GO

Auto musí prísť do cieľa 

 Nechcený príkaz vmaž tlačidlom Clear

Autíčko programujeme mimo dráhy. Pri dráhe trpezlivo čakám v rade. 

Obrázok 2: Učebná pomôcka pre prácu s Probotom

Táto aktivita rozvíja algoritmické myslenie a priestorovú predstavivosť. Keďže skupín máme 5 až 6 a na bludisko sa zmesť

len jedno auto, žiaci musia trpezlivo počkať, kým sa dostanú na rad. Aby si navzájom nezavadzali, autá musia programovať mimo stola – na zemi. Práve toto rozvíja priestorovú orientáciu, pri čakaní na stôl sa učia trpezlivosti a rešpektu voči druhým spolužiakom.

Počas úvodnej hodine je bludisko veľmi jednoduché – s pravými uhlami a vzdialenosťami v násobkoch 30 cm, aby žiaci nemuseli používať parametre. Pri nasledujúcich hodinách skladáme zložitejšie bludiská a žiakom iba ukážeme, ako majú parameter zadať. Ako funguje, teda o koľko sa auto otočí alebo posunie, musia zistiť sami.

Pre túto aktivitu sme vytvorili plagát, v ktorom sú zhrnuté základné zásady programovania út (pozri Obrázok 2).

2.2 SCRATCH KARTY

Scratch¹ je detský programovací jazyk s grafickým rozhraním, vyvíjaný pod záštitou Massachusetts Institute of Technology. Používanie programu je bezplatné. Je dostupný vo viac ako v 150 krajinách sveta, preložený bol do vyše 60 jazykov (vrátane slovenčiny) [3].

Pomôcku pre rozvoj algoritmického myslenia s použitím programovacieho prostredia Scartch predstavujú Scratch karty. Tieto karty tvoria sériu programov a sú zoradené od jednoduchých po zložité. Pri ich tvorbe sme použili online prostredie Canva². Každá kartička má úvodnú stranu s obsahom a krátkym popisom hry, ktorú budeme tvoriť.

2.2.1 Chytanie jabĺk

Veľmi jednoduchá hra je vhodná pre žiakov 5. a 6. ročníka. Tento motív hry je veľmi známy a žiaci zvyčajne nemajú problém tento program vytvoriť – pri tomto viac-menej odpisujú program z kartičky. Cieľom aktivity je zoznámenie sa s prostredím, kreslením postáv, jednoduchými scenármi a rozdelením príkazov do kategórií. Na konci kartičky sa nachádzajú rôzne návrhy na vylepšenia, ktoré môžu šikovní žiaci skúsiť naprogramovať.

2.2.2 Monster Clicker

Ďalšia jednoduchá hra so známym motívom. Aj táto karta je vhodná pre žiakov od 5. či 6. ročníka. V tejto kartičke sa už vyskytuje aj jednoduchá animácia pri kliknutí na postavu.

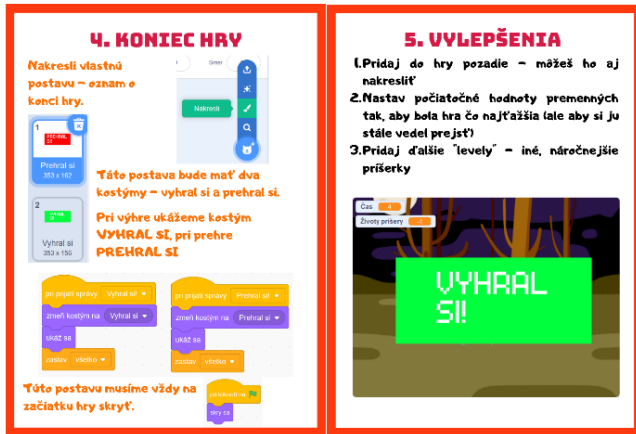
Žiaci si osvoja aj prácu s premennými a riadiaci blok podmienky. Aj tu majú žiaci možnosť použiť svoju tvorivosť pri kartičke Vylepšenia. Všetky strany tejto karty nájdeme na Obrázku 3 - 5.



Obrázok 3: Prvá strana materiálu

¹ <https://scratch.mit.edu/>

² <https://www.canva.com/>



Obrázok 4: Druhá strana materiálu



Obrázok 5: Tretia strana materiálu



Obrázok 6: Prvá strana vyučovacej pomôcky

2.2.3 Hádaj čísla

Táto karta patrí medzi pokročilejšie. Je vhodná pre žiakov 7. alebo 8. ročníka. Na rozdiel od predchádzajúcich kariet neobsahuje inštrukcie krok po kroku a od žiaka vyžaduje aj tvorivé a samostatné myslenie. V karte používame premenné, podmienky, cykly, generovanie náhodných premenných a načítavanie z klávesnice. Túto kartu sme vytvorili v čase prvej vlny pandémie pre online vyučovanie. Karty Chtanie jablka a Monster Clicker sú staršie, v škole sme ich vytlačili na farebnej tlačiarne a zalaminovali. Túto kartu sme však šírili výhradne digitálne vo forme PDF súboru, preto má aj iný vzhľad. Keďže samotná karta vyžaduje tvorivé myslenie, už tu nenájdeme možné návrhy na vylepšenie. Ukážku úvodnej strany karty vidíme na Obrázku 6.

2.2.4 Lovec

Najnáročnejšia karta je určená pre žiakov 9. ročníka a krúžku programovania. Používame grafické efekty, klonovanie, podmienku založenú na dotyku postáv. Táto karta patrí medzi najmenej odskúšané, najmä kvôli vzniknutej situácii ohľadom pandémie.

2.3 PROGRAMOVANIE LEGO ROBOTOV

Po úspechu našich kariet na záujmovom krúžku programovania hier a hodínach informatiky sme podobný nápad realizovali aj na krúžku robotiky. Tento záujmový krúžok je určený žiakom našej školy od 4. po 9. ročník. Z dôvodu toho, že sa na krúžku vyskytujú

veľmi pestré vekové skupiny žiakov, je niekedy náročné venovať sa rozvíjaniu základov programovania LEGO robotov.

V minulosti sme na krúžku programovali LEGO roboty v prostredí LEGO Mindstorms EV3, ktoré sme sa však rozhodli vymeniť za LEGO Classroom³, ktoré je oveľa prirodzenejšie a jednoduchšie na porozumenie, nakoľko je to detské blokové prostredie podobné ako Scratch. Naše karty, ktoré sme mali vytvorené na programovanie v LEGO Mindstorms EV3 sme prerobili na LEGO Classroom, ktorý by sme chceli používať na programovanie robotov.

Práca s výukovými kartičkami je veľmi jednoduchá a veľmi efektívna. Podľa úrovne znalostí žiakov / skupín žiakov sa jednotlivé kartičky rozdeľujú v triede. Od najjednoduchších kartičiek (pozri Obrázok 7 a 8) pre začiatočníkov po zložitejšie programy pre pokročilých. Následne už podľa nášho názoru nie je nutné, aby boli žiaci vedení prostredníctvom týchto výukových kartičiek, nakoľko už si úlohy vymýšľajú sami alebo navzájom.

Žiakom je takouto vyučovacou metódou vštiepená samostatnosť, respektíve spolupráca v skupinách, podporuje sa čitateľská gramotnosť a kritické myslenie, keďže často prístupujú so zadanými informáciami k vyhľadávaniu na internete.

Vďaka hmotnej podobe kartičky je pre mladších žiakov jednoduchšie pracovať s počítačom, pretože nemusia prepínať

³ <https://education.lego.com/en-us/downloads/mindstorms-ev3/software#downloads>



Obrázok 7: Vybraná strana z úvodnej karty



Obrázok 8: Vybraná strana z úvodnej karty

3 DISKUSIA

V Tabuľke 1 sme zoznámili, v ktorých ročníkoch a s akým počtom žiakov sme pri jednotlivých aktivitách pracovali. Pozn.: naši žiaci ôsmeho a deviatego ročníka začali pracovať s programovacím prostredím Sertach až v siedmom, resp. ôsmom ročníku – preto majú aktivitu vhodnú skôr pre mladšie ročníky.

Tabuľka 1: Zapojenosť žiakov

Aktivita	Ročník	Počet žiakov
Probot	Tretí	4 triedy po cca 22 žiakov
	Štvrtý	5 tried po cca 20 žiakov
Chytanie jabĺk	Šiesty	3 triedy po cca 26 žiakov
	Krúžok	15 žiakov druhého stupňa
Monster Clicker	Siedmy	4 triedy po cca 20 žiakov
	Ôsmy	3 triedy po cca 19 žiakov
	Krúžok	15 žiakov druhého stupňa
Hádaj čísla	Ôsmy	3 triedy po cca 19 žiakov
Lovec	Krúžok	15 žiakov druhého stupňa
LEGO karty	Krúžok	6 žiakov 4. ročníka, 10 žiakov druhého stupňa

Tieto kartičky nám výrazne uľahčili vyučovanie programovania – žiaci si nemuseli pred programovaním zdĺhavo zapisovať poznámky k téme hodiny.

Hneď po krátkom motivačnom úvode sme mohli začať tvoriť programy a tým sa učiť kreativite a algoritmickému mysleniu. Nadaní žiaci nemuseli čakať na zvyšok triedy, ale pomocou vylepšení dokázali ukázať svoju šikovnosť a zároveň ďalej rozvíjať svoj talent.

Od žiakov sme dostali pozitívnu spätnú väzbu. Kartičky im pomohli pri zorientovaní sa v programovacom prostredí, mohli sa vrátiť v programe a opraviť si chyby a grafickou úpravou ich zaujali viac, ako klasické poznámky či pracovné listy.

4 ZÁVER

Probot sme používali v školskom roku 2019/2020 na prvom stupni základnej školy. Kvôli pandemickej situácii sme v novom školskom roku na tieto aktivity nemohli nadviazať. Po uvoľnení situácie plánujeme v týchto aktivitách pokračovať.

Scratch karty sme vytvárali v priebehu školského roka 2019/2020 a využívali sme ich počas celého školského roka. Prirodzene, využívame ich aj v tomto školskom roku. Okrem týchto kariet máme v zbierke ešte približne 6 ďalších návodov, ktoré by sme však pred použitím na vyučovaní chceli upraviť. Niektoré z nich sme nahradili novšími kartami a, pravdepodobne, ich už používať nebudeme. To sú hlavné dôvody, prečo sme ich v tomto článku nepublikovali. V budúcnosti by sme na krúžku programovania hier, prípadne aj vo vyučovaní informatiky radi zaviedli aj prostredie MakeCode Arcade⁴.

Karty na prácu s LEGO robotmi sme pripravili počas školského roka 2020/2021, ale kvôli pandemickej situácii sme ich nemohli využiť na krúžku robotiky. Naše karty pre prostredie LEGO Mindstorms EV3 postupne prerábame pre nové prostredie LEGO Classroom.

Úplné verzie našich materiálov si môžete stiahnuť na odkaze: <https://polonec.eu/vyucovacie-pomocky/>. Prirodzene, v priebehu príprav na vyučovanie vytvárame ďalšie učebné pomôcky. Všetky učebné materiály, ktoré máme uverejnené na internete môžete nájsť na stránke polonec.eu.

Veríme, že vidíte v takýchto materiáloch a pomôckach na vyučovaní informatiky zmysel a inšpirovali sme Vás k tvorbe vlastných materiálov.

POĎAKOVANIE

Ďakujeme PaedDr. Patrikovi Voštinárovi, PhD. za cenné rady pri písaní tohto článku.

Tento článok bol vytvorený v rámci národného projektu IT Akadémia – vzdelávanie pre 21. storočie, ktorý sa realizuje vďaka podpore z Európskeho sociálneho fondu a Európskeho fondu regionálneho rozvoja v rámci Operačného programu Ľudské zdroje.

BIBLIOGRAFICKÉ ODKAZY

[1] FUTSCHEK, G. *Algorithmic Thinking: The Key for Understanding Computer Science*. Viedeň: Vienna University of Technology, 2006, p. 10, cit. 2021-01-21, dostupné on-line: https://publik.tuwien.ac.at/files/PubDat_140308.pdf

[2] CZAKÓOVÁ, K., STOFFOVÁ, V. *Hravá forma rozvíjania algoritmického myslenia na základnej škole*. In DidInfo 2020. Liberec: Technická univerzita v Liberci, Fakulta

⁴ <https://arcade.makecode.com/>

DIDINFO 2021 Univerzita Mateja Bela, Banská Bystrica

ISBN 978-80-557-1823-1 ISSN 2454-051X

přírodovědně-humanitní a pedagogická, 2020, s. 104 - 111.

ISBN 978-80-7494-532-8. Dostupné tiež na:

http://didinfo.net/images/DidInfo/files/Didinfo_2020.pdf

- [3] SCRATCH. *About Scratch*. Cit. 2021-01-21. Dostupné on-line: <https://scratch.mit.edu/about>

Porovnávacia štúdia pravdepodobnostného prístupu a digitálneho vodoznaku pri odhaľovaní plagiátorstva v dištančnom kurze programovania

Comparative study of probabilistic approach and digital watermarking in the detection of plagiarism in the distance course of programming

Lubomír Salanci

Fakulta matematiky, fyziky a informatiky Univerzita Komenského
Mlynská dolina
842 48 Bratislava
Slovenská republika
salanci@fmph.uniba.sk

ABSTRACT

A probabilistic approach based on source code comparisons, for example using the greedy string tiling algorithm, is often used to detect plagiarism. On the contrary, digital watermark is used more in the field of copyright protection. We used both approaches to detect plagiarism in our distance course of programming. During the semester, we collected a lot of empirical data that allowed us to compare both approaches. Since our students were solving trivial problems, the use of the probabilistic approach was delicate not only in terms of detection but also in subsequent argumentation. The digital watermark was effective in situations where students copied the source code with a watermark. This allowed us to understand the work of students during distance learning and, in many cases, to detect plagiarism at the level of individual lines in the source code.

Keywords

Programming. Distance learning. Plagiarism. Greedy string tiling. Digital watermark.

ABSTRAKT

Pri odhaľovaní plagiátorstva sa často používa pravdepodobnostný prístup založený na porovnávaní zdrojových kódov, napríklad pomocou algoritmu greedy string tiling. Naopak, digitálny vodoznak sa používa skôr v oblasti ochrany autorských práv. Pri odhaľovaní plagiátorstva v dištančnom kurze programovania sme použili oba prístupy. Počas semestra sme nazbierali množstvo empirických údajov, ktoré nám dovolili oba prístupy porovnať. Keďže naši študenti riešili triviálne úlohy, použitie pravdepodobnostného prístupu bolo chýlostivé nielen z hľadiska detekcie, ale aj následnej argumentácie. Digitálny vodoznak bol efektívny v situáciách, keď študenti skopirovali kód s vodoznakom. Zároveň nám umožnil porozumieť práci študentov počas dištančného vzdelávania a odhaľovať plagiátorstvo až na úrovni jednotlivých riadkov v ich zdrojovom kóde.

Kľúčové slová

Programovanie. Dištančné vzdelávanie. Plagiátorstvo. Greedy string tiling. Digitálny vodoznak.

1 ÚVOD

V úvodnom kurze programovania pre prvákov matematikov na vysokej škole kladieme veľký dôraz na to, aby sa študenti počas semestra trénovali v rozmyšľaní a získavali praktické skúsenosti

v programovaní. Prezenčné cvičenia pripomínajú skôr prácu v laboratóriu, kde každý študent rieši vlastným tempom zadané úlohy na svojom počítači. V našom prieskume sa približne 50 % študentov vyjadriilo, že nikdy neprogramovali. Preto máme kurz programovania nastavený tak, aby ho zvládali na známku A aj úplní začiatocníci. Tak sa môžu úlohy, ktoré naši študenti riešia, javiť skúsenému programátorovi ako triviálne.

S príchodom pandémie sme museli kurz programovania transformovať do dištančnej podoby, a to podľa daných pokynov. Teda, prednášky aj cvičenia majú prebiehať on-line. Zároveň sa vynorili mnohé nejasnosti, napríklad, či môžeme od študentov vyžadovať on-line účasť na cvičeniach a skúškach. Aby sme minimalizovali riziká, ktoré vyplývali z nejasných pravidiel, upravili sme kritériá pre absolvovanie dištančného kurzu tak, aby boli podmnožinou kritérií prezenčného kurzu – zrušili sme skúšku a on-line cvičenia sa stali nepovinnými. Stále sme však trvali na tom, aby počas semestra študenti pracovali a preukazovali to tým, že nám budú odovzdávať vyriešené úlohy z cvičení.

Dištančné vzdelávanie a e-learning sú veľmi rozsiahle, aktuálne a diskutované oblasti, majú svoje výhody aj nevýhody [1]. Napríklad, na prezenčnom cvičení sme mali prehľad o tom, ako sa ktorému študentovi darí a dokázali sme ihneď reagovať. Ešte dôležitejšie však bolo, že študenti sa medzi sebou, miestami až vášnivo, bavili o úlohách a (na) svojich riešeniach. To zároveň vytváralo výbornú pracovnú atmosféru. V dištančnom kurze sa takáto sociálna zložka úplne vytratila.

Naše on-line prednášky a cvičenia prebiehali cez MS Teams. Narozdiel od prezenčných prednášok, na ktorých sa zúčastňovalo 50 % – 90 % študentov, sa na on-line prednáškach a cvičeniach zúčastňovalo iba 11 % študentov. Pravdepodobne študentom viac vyhovovali web stránky predmetu a e-mailová komunikácia. Na web stránkach sme postupne zverejňovali prednášky, cvičenia a vzorové riešenie k úlohám z cvičení. Študenti odovzdávali vyriešené úlohy podľa pokynov do systému Moodle, v ktorom zároveň videli aj výsledné hodnotenie a naše komentáre. Na hodnotení sa podieľali 4 učitelia, ktorí počas semestra skontrolovali približne 7000 úloh.

Medzi odovzďávaním úloh a zverejňovaním vzorových riešení sme hľadali kompromis. Nechceli sme, aby študenti podvádzali tým, že by riešenia opisovali alebo kopirovali. Tak by sa programovať nenaučili. Zatajovať riešenia však nemalo zmysel,

keďže tie už niekoľko rokov kolovali medzi študentmi. Na druhej strane, vymýšľať nové a kvalitné úlohy by sme nestíhali. Tiež sa nám zdalo férové, aby mali študenti pomôcku, ak by sa s niektorou úlohou veľmi trápili, a my by sme neboli k dispozícii, prípadne by si chceli porovnať svoje riešenie s našim vzorovým riešením. Veď podobne to funguje aj v matematike, keď na konci cvičebnice nájdeme riešenia úloh. Preto sme sa rozhodli, že vzorové riešenia budeme zverejňovať týždeň po cvičeniach a študentom zostane ešte týždeň času na odovzdanie svojich riešení.

2 PLAGIÁTORSTVO A PODVÁDZANIE

S podvádzaním študentov máme dlhoročné skúsenosti. Najviac pokusov o podvádzanie vidíme pri študentoch prvého ročníka. Vo vyšších ročníkoch miera podvádzania klesá. Na elimináciu podvádzania existuje niekoľko rôznorodých stratégií [2].

Napríklad, kedysi dávno tvoril semestrálny projekt v našom kurze významnú zložku hodnotenia. Našli sa však študenti, ktorí si projekt nechali naprogramovať kamarátom alebo na objednávku. Pri dokazovaní podvádzania sme potom zažívali rôzne konfliktné situácie, ktoré nás stáli veľa času a množstvo energie. Tak sme dospeli k rozhodnutiu, že tie činnosti študentov, ktoré nemáme pod vlastnou vizuálnou kontrolou, budú mať na výsledné hodnotenie minimálny vplyv. Tento princíp však nedokážeme aplikovať pri dištančnom vzdelávaní.

Jednou z možností by bolo, že podvádzanie si nebudeme všimáť, budeme ho ignorovať. Spoločne sa na to, že študenti sa predsa prišli na vysokú školu učiť, a ten, kto podvádzá, klame seba. Má to však niekoľko rizík. Okrem toho, že tolerovanie podvádzania znechucuje poctivých študentov a znižuje kredit vysokej školy, môže sa obráť aj proti nám. Napríklad vtedy, keď učiteľia nadväzujúcich predmetov očakávajú deklarované znalosti od študentov, ktorí náš predmet úspešne absolvovali.

Ďalšou z možností, ktorá je obsiahnutá aj v študijnom poriadku, je penalizácia. Aj v našom dištančnom kurze sme mali pravidlo, že hocijaký pokus o podvádzanie automaticky znamená hodnotenie FX. Napriek tomu sa našlo zopár študentov, ktorí nás chceli ihneď na začiatku kurzu podvieť tým, že odovzdali identické riešenia. Takýto spôsob podvádzania sme ľahko odhalili, keď sme porovnávali veľkosti odovzdaných súborov. Následne sme videli, že súhlasil čas modifikácie, obsahy súborov a UUID (universally unique identifier) oboch projektov.

Najväčšiu mieru podvádzania v našom dištančnom kurze tvorilo kopírovanie alebo opisovanie zo vzorových riešení, ktoré mali slúžiť ako pomôcka. Študenti tak robili napriek zákazu a upozorneniu, že takéto podvádzanie dokážeme odhaliť. Kopírovanie cudzích riešení je bežný a známy jav. Vidno to aj na veľkom množstve článkov, ktoré sa venujú softvérovému plagiátorstvu [3, 4, 5].

3 PRAVDEPODOBNOSTNÝ PRÍSTUP

Existuje aj veľa nástrojov na detekciu plagiátorstva [6, 7], ktoré by sme mohli použiť. Tieto nástroje sú založené na pravdepodobnostnom princípe, ktorý ilustrujeme na jednej z úloh z našich cvičení:

Benzínová stanica, šofér načerpal benzín. Do textového políčka zadáme počet litrov (môžu byť aj desatiny) a počítač nám zobrazí, koľko má zákazník zaplatiť. Cenu benzínu odhadnite.

```
label1.Text = Convert.ToString(Convert.ToDouble(textBox1.Text) * 1.2);
```

Obrázok 1: Riešenie bez premenných

```
double a;  
a = Convert.ToDouble(textBox1.Text);  
label1.Text = Convert.ToString(a * 1.145);
```

Obrázok 2: Riešenie s premennou

```
double cena;  
cena = 1.17;  
label1.Text=Convert.ToString(cena * Convert.ToDouble(textBox1.Text));
```

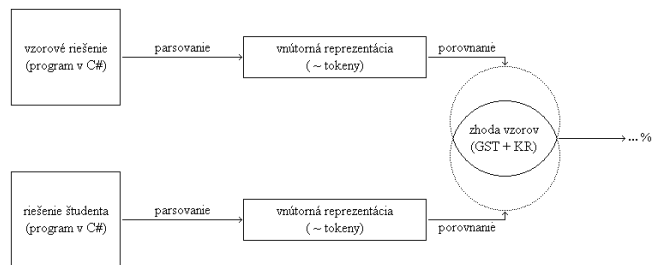
Obrázok 3: Riešenie akoby s konštantou

```
Graphics g = CreateGraphics();  
double l;  
string a;  
l = Convert.ToDouble(textBox2.Text) * 1.163;  
a = Convert.ToString(l);  
g.DrawString(a, new Font("Arial",10), Brushes.Black, 215, 60);
```

Obrázok 4: Riešenie s grafickým výstupom

Na Obrázkoch 1, 2, 3, 4 vidíme, že štyria rôzni študenti zapísali riešenie aj veľmi triviálnej úlohy štyrmi rôznymi spôsobmi, nehovoriac o tom, že cenu benzínu treba odhadnúť a že študenti si môžu zvoliť iné mená premenných, ak ich použijú.

Podobne môžeme uvažovať aj o ostatných úlohách, pričom každá z nich sa dá riešiť dvoma alebo aj tisíc rôznymi spôsobmi. Ak cvičenie obsahuje 10 triviálnych úloh, počet kombinácií, ako môže celé vyriešené cvičenie vyzeráť, rastie exponenciálne. Teda, pravdepodobnosť, že niekto vyrieši celé cvičenie identicky s naším vzorovým riešením, s počtom úloh rýchlo klesá.



Obrázok 5: Tok údajov a operácii pri porovnávaní

Porovnávanie odovzdaných úloh so vzorovými riešeniami sme realizovali našim vlastným nástrojom (pozri Obrázok 5). Vstupné textové súbory sú najskôr spracované parserom, ktorého úlohou je vykonať abstrakciu nad zdrojovým kódom, odfiltrovať komentáre a nezaujímavé znaky, ale aj zabezpečiť odolnosť voči jednoduchým transformáciám kódu, akými sú premenovanie premenných alebo, zmena konštant. Postupnosti lexém sú následne porovnávané algoritmom greedy string tiling (ďalej len GST) s rýchlym vyhľadávaním Karp-Rabin [8]. Výsledkom porovnania je číslo, ktoré určuje podobnosť riešení v percentách.

Popis implementačných a kalibračných detailov nášho nástroja presahuje rámec príspevku. Len skonštatujeme, že nástroj sme postupne vyladili tak, aby naše kolegyně, ktoré ho používali, dokázali výsledky porovnania správne interpretovať:

- 90 % a vyššia zhoda znamená, že je veľká pravdepodobnosť podvádzania,
- nad 80 % znamená, že riešenie je podozrivé.

Najväčšou nevýhodou akéhokoľvek podobného systému na odhaľovanie plagiátorstva je jeho pravdepodobnostná povaha. Aj preto sme intenzívne premýšľali, ako by sme exaktne preukázali, že niekto kopíroval zo vzorových riešení.

4 DIGITÁLNY VODOZNAK

Nakoniec sme vymysleli pomerne jednoduchý, ale účinný spôsob, ktorý sa v literatúre označuje ako digitálny vodoznak [9]. Ten sme začali vkladať do vzorových riešení. Použili sme vhodnú kombináciu bielych znakov (white spaces) zo sady UNICODE, ktoré nenarušujú čitateľnosť programového kódu ani na web stránke, ani v textovom editore vývojového prostredia SharpDevelop. Kompilátor jazyka C# si s nimi poradí a vzniká nulová šanca, že by študent omylom takéto znaky napísal.

Ak študent nedodrží naše pravidlá a časť kódu s vodoznakom skopíruje, „kontaminuje“ si svoje riešenie. Potom, pomocou vhodného textového editora, dokážeme v odovzdanom súbore vodoznak ľahko odhaliť. Často dokonca vieme určiť, a to až na úrovni jednotlivých riadkov, ako pri riešení postupoval.

Takýto vodoznak je odolný voči viacerým úpravám textu: formátovanie programového kódu, zmena poradia príkazov, premenovanie premenných, zmeny číselných konštánt a ďalšie.

Nevýhodou nášho digitálneho vodoznaku je, že sa dá ľahko odstrániť. Navyše, ak ho niektorý študent objaví, určite sa pochváli ostatným spolužiakom a táto technika odhaľovania plagiátorstva prestane fungovať.

S vodoznakom musíme narábať veľmi opatrne. Jednak preto, aby sme zbytočne našu stratégiu neprezradili, a potom aj preto, aby sa vodoznak príliš nerozšíril a nechceme nekontaminovať iné riešenia napríklad tým, že by študent pri riešení nového cvičenia občas skopíroval časť kódu (typicky inicializáciu grafiku, výraz pre náhodné generovanie farby, nové písmo a podobne) zo svojich starších kontaminovaných riešení. Tomu sa dá zabrániť, ak na web stránke bude riešenie s vodoznakom zverejnené iba počas nevyhnutne krátkej doby, napríklad iba do termínu odovzdávania. Potom zverejnené riešenie nahradíme čistým vzorovým riešením, bez vodoznaku. Aby sme rozlíšili, z ktorého vzorového riešenia vodoznak pochádza, môžeme použiť pri každom cvičení iný vodoznak.

5 ZBER A SPRACOVANIE ÚDAJOV

Naši prváci, ktorých sme učili v zimnom semestri, boli prví, ktorých zasiahla pandémia a dištančné vzdelávanie ešte na strednej škole vo 4. ročníku.

Počas semestra sme realizovali 11 prednášok a 10 cvičení:

- 1. prednáška a cvičenie bolo ešte prezenčné, a teda študenti nemuseli svoje riešenia z tohto cvičenia odovzdávať. Preto ani nie sú zahrnuté v našich výsledkoch.
- 2. prednáška bola prezenčná, ale cvičenie bolo realizované čiastočne prezenčne a čiastočne dištančne. Vzhľadom na avizované pandemické opatrenia sa už časť študentov nemohla zúčastniť prezenčného cvičenia. Tí, ktorí prezenčné cvičenie absolvovali, riešili prvú polhodinu zadané úlohy. Ďalšiu časť cvičenia sme venovali systému Moodle. Študenti si vyskúšali prihlasovanie, orientáciu a odovzdávanie svojich riešení. Následne mali doma dokončiť úlohy z cvičenia a finálnu verziu odovzdať do systému Moodle. Kontrolovali sme iba, či študenti dodržali formálne kritériá: termín odovzdania a odovzdanie požadovaných súborov. Preto ani tieto cvičenia nie sú zahrnuté vo výsledkoch. Pri manuálnej kontrole odovzdaných riešení sme si však

všimli, že viacerí študenti nepracujú podľa našich pokynov – teda, riešenia nevymýšľajú, ale kopírujú.

- 3. prednáška a cvičenie bolo realizované už iba dištančne. Ešte sme nepoužívali žiadne mechanizmy na odhaľovanie plagiátorstva.
- 4. až 10. prednáška a cvičenie boli tiež realizované dištančne. Pri kontrole odovzdaných riešení sme však už používali náš nástroj na porovnávanie riešení a zverejňovali sme vzorové riešenia, do ktorých sme vkladali vodoznak.
- 11 prednáška bola dištančná, k nej cvičenia neboli.

Zverejňovanie vzorových riešení počas dištančného vzdelávania na webe prebiehalo nasledovne:

- zverejnili sme prednášku a cvičenie,
- po týždni sme zverejnili vzorové riešene s vloženým vodoznakom (pre každé cvičenie iný variant),
- po ďalšom týždni sme stiahli zo systému Moodle riešenia študentov,
- odstránili sme vodoznak zo vzorových riešení na webe.

Pri riešení úloh mali študenti dovolené kopírovať kód z prednášky a oficiálneho ťaháka. Ten obsahoval zoznam príkazov, ktoré sa postupne dozvedali. Kopírovanie a opisovanie kódu zo vzorových riešení mali študenti prísne zakázané. Informovali sme ich, že takéto plagiátorstvo vieme zistiť a upozornili na to, že v prípade podvádzania im z predmetu hrozí hodnotenie FX. Študenti sa ale mohli radiť medzi sebou. Tiež sa pýtali a radili aj s nami – v čase prednášky alebo cvičenia cez MS Teams, inokedy cez e-mail.

Riešenia, ktoré študenti odovzdali do systému Moodle, sme stiahli a hodnotili v niekoľkých fázach:

- kontrola, či študent odovzdal potrebné súbory,
- kontrola originality (porovnanie riešení so spolužiakmi, porovnanie so vzorovým riešením, detekcia vodoznaku),
- kontrola správnosti a funkčnosti riešenia,
- diskusia s kolegyňami o problematických riešeniach a ich hodnotení,
- hodnotenie a spätná väzba študentovi (pochvaly, komentáre, v prípade podvádzania aj varovanie).

V 2. a 3. cvičení sme originalitu študentských riešení kontrolovali manuálne. Pri cvičeniach 4 až 10 sme použili náš nástroj na porovnávanie riešení. V týchto riešeniach sme zároveň zisťovali aj prítomnosť digitálneho vodoznaku. Pre zaujímavosť a potreby toho článku sme pomocou nášho GST nástroja spätne porovnali aj riešenia z 3. cvičenia. Prirodzene, prítomnosť vodoznaku sme v nich už zisťovať nemohli.

6 VÝSLEDKY

Kurz programovania si zapísalo 104 študentov. Počas dištančného vzdelávania sme zistili, že 66 študentov sa nás pokúsilo nejakým spôsobom podviesť. To je približne 63 % študentov.

Podľa mien študentov sme odhadli, že kurzu sa zúčastnilo 44 mužov a 60 žien. Podvádzalo 28 mužov a 38 žien. Prepočítané na percentá, je pomer podvodníkov a podvodníčok približne rovnaký.

Ďalej sa v tejto kapitole budeme zaoberať iba podvádzaním, ktoré bolo založené na kopírovaní alebo opisovaní kódu zo vzorových riešení k cvičeniam. Prvé cvičenie bolo prezenčné, takže výsledky

z neho neuvádzame. Podobne neuvádzame ani výsledky z druhého cvičenie, v ktorom sme kopírovanie a opisovanie ešte nezisťovali.

Na Obrázku 7 vidíme výsledky detekcie plagiátorstva pomocou algoritmu GST a vodoznaku.

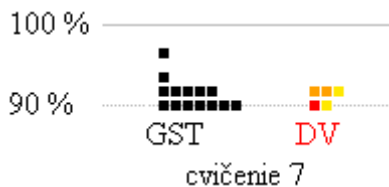
X-ová os obsahuje 7 skupín, ktoré zodpovedajú cvičeniam 4 až 10. V každej skupine potom vidíme dvojice:

- GST = podobnosť podľa GST,
- DV = zastúpenie digitálneho vodoznaku.

Y-ová os zodpovedá škále podobnosti 0 % – 100 % podľa GST.

Význam značiek v grafe je nasledovný:

- čierna značka = nakoľko sa študentovo riešenie podobá vzorovému riešeniu na webe podľa algoritmu GST (výsledok v % odčítame na y-ovej osi),
- farebná značka = odovzdané riešenie obsahovalo vodoznak, pričom červený štvorček znamená, že v odovzdanom riešení bolo 75 % až 100 % riadkov skopírovaných, oranžový štvorček znamená, že študent skopíroval 30 % až 74 % riadkov a žltý štvorček označuje riešenie, v ktorom bol aspoň jeden, ale menej ako 30 % riadkov skopírovaných zo vzorového riešenia,
- aby sa značky neprekrývali, sú za sebou kumulované, čím vzniká histogram.



Obrázok 6: Detail k porovnaniu GST a vodoznaku.

Graf treba čítať nasledovne (pozri Obrázok 6): vidíme, že riešenia siedmich študentov zo 7. cvičenia sú na 90 % podobné vzorovým riešeniam, pričom jeden z týchto študentov skopíroval viac ako 75 % riadkov a iný do 30 % riadkov kódu zo vzorových riešení. Nad

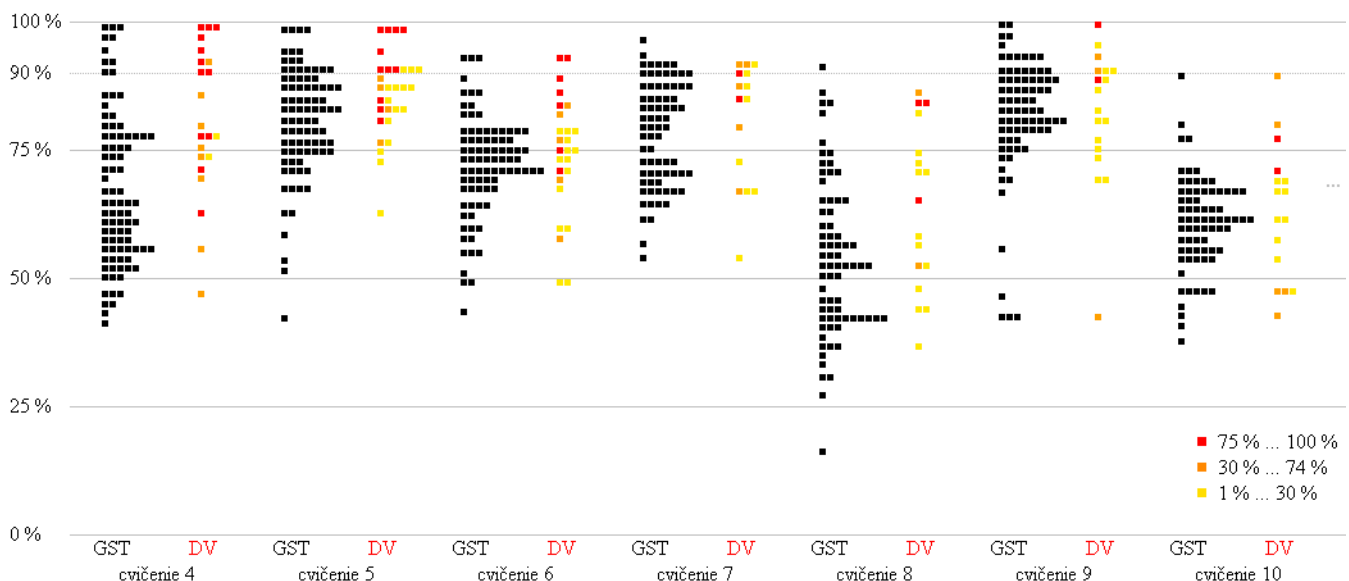
tým môžeme vidieť, že riešenia piatich študentov sa podobali vzorovým riešeniam približne na 92 %, pričom dvaja z nich skopírovali od 30 % do 74 % riadkov a jeden do 30 % riadkov kódu. Nad nimi vidíme riešenia ďalších dvoch študentov, ktoré sa podobali vzorovým riešeniam ešte viac, no títo študenti nič nekopírovali.

Môžeme si všimnúť, že väčšina čiernych značiek GST na Obrázku 7 sa nachádza nad hranicou 50 %. Takáto vysoká podobnosť študentských riešení so vzorovými riešeniami vyplýva z toho, že študenti riešili triviálne úlohy. Navyše, aj samotný SharpDevelop automaticky generoval časť študentovho riešenia (hlavičky metód pre obsluhu udalostí). Prirodzene, tento kód sa tiež vyskytoval vo vzorových riešeniach. Podobné výsledky sme získali aj pri použití nástroja JPlag.

Ďalej môžeme na Obrázku 7 vidieť, že červené značky pre vodoznak, ktoré zodpovedajú vysokej miere kopírovania, ležia prevažne v hornej časti grafu, a teda zodpovedajú vysokej miere podobnosti podľa GST. Avšak toto nie je vždy pravidlom.

Môžeme nájsť viacero situácií, keď riešenie študenta vykazuje podľa vodoznaku vysokú mieru kopírovania, ale nízku (približne iba 60 % až 70 %) podobnosť podľa GST.

Pri analýze odovzdaných zdrojových kódov sme zistili, že títo študenti veľa vecí skopírovali z riešení, ale zároveň sa buď snažili časti úloh riešiť samostatne, alebo sa pokúšali o to, aby sa výsledok čo najmenej podobal zverejnenému riešeniu. Nezriedka sa študenti pri kopírovaní pomýlili, a potom sa snažili chybu opraviť. Napríklad, na 6. prednáške sme preberali podprogramy. Na cvičeniach mali trénovať definovanie a volanie vlastných podprogramov. Niektorí študenti však kopírovali a definíciu vložili na nesprávne miesto alebo zabudli na volanie podprogramu. Aplikácia sa potom nedala spustiť alebo nezobrazila žiaden výsledok. Situáciu študenti zachraňovali tým, že telo podprogramu nakopírovali na miesta, na ktorých malo byť volanie podprogramu a v skopírovaných príkazoch upravili parametre (tzv. inline expansion). Preto odovzdané riešenie obsahovalo množstvo riadkov s vodoznakom, bolo však príliš odlišné od vzorového riešenia, a tak nebolo podľa GST podozrivé.



Obrázok 7: Porovnanie GST a vodoznaku.

Obrázky 8, 9 a 10 nám dávajú iný pohľad na výsledky detekcie kopírovania alebo opisovania z cvičení 3 až 9. Jednotlivé stĺpce zodpovedajú 73 študentom, ktorými sa budeme ďalej zaoberať. Nie všetci z nich podvádzali. Krížik znamená, že študent riešenie neodovzdal.

Čiernym štvorčekom na Obrázku 8 sme označili tie riešenia študentov, ktoré boli aspoň na 90 % podobné vzorovému riešeniu. Už sme spomínali, že vzhľadom na triviálny charakter úloh v našom kurze, nemôžeme o týchto riešeniach automaticky prehlásiť, že sú plagiáty. Môžeme iba povedať, že tieto riešenia sú veľmi podozrivé a ešte ich treba manuálne skontrolovať.

Vďaka vodoznakom môžeme na Obrázku 9 sledovať individuálne správanie jednotlivých študentov:

- zopár študentov sa pokúsilo kopírovať iba raz (napríklad, študenti 28 alebo 56),
- viacerí študenti kopírovali sporadicky,
- študent 46 neustále kopíroval, ale iba určité časti riešenia.

Vidíme, že vodoznak sa vyskytoval v niektorom z riešení u 60 študentov. O nich môžeme s určitosťou prehlásiť, že podvádzali. Podvádzateľ mohli aj ostatní študenti, ale pri nich už takéto exaktný dôkaz nemáme.

Ďalšie súvislosti môžeme sledovať na Obrázku 10, ktorý vznikol spojením Obrázkov 8 a 9. Čiernou bodkou sme navyše označili riešenia, ktoré sme nakoniec prehlásili za plagiáty alebo čiastočné plagiáty (zhodli sa na tom aspoň 2 učители). Takýchto prípadov bolo 147. Pre zaujímavosť uvádzame v spodnom riadku finálne hodnotenie študenta z predmetu.

Na obrázku si môžeme všimnúť tieto situácie:

- Falošná negativita GST pri 90 % podobnosti – hoci 96 študentských riešení vykazovalo podľa GST nižšiu ako 90 % podobnosť so vzorovými riešeniami, boli tieto riešenia plagiátmi. Z nich sme 72 identifikovali vďaka

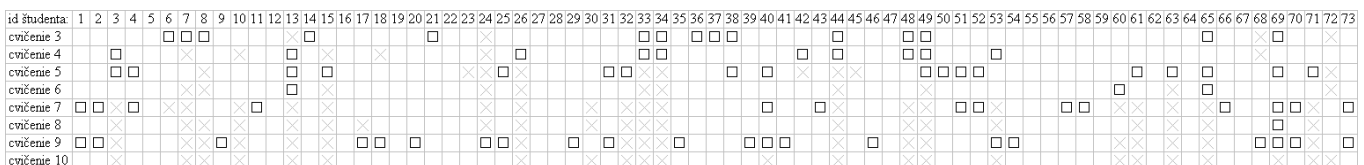
prítomnosti vodoznaku (farebné políčka s bodkou, bez čierneho štvorčeka). Po manuálnej kontrole sme za plagiát označili ešte aj ďalších 24 riešení, v ktorých nebol prítomný vodoznak (políčko s čiernou bodkou).

- Falošná pozitivita GST pri 90 % podobnosti – hoci GST signalizovalo v 30 prípadoch vysokú pravdepodobnosť podvádzania, po manuálnej kontrole sme si tieto riešenia študentov netrúfli označiť ako plagiáty. Zároveň tieto riešenia neobsahovali vodoznak (políčko, na ktorom je iba čierny štvorček).
- Viaceré riešenia vykazovali nízky počet riadkov s vodoznakom, a zároveň aj mali nízku podobnosť podľa GST (iba farebné políčko bez ďalších značiek). Pri manuálnej kontrole sme videli, že študent skopíroval iba neďôležitú časť, zväčša iba jeden riadok (napríklad inicializáciu grafiky). Vtedy sa nám neoplatilo riskovať prezradenie našej stratégie s vodoznakom, preto sme takéto riešenie študenta neoznačovali ako plagiát, a ani sme ho pri hodnotení nepenalizovali.
- Naopak, takmer všetky riešenia s nízkym zastúpením vodoznaku, ale vysokou podobnosťou podľa GST sme označili ako plagiát (žlté políčko s čiernym štvorčekom a bodkou). Pri manuálnej kontrole sme videli, že študent časť riešenia skopíroval a zvyšok opísal zo vzorových riešení.

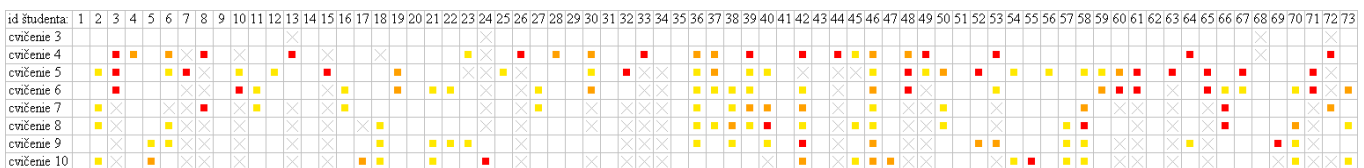
Na cvičení 3 sme riešenia posudzovali iba manuálne a podobnosť sme zisťovali až dodatočne. Vidíme však, nakoľko manuálne posúdenie korešponduje s 90 % podobnosťou podľa GST.

7 ZÁVER

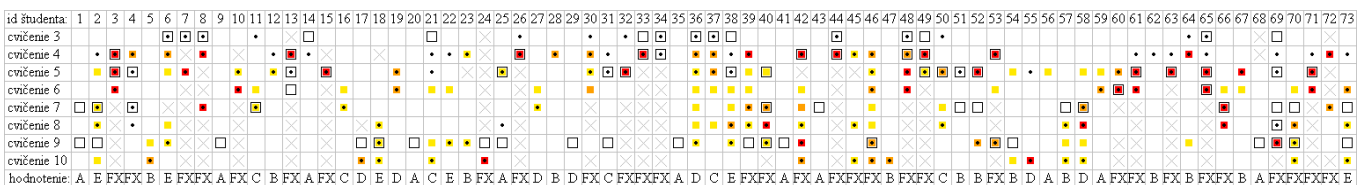
Cieľom cvičení bolo, aby študenti počas semestra pracovali a vymýšľali riešenia triviálnych úloh. Cvičenie nemalo byť skúškou alebo testom, malo byť tréningom. S takouto filozofiou sme pristupovali aj k hodnoteniu, ktoré bolo založené na poctivom prístupe k práci, nie na kvalite alebo úplnej správnosti riešenia.



Obrázok 8: Podobnosť nad 90 % podľa GST.



Obrázok 9: Detekcia pomocou vodoznaku.



Obrázok 10: Porovnanie detekcie GST, prítomnosti vodoznaku a hodnotenia.

Preto nás prekvapila miera podvádzania, s ktorou sme sa stretli. Očakávali sme, že podvádzat' bude do 10 študentov, ale nie 66, mnohí z nich aj opakovane a po upozornení. Vďaka použitiu digitálneho vodoznaku sme mohli exaktne a s istotou zistiť plagiátorstvo pri 60 študentoch. Nie všetkých sme však obvinili.

Pravdepodobnostný nástroj GST slúžil väčšinou ako pomôcka, vďaka ktorej sme ľahšie dokázali vytipovať podozrivé riešenia. Pri definitívnom rozhodovaní sa o tom, či podozrivé riešenie je alebo nie je podvodné, sme museli byť opatrní. Takmer istí sme si boli iba pri riešeniach, ktoré boli nad 95 % podobné so vzorovým riešením, aj to iba po vizuálnej kontrole a prípadnej diskusii s ďalšími kolegyňami. Vždy to bolo veľmi chúlостivé rozhodovanie a výrazne záviselo od riešených úloh.

Korelácia medzi detekciou pomocou digitálneho vodoznaku a podobnosťou z GST porovnávaní je veľmi slabá. Korelačný koeficient sa pohybuje od 0.1 po 0.5 v závislosti od cvičenia.

Zaujímavý je aj sociálny a psychologický aspekt podvádzania. Iba naznačíme, že mnoho študentov bolo veľmi prekvapených, že plagiátorstvo dokážeme odhaliť. Ešte viac ich rozrušovalo, že z cvičenia, pri ktorých podvádzali, nezískavali body. Pritom bolo zaujímavé, že najviac sa sťažovali a protestovali študenti, ktorí aj najviac podvádzali. Keď neuspeli u nás, sťažovali sa rodičom, gestorom štúdia, prodekanke, prípade sa vyhrážali súdmi. Táto komunikácia nás extrémne zatažovala a vyčerpávala. Nebyť vodoznaku, veľmi ťažko by sme ju ustáli. Takto sme mohli na útoky študentov reagovať s prehľadom a s čistým svedomím. Iba zopár študentov sa ospravedlnilo. Jedna študentka to vysvetlila tak, že konala skratovo.

Z nášho pohľadu by sme činnosti, ktoré súviseli s podvádzaním študentov, mohli rozdeliť do troch fáz:

- detekcia,
- rozhodnutie – označenie / obvinenie,
- komunikácia – v prípade sťažnosti študenta.

Tieto fázy sa v rôznej miere kombinovali, napríklad:

- študent skopíroval časť alebo celé riešenie, prípadne skopírovaný kód zmodifikoval – vďaka prítomnosti digitálneho vodoznaku sme podvod ľahko odhalili, mali sme silný argument, exaktný dôkaz a sebedomý postoj pri komunikácii, ale študentovi sme nesmeli prezradiť o existencii vodoznaku,
- študent ručne opísal celé riešenie, prípadne zmenil mená premenných a konštanty – podvod sme ľahko odhalili pomocou nástroja GST, ale obhájenie nášho postoja bolo náročnejšia na argumentáciu,
- študent ručne opísal iba časť riešenia, prípadne zmenil mená premenných a konštanty – podvod sme odhalili iba pri určitých úlohách a iba pri manuálnej kontrole, vo vyvodení uzáverov sme museli byť obzvlášť opatrní.

Bez použitia vodoznaku by sme veľa podvodov nemali šancu odhaliť, ale čo je hlavné, nedokázali by sme ich vyargumentovať. Digitálny vodoznak mal silnú dôkaznú hodnotu. Vďaka nemu sme získali obrovskú výhodu.

Aj kvôli právnymi dôsledkom sme museli postupovať veľmi obozretne a rozlišovať, kedy riešenie označíme za podvod. Týkalo sa to najmä nástroja GST. Ukázalo sa, že málokto zo študentov a rodičov rozumie a dôveruje pravdepodobnostným technikám („nie je predsa možné úlohu riešiť viacerými spôsobmi – to mi potvrdil

profesionál“). Preto v prípade úloh, ktoré riešia naši študenti ani nevidíme zmysel v používaní prefikanejších algoritmov na odhaľovanie plagiátorstva, keďže sa ukázalo, že nestačí podvod iba odhaliť, ale náš postoj potrebujeme aj obhájiť (napríklad, pred disciplinárnou komisiou).

Dištančný kurz nezvládlo 38 študentov, čo je približne 37 %, študentov. Pre zaujímavosť, prezenčné kurzy v minulých rokoch nezvládalo približne 20 % študentovo. Študenti získali zlé hodnotenie zväčša kvôli podvádzaniu s kopírovaním. Naopak, A dosiahlo 33 študentov, B 12, C 7, D 7 a hodnotenie E 7 študentov.

Aj keď sa zverejňovanie riešení ešte pred termínom odovzdania môže javiť ako veľmi zlý nápad, ktorý viedol k obrovskému počtu podvodov, myslíme si, že sa na tento prístup dá pozerat' aj inak. Vďaka zverejňovaniu riešení sme lepšie porozumeli tomu, ako študenti počas semestra v dištančnom vzdelávaní fungujú. Domnievame sa, že prácu študentov sme mali oveľa viac pod kontrolou, než keby sme riešenia utajovali. Utajenie vzorových riešení by viedlo k tomu, že študenti by riešenia kopírovali navzájom od seba, z archívu alebo z iných zdrojov. Takéto podvádzanie by sme odhaľovali oveľa ťažšie, prípadne by sme ho ani neodhalili. Ešte náročnejšia na zvládnutie by bola komunikácia, ktorá po obvinení z podvádzania nezriedka nasledovala.

BIBLIOGRAFICKÉ ODKAZY

- [1] ARKORFUL, V., ABAIDOO, N. The role of e-learning, advantages and disadvantages of its adoption in higher education. In *International Journal of Education and Research*, 2014, vol. 12, no. 1, p. 397-410. ISSN 1550-6908.
- [2] NGO, M.. Eliminating Plagiarism in Programming Courses through Assessment Design. In *International Journal of Information and Education Technology*, 2016, vol. 6, no. 11, p. 873- 879. ISSN 2010-3689.
- [3] LEE, Y. J., et al. Plagiarism Detection among Source Codes using Adaptive Methods. In *KSSI transactions on internet and information systems*, 2012, vol. 6, no. 6, p. 1627-1648. ISSN 1976-7277.
- [4] GNIAZDOWSKI, Z., BONIECKI, M. Detection of a Source Code Plagiarism in a Student Programming Competition. In *Zeszyty Naukowe Warszawskiej Wyższej Szkoły Informatyki* 2019, vol. 21, p. 74-94. doi:10.26348/znwysi.21.74. ISSN 1896-396X.
- [5] DURACIK, M., HRKUT, P., KRSAK, E., TOTH, S. Abstract Syntax Tree Based Source Code Antiplagiarism System for Large Projects Set. In *IEEE Access*, 2020, vol. 8, p. 175347-175359, doi:10.1109/ACCESS.2020.3026422.
- [6] HAGE, J., RADEMAKER, P., VAN VUGT, N. A. A comparison of plagiarism detection tools. Utrecht : Utrecht University, 2010. ISSN 0924-3275.
- [7] LUKE, D., et al. Software Plagiarism Detection Techniques : A Comparative Study. In *International Journal of Computer Science and Information Technologies*, 2014, vol. 5, p. 5020-5024. ISSN 0975-9646.
- [8] WISE, M, String Similarity via Greedy String Tiling and Running Karp–Rabin Matching. Unpublished Basser Department of Computer Science Report, University of Sydney, 1993.
- [9] RIZZO, S., BERTINI, F., MONTESI, D, Fine-grain watermarking for intellectual property protection. In *EURASIP Journal on Information Security*, 2019. ISSN 2510-523X.

Terminologická príprava budúcich edukátorov

Terminological training of future educators

Ján Stoffa

Pedagogická fakulta Trnavskej univerzity v Trnave
Priemyselná 4, P. O. BOX 9
918 43 Trnava
Slovenská republika
StoffaJan@seznam.cz

Veronika Stoffová

Pedagogická fakulta Trnavskej univerzity v Trnave
Priemyselná 4, P. O. BOX 9
918 43 Trnava
Slovenská republika
veronika.stoffova@truni.sk

Abstract

The study deals with the issues of training future educators during their university studies and after their completion in the process of lifelong learning. It characterizes the position of the educator in relation to the terminology and the terminology of his own professional specialization and the existing terminological relations to the basic and related scientific disciplines, as well as school subject relations. It points out that although the educator is usually only a user of terminology, he is required to have the appropriate terminological competence, terminological culture and terminological discipline. Terminological competence consists primarily in the acquisition of the principles and basic knowledge of terminological science which form the methodological basis for solving terminological problems in practice. Terminological culture consists mainly in mastering the issue of literacy of terms used and in the identification of non-terms, resp. colloquial and substandard equivalents of terms. Terminological discipline consists primarily in respecting existing international and state standards, as well as narrow, respectively practices of the relevant professional community. Acquiring the appropriate level in the above three areas is a basic prerequisite for the successful solution of terminological problems in the educational process and in lifelong self-education after graduation.

Keywords

Terminological Preparation of Future Educators. Terminological Competence of the Educator. Terminological Culture of the Educator. Terminological Discipline of the Educator.

Abstrakt

Štúdia sa zaoberá otázkami prípravy budúcich edukátorov počas ich vysokoškolského štúdia a po jeho ukončení v procese celoživotného vzdelávania. Charakterizuje postavenie edukátora vo vzťahu k terminovede a pojmosloviu jeho vlastnej odbornej špecializácie a existujúce terminologické vzťahy k základným a príbuzným vedným odborom a tiež medzipredmetové vzťahy. Poukazuje na to, že hoci edukátor je spravidla len používateľom pojmoslovia, vyžaduje sa od neho náležitá terminologická kompetencia, terminologická kultúra a terminologická disciplína. Terminologická kompetencia spočíva predovšetkým v osvojení princípov a základných poznatkov terminovedy, ktoré tvoria metodologické východisko k riešeniu terminologických problémov v praxi. Terminologická kultúra spočíva predovšetkým v osvojení problematiky spisovnosti používaných termínov a v identifikácii netermínov, resp. hovorových a subštandardných ekvivalentov termínov. Terminologická disciplína spočíva predovšetkým v rešpektovaní existujúcich medzinárodných a štátnych noriem a tiež úzu, resp. zvyklostí príslušnej odbornej komunity. Získanie náležitej úrovne v uvedených troch oblastiach je základným

predpokladom úspešného riešenia pojmoslovných problémov v edukačnom procese a v celoživotnom sebazvzdelávaní edukátora po ukončení vysokoškolského štúdia.

Kľúčové slová

Terminologická príprava budúcich edukátorov. terminologická kompetencia edukátora. Terminologická kultúra edukátora. Terminologická disciplína edukátora.

1 ÚVOD

Okrem niektorých výnimiek sa ľubovoľná edukácia sústreďuje na osvojenie určitej množiny pojmov, ktoré tvoria ucelený systém daného učebného predmetu. Osvojenie týchto pojmov riadi kvalifikovaný edukátor s príslušnou aprobáciou. Pojmy a ich pomenovania – termíny sú preto celoživotným údelom každého edukátora [1]. Termín *edukátor* je významovo širší a zahŕňa kohokoľvek, kto sa podieľa na edukácii, t. j. nielen učiteľov profesionálov. Dôležitosť osvojovania termínov vyjadruje aj Štátny vzdelávací program ako aj vzdelávacie štandardy, kde v úvode každého tematického celku v požiadavkách na poznatky edukantov je uvedená požiadavka osvojovania základných termínov – budovania pojmoslovného systému. Zvlášť dôležité je to v oblasti informatiky, kde nárast nových pojmov a termínov exponenciálne rastie a časový interval zdvojnásobenia poznatkov je medzi existujúcimi vednými odbormi najkratší. Pojmoslovný systém je zvlášť dôležitý v dištančnom vzdelávaní a samovzdelávaní, keď získavanie nových poznatkov je založené hlavne na čítaní textu s porozumením. Čítanie textu s porozumením, často označované ako čitateľská gramotnosť mnohí považujú za nadpredmetovú kompetenciu. To, že v oblasti čítania odborného textu slovenskí žiaci základných a stredných majú veľké rezervy a že máme, čo doháňať, ukázali aj výsledky výskumu, ktorý bol realizovaný v roku 2018. (PISA 2018: Insights and Interpretations [32]). Výsledky v grafickom a číselnom vyjadrení sú na obr. 1, kde sme vybrali z úplnej tabuľky len začiatok, určitý úsek, kde je zaradené aj Slovensko a tiež koniec tabuľky. Výsledky nie sú uspokojivé, hoci sme sa na testovanie pripravovali, bola vypracovaná Ministerstvom školstva vedy, výskumu a športu Slovenskej republiky Národná stratégia zvyšovania úrovne a kontinuálneho rozvíjania čitateľskej gramotnosti, ktorá mala napomôcť dosiahnuť dobré výsledky [33].

Osvojovanie pojmov a termínov má aj značný poznávací potenciál [2] [3] a rad pozoruhodných aspektov [4].

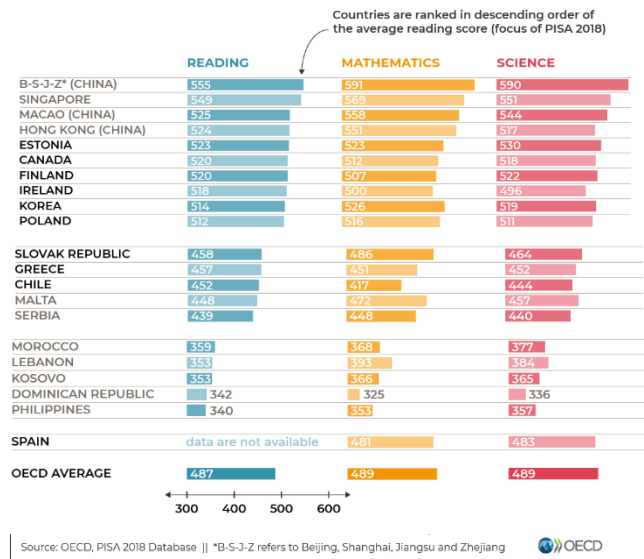
Osvojenie pojmu zahŕňa súčasne osvojenie jeho obsahu a komunikačného ekvivalentu pojmu, ktorým je jeho pomenovanie – termín.

Obsah pojmu prezentuje edukátor vo forme didakticky transformovanej definície. Použitú definíciu nemusí pochopiť

každý edukant, preto edukátor musí mať v zásobe niekoľko ďalších definícií, ktoré využíva dovtedy, kým obsah pojmu nepochopia aj mentálne najmenej disponovaní edukanti.

PISA 2018 results

Snapshot of students' performance in reading, mathematics and science



Obrázok 1: Výsledky PISA 2018

Didakticky transformované definície edukátor často čerpa hotové z existujúcich učebných pomôcok. Pri ich absencii, alebo v prípade nových pojmov ich vytvára sám na základe definícií publikovaných v odborných zdrojoch informácie. Tieto definície sú spravidla určené kvalifikovaným odborníkom a často obsahujú pojmy, ktoré edukanti nepoznajú.

Didakticky transformovaná definícia nemusí byť vedecky exaktná. Je vždy do istej miery zjednodušená a používajú sa v nej len pojmy, ktoré si už edukanti osvojili, buď v rámci daného učebného predmetu alebo v rámci niektorého z predmetov, s ktorými má daný predmet medzipredmetové vzťahy. Prirodzenými požiadavkami na didakticky transformovanú definíciu sú logická správnosť, systémovosť vyjadrená pomocou najbližšieho nadsadeného pojmu, spisovnosť a použitie správnych termínov pre definovaný pojem aj pre všetky pojmy použité v texte definície.

Výber jednotlivých termínov môže pre edukátora predstavovať problém. Platí to predovšetkým pre nové termíny – neologizmy. V ich prípade môžu nastať ťažkosti spojené so získaním informácií o obsahu nových pojmov, ktoré treba nezriedka čerpať z cudzojazyčných informačných zdrojov, napr. z [5] a [6]. Navyše edukátor môže naraziť na nedostatok informácií o správnom pravopise a výslovnosti nových termínov.

Problémy však môže mať edukátor aj s ustálenými termínmi a z nich najmä pri výbere optimálneho termínu z množiny synonymných termínov a tiež s termínmi homonymného charakteru, ktoré môžu interferovať s ich už osvojenými náprotivkami z iných učebných predmetov.

Problémy môže mať edukátor aj so zastaranými termínmi.

Osobitnú skupinu problémov tvoria tie, ktoré súvisia s imperatívnymi požiadavkami na spisovnosť komunikácie v edukačných procesoch. Ide najmä o identifikáciu netermínov,

poloterminov a nenáležité použitie hovorových alebo subštandardných ekvivalentov termínov.

Ďalšiu osobitnú skupinu problémov predstavujú skráteniny a neverbálne prvky, rôzneho druhu (skratky, značky, symboly, ideogramy a pod.), ktoré sa v praxi často používajú ako ekvivalenty termínov.

Cieľom tejto štúdie je ukázať, že úspešné riešenie uvedených problémov vyžaduje, aby si budúci edukátori už počas vysokoškolského štúdia osvojili základy terminovedy, pričom za optimálne považujeme zaradenie osobitného študijného predmetu do študijného programu a osvojenie terminologického minima edukátora [7].

2 Postavenie edukátora k terminovede a pojmosloviu

Edukátor má k terminovede a pojmosloviu špecifické postavenie. Jadro jeho špecifickosti vyplýva z toho, že:

- Okrem málo výnimiek, je edukátor prakticky vždy len používateľom pojmoslovia;
- Okrem niekoľkých odborov a vysokoškolských pracovísk nedostáva v rámci svojho štúdia žiadnu prípravu z terminovedy;
- Pri použití termínov ako verbálnych ekvivalentov pojmov má edukátor postavenie verejného činiteľa a zákon o štátnom jazyku ho zaväzuje realizovať komunikáciu s edukantami v spisovnom jazyku;
- Pri výbere jazykových prostriedkov v komunikácii je edukátor viazaný odborným štýlom vyjadrovania, čo znamená, že sa vyhýba používaniu viacerých špecifických slov a výrazov, napr. hovorových, citovo zafarbených, expresívnych, subštandardných a pod.;
- Poznatky získané z odborných informačných zdrojov edukátor spravidla nemôže prezentovať vo forme ako ich získal, ale musí ich didakticky transformovať, t. j. uviesť do formy, v ktorej si ich edukanti budú osvojovať. Táto požiadavka je osobitne významná v rámci všeobecného vzdelávania, pri ktorom sa edukátor musí vyhýbať nadmernému používaniu cudzích a významovo nepriezračných termínov a výrazov;
- Keďže väčšina didaktických prezentácií sa realizuje v hovorovej forme, od edukátora sa vyžaduje nadštandardná úroveň výslovnosti.
- Ak sa edukátor vo výučbe dopustí terminologických chýb a nedostatkov, ktoré si edukanti osvoja, **počet ich nositeľov sa znásobuje počtom edukantov**. Navyše nesprávne osvojené poznatky je ťažšie napraviť ako ich získať "z prvej ruky".
- V mnohých prípadoch je edukátor viazaný nielen požiadavkami na spisovnosť komunikácie, ale aj tým, že istá časť termínov je v každom odbore štandardizovaná a teda pre používateľov záväzná.
- Aj v prípade termínov, ktoré nie sú štandardizované, má edukátor rešpektovať úzus danej odbornej komunity, t. j. nepísané pravidlá a zvyklosti.
- V dôsledku stále intenzívnejšej internacionalizácie pojmoslovia edukátor čelí prílivu cudzojazyčných termínov. V súvislosti s tým sa kladú väčšie nároky na znalosti cudzích jazykov a na sledovanie adaptačných procesov pri preberaní cudzojazyčných termínov.

Je naivné predpokladať, že edukátori si osvoja pojmoslovie svojho odboru/svojich odborov bez osobitnej prípravy v oblasti terminovedy. V dôsledku žiadnej alebo nedostatočnej terminologickej prípravy sa používatelia dopúšťajú v praxi rôznych chýb a nedostatkov, a často ich aj rozmnožujú. Na niektoré z nich sme poukázali v osobitných štúdiách, napr. v [8] a [9] a tiež v monografii [10, s. 101 – 121].

3 Terminologická gramotnosť edukátora

Aby edukátor mohol úspešne riešiť všetky problémy spojené s osvojovaním obsahov pojmov a ich pomenovaniami – termínmi, musí si osvojiť istú minimálnu úroveň poznatkov z terminovedy a získať tým potrebnú terminologickú gramotnosť. Obsah nami zavedeného pojmu terminologická gramotnosť sme postupne vymedzovali a spresňovali v štúdiách [11] až [14] a v monografii [10, s. 23 – 25], preto ich na tomto mieste nebudeme opakovať. Sústreďme sa len na tie zložky terminologickej gramotnosti, ktoré predstavujú pre edukátorov najväčšie problémy, alebo pri ktorých sa najčastejšie dopúšťajú chýb. Sú to najčastejšie tieto prípady:

- Neznalosť základných poznatkov terminovedy spôsobuje výskyt mnohých elementárnych terminologických chýb. Najčastejšie je to nenáležité stotožňovanie pojmu a termínu, a tiež termínu a názvu, terminológie a názvoslovia.
- Pri definovaní pojmov edukátori často nerešpektujú systémové hierarchické vzťahy nadradenosti a podradenosti a definovaný pojem nevymedzujú pomocou najbližšieho hierarchicky nadradeného pojmu. Tým, že neuplatňujú systémový prístup oslabujú systémové poňatie pojmoslovia v svojej oblasti a neprezentujú pojem ako súčasť uceleného systému poznatkov.
- Edukátori nevenujú náležitú pozornosť terminologickým vzťahom svojho odboru a učebného predmetu s príbuznými odbornými a predmetmi. Opätovne a nie vždy rovnako prezentujú pojmy a termíny, ktoré už majú edukanti osvojené z príbuzných predmetov.
- Edukátori nie vždy poukazujú na existenciu homonymných termínov, ktoré môžu robiť edukantom problémy. Ako príklad možno uviesť termín *prímes*, ktorý je v chémii a príbuzných odboroch pomenovaním nežiadúcej zložky niektorej látky, zatiaľ čo vo fyzike, elektronike a elektrotechnike je pomenovaním žiadúcej, zámerne pridávanej látky do východiskovej látky za účelom riadenia jej vlastností.
- V súčasnosti sa v pojmosloví každého odboru a učebného predmetu vyskytuje mnoho synonymných termínov a neexistujú informácie zdroje, z ktorých by edukátor spoľahlivo zistil koľko synonymných ekvivalentov má daný termín a aká je ich správna pravopisná forma a výslovnosť. Existujúce slovníky synonymom pokrývajú pojmoslovie prírodných a technických odborov len sčasti a na rozdiel od terminovedy považujú za synonymá aj slová blízke svojím významom, t. j. aj nie úplne ekvivalentné. Mnohé synonymné termíny predstavujú cudzie slová a ich osvojenie predpokladá predchádzajúce vysvetlenie ich obsahu.
- V dôsledku absencie potrebného teoretického minima z terminovedy mnohí edukátori nie sú spôsobilí uplatňovať kritický prístup pri výbere termínu na osvojovanie. Výber je často náhodný, len na základe jedného informačného

zdroja a nezriedka aj nesprávny. Je prekvapujúce ako často sa ešte aj v súčasnosti stretávame s termínmi *umelá hmota*, *umelohmotný* a ich odvodeninami napriek tomu, že tieto termíny boli z pojmoslovných systémov radu odborov vyradené už pred desiatkami rokov. Ich použitie je nielen hrubou terminologickou chybou, ale aj hrubou vecnou chybou narušujúcou medzipredmetové vzťahy s fyzikou, v ktorej sa edukanti učia, že umelá hmota neexistuje a že hmota má nielen látkové formy.

- Edukátori, ktorí sa snažia edukačný proces inovovať a zoznamovať svojich zverencov aj s najnovšími poznatkami z oblasti vedy a techniky narážajú na problém nových termínov – neologizmov. Slovníky, ktoré by reflektovali neologizmy z oblasti prírodných a technických vied zatiaľ v SR neboli vytvorené. Edukátorom odporúčame priebežne si zostavovať vlastné slovníky podľa vzoru českých slovníkov neologizmov [5] a [6].
- Existuje veľké množstvo zdrojov, ktoré obsahujú terminologické informácie. Tieto zdroje však nie sú rovnocenné, viaceré vecne alebo morálne zastarali a novšie, najmä novovydané slovníkové diela nemajú vždy potrebnú kvalitu. Napr. v edukačne orientovanom slovníku [15] na jednu stranu textu pripadá v priemere niekoľko chýb. Správne využívanie jednotlivých zdrojov vždy vyžaduje uplatnenie kritického prístupu a overovanie správnosti na základe niekoľkých nezávislých a spoľahlivých informačných zdrojov.
- Spomedzi najdôležitejších vlastností termínov sa v terminovede venuje najviac pozornosti ich jednoznačnosti, ustálenosti a systémovosti. V edukácii však má nemenej dôležité postavenie vlastnosť, pre ktorú sme zaviedli termín *osvojiteľnosť*. Je to vlastnosť integrálneho charakteru, ktorú sme bližšie opísali v štúdiu [16] a v monografii [10, s. 62, 81 – 82]. Iba porovnanie vlastností posudzovaného termínu s vlastnosťami ideálneho termínu umožňuje optimálny výber termínu na osvojovanie v edukačnom procese. Pritom sa musí prihliadať na vekové a mentálne osobitosti edukantov, tak pri výbere termínu ako aj pri didaktickej transformácii definície termínom pomenovaného pojmu.
- Špecifikou prírodných a technických vied je fakt, že v mnohých prípadoch je výhodnejšie použiť namiesto termínov, ktoré sú vždy verbálneho charakteru, ich neverbálne ekvivalenty. Najčastejšie sú to rôzne skrátenejšie a symboly. Edukátori musia zvládnuť a **vždy** formou vhodnej legendy uvádzať správne priradenie príslušných termínov k jednotlivým neverbálnym ekvivalentom. Nesmú pritom obchádzať fakt, že mnohé z týchto ekvivalentov majú homonymný charakter.

4 Terminologická kultúra edukátora

Postavenie edukátora ako verejného činiteľa vyžaduje, aby sa komunikácia s edukantmi realizovala na vysokej kultúrnej úrovni. Terminologická kultúra edukátora je zložitý pojem, ktorého obsah a zložky v prípade infoedukácie sme prezentovali vo viacerých štúdiách, napr. [17]) až [22] a v monografii [10, s. 25 – 29]. Aj v tejto oblasti sme zaznamenali početné nedostatky a chyby. Ich objektívnou príčinou je fakt, že v rámci vysokoškolských študijných programov absentuje predmet zameraný na jazykovú stránku odbornej komunikácie. Zrejme sa predpokladá, že edukanti vystačia z poznatkami získanými z príslušného jazyka počas štúdia

na strednej škole. Výnimku z tohto stavu tvoria niektoré vysoké školy. Krátkodobu alebo v dlhodobu sa predmet odborná terminológia vyučoval alebo vyučuje na UKF v Nitre, na Prešovskej univerzite, na UJS v Komárne, na UP v Olomouci a na TU v Trnave, pričom o získaných skúsenostiach sme verejnosť informovali vo viacerých štúdiách, napr. [23].

Subjektívnou príčinou chýb a nedostatkov je, že mnohí edukátori jazykovú stránku komunikácie podceňujú, ďalšie vzdelávanie v oblasti jazyka nerealizujú a vývoj jazyka, v ktorom neustále prebiehajú rôzne zmeny, nesledujú. Medzi najčastejšie dôsledky tohto stavu patrí časté používanie nespisovných, subštandardných, štylisticky nevhodných jazykových prostriedkov a nenáležité preberanie cudzojazyčných termínov.

V prípade termínov je ich základnou vlastnosťou to, že majú verbálnu, t. j. slovnú povahu a sú preto organickou súčasťou slovného zásoby príslušného jazyka. Keďže jazyk je hlavnou zložkou identity každého národa, každé spoločenstvo prikladá jeho ochrane a jeho výučbe v rámci systému všeobecného vzdelávania veľkú pozornosť. Termíny tvoriace pojmoslovie každého odboru majú teda nielen verbálnu, ale aj národný charakter. Na tom nič nemení fakt, že v každom pojmosloví je určité množstvo termínov prevzatých z iných jazykov, identických alebo, väčšinou, adaptovaných. Spôsob prevzatia cudzojazyčných termínov je plne v kompetencii príslušného národného spoločenstva. Často je spoločný len základ a iné zložky termínu sú už odlišné. Názorne to možno preukázať na pojmosloví termínu:

term (sq, en), *terme* (ca, fr), *termen* (ro), *termi* (fi), *termijn* (nl), *termin* (be, bg, et, hr, mk, ru, sr, pl, sv, uk), *termín* (sk, cs), *terminas* (lit), *termine* (it), *termino* (es), *terminon* (eo), *terminu* (mt), *terminus* (la). Pomenovanie termín nebolo prevzaté do gréčtiny, islandčiny a írčiny, ktoré používajú v uvedenom poradí pomenovania *ορολογία*, *hugtok* a *tearmaiocht*. (Skratky jazykov uvedené v zátvorkách sú v súlade s ISO 639-1:2002 Language Code List. In: https://www.loc.gov/standards/iso639-2/php/code_list.php (Cit. 7.5.2020))

Z hľadiska terminologickej kultúry možno za najvýznamnejšie problémy v slovenskom pojmosloví považovať:

- Správnu identifikáciu pojmu ako pomenovania abstraktného pojmu v konkrétnom texte prezentácie a jeho odlišenie od názvov (ako pomenovaní konkrétnych objektov), poloterminov a neterminov. Napr. z pomenovaní silon/silón, dederon/dederón a *polyamid*, sú prvé názvami, a len posledné je termín.
- Odlišenie bezpríznakových, t. j. štylisticky neutrálnych výrazov od ostatných, najmä hovorových, citovo zafarbených, subštandardných a pod. Za najkrikľavejší nedostatok z tejto kategórie je treba považovať použitie slangových výrazov, napr. použitie výrazov antiviráč, diplomka, počítač, kraťas, nulák, písíčko namiesto ich správnych ekvivalentov *antivírusový program*, *diplomová práca*, *počítač*, *skrat*, *nulový vodič*, *osobný počítač*.
- Zistenie pravopisu a výslovnosti nových termínov – neologizmov. Napr. v prípade neologizmu *storyboard* (= grafická podoba scenára) používateľ v bežných zdrojoch nenájde informáciu, či sa má používať v pôvodnej pravopisnej forme, alebo v adaptovanej forme *storybord*, resp. *storybórd* a či sa má vyslovovať *storybord*, *storybórd*, *stórybord* alebo *stórybórd*.

- Overenie ekvivalencie dvoj- a viactvarov termínov a iných výrazov a ich optimálne ohodnotenie z didaktického hľadiska. Napr. v prípade termínu pre vernosť akustickej reprodukcie nájde v [24, s. 460] štyri rovnocenné varianty termínu: *hi-fi*, *Hifi*, *Hi-Fi*, *Hi-fi*. Podľa [25, s. 80]) spisovné sú len dva varianty: *hi-fi* a *hifi*.
- Identifikáciu tých cudzojazyčných termínov a výrazov, ktoré neboli prevzaté do príslušného jazyka v pôvodnej alebo adaptovanej forme, v dôsledku čoho ich treba považovať za nespisovné. Osobitný problém v slovenskom kontexte predstavujú neprevzaté bohemizmy a pseudobohemizmy. Ako príklady často a nenáležite používaných bohemizmov možno uviesť *čidlo*, *doprovod*, *obor*, *pájka*, *tlačítka*, *výuka* namiesto ich správnych slovenských ekvivalentov *snímač*, *sprievod*, *odbor*, *spájka*, *tlačidlo*, *výučba*. Ako príklady často a nenáležite používaných pseudobohemizmov možno uviesť *detekovať*, *dieľič*, *doporučiť*, *nápoved*, *odozva* (správne *detegovať*, *čiasťkový*, *odporučiť*, *našepkávanie*, *ohlas*).
- Zistenie správnej výslovnosti použitých termínov a iných výrazov, vrátane skrátení. Napr. pôvodom anglická skratka IT sa v slovenskom kontexte často hláskuje „aj tí“. Správna výslovnosť je „í té“ [25, s. 419].
- Nekritické a didakticky pochybné uprednostňovanie cudzojazyčných termínov a iných výrazov, pre ktoré existujú ich slovenské, významovo priezračnejšie náprotivky, resp. ekvivalenty. Napr. namiesto termínov *joystick/džojstik*, *kolor*, *komponent* sú didakticky vhodnejšie slovenské termíny *pákový ovládač*, *farebný tón/farebnosť*, *zložka/časť*.
- Identifikácia nesprávne preložených cudzojazyčných termínov a interhomonym (medzijazykových homonym). Napr. anglické termíny *billion*, *carbon steel*, *technology* sú často prekladané ako bilión, karbónová oceľ, technológia (správne *miliarda*, *uhlíková oceľ*, *technika*). V angličtine *control* znamená riadenie, v slovenčine kontrolu, resp. preverovanie. V maďarčine *kar* je pomenovanie katedry, v slovenčine pohrebnej hostiny. V poľštine *odmiana* znamená variant, alternatívu, v slovenčine hmotnú odplatu.

5 Terminologická disciplína edukátora

Z postavenia edukátora ako používateľa pojmoslovie vyplýva, že pri používaní termínov musí dodržiavať terminologickú disciplínu. Obsah pojmu terminologická disciplína sme bližšie charakterizovali v štúdiu [26].

Príslušné termíny musí edukátor používať v súlade s existujúcimi štátnymi a odborovými normami/štandardmi. Za cenný zdroj informácií o normalizovaných termínoch považujeme v súčasnosti aj už neplatné normy, ktoré vznikli ešte počas existencie spoločného štátu Čechov a Slovákov a sú označené skratkou ČSN (= Československá štátna norma). Takýchto noriem existuje rádovo niekoľko stoviek. Väčšina z nich má v názve mätúce pomenovanie „názvoslovie“ a namiesto pomenovania *termín* často používajú nenáležité pomenovanie *názov*. V skutočnosti však v ich obsahu vysoko prevažujú nie názvy, ale termíny. Napriek ich zdanlivej zastaranosti a neplatnosti ich využívanie odporúčame z týchto dôvodov:

1. V dôsledku platnosti princípu ustálenosti sú takmer všetky v nich obsiahnuté termíny platné aj v súčasnosti. Nie je však

na škodu poznať z týchto noriem aj zastarané alebo z pojmoslovia neskôr vyradené termíny a definície nimi pomenovaných pojmov.

2. Každý v nich uvedený termín má priradenú definíciu, ktorá môže byť po náležitej didaktickej transformácii využitá na edukačné účely.
3. Normy vydané v slovenčine prešli jazykovou kontrolou Jazykovedného ústavu Ľudovíta Štúra SAV, preto termíny a definície pojmov sú jazykovo správne a vyhovujú požiadavke spisovnosti. Veľká väčšina noriem je vydaná v češtine. Aj tie spravidla obsahujú slovenské ekvivalenty českých termínov. Slovenským používateľom sú české odborné texty všeobecne dobre zrozumiteľné, ale pri preklade českých termínov a výrazov treba uplatňovať kritický prístup.
4. Normy často obsahujú okrem českých a slovenských termínov aj ekvivalentné termíny v jazykoch početnejších spoločností, väčšinou v angličtine, francúzštine, nemčine a ruštine. Normy z čias existencie RVHP (Rady vzájomnej hospodárskej pomoci) obsahujú v niektorých prípadoch aj bulharské, maďarské, mongolské, poľské, rumunské, španielske a vietnamské termíny.
5. Normy obsahujú aj synonymné termíny.
6. Normy v mnohých prípadoch poukazujú na nesprávne termíny.

Terminologické informácie menšieho rozsahu môžu používatelia čerpať aj z ostatných (neterminologických) mnohé z nich v úvodnej časti obsahuje kľúčové termíny predmetnej oblasti a ich definície. Napr. ČSN 36 8210 Mikrofony obsahuje v úvodnej časti 54 termínov a definícií nimi pomenovaných pojmov.

Veľká časť pojmoslovia nie je pokrytá normami. V tomto prípade sa edukátori musia riadiť úzom, resp. zvyklosťami príslušnej odbornej komunity.

Z hľadiska terminologickej disciplíny by edukátori mali navyše rešpektovať tieto odporúčania:

- ❖ V prípade existencie viacerých synonymných termínov používať v edukácii dôsledne len ten, ktorý je optimálny z didaktického hľadiska. Na existenciu ostatných synonymných termínov treba upozorniť pri prvom výskyte optimálneho termínu.
- ❖ Rovnako treba postupovať pri existencii prípustných dvoj- a viactvarov termínov.
- ❖ V prípade viac-slovných termínov nemeniť svojvoľne poradie ich termínových prvkov ani ich nenahrádzať inými prvkami s rovnakým významom.
- ❖ V prípade homonymných termínov alebo skrátení vždy vysvetliť ich význam pri prvom použití, buď priamo v texte, alebo v poznámke pod čiarou na príslušnej strane textu.
- ❖ Nepoužívať v prezentácii pracovné alebo nedefinované skráteniny.
- ❖ V prípade termínov s dvojakou správnou výslovnosťou upozorňovať edukantov na túto skutočnosť, avšak používať len jeden variant výslovnosti.

- ❖ Nepoužívať termíny bez predchádzajúceho overenia ich spisovnosti na základe kodifikačných zdrojov príslušného jazyka. Pre slovenčinu sú to zdroje uvedené v zozname bibliografických odkazov pod číslami [25], [27] až [31].

6 Terminológia v procese celoživotného vzdelávania učiteľa

Ani terminologické minimum, resp. terminologická gramotnosť edukátora získaná počas vysokoškolského štúdia edukátorovi nepostačuje na celé obdobie jeho pôsobenia. Získané poznatky z oblasti terminovedy nestačí len uchovať a riadiť sa nimi v každodennej edukačnej praxi, ale edukátor musí zvyšovať svoju terminologickú erudíciu, najmä z nasledovných dôvodov:

- V dôsledku informačnej explózie sa neustále zvyšuje počet nových pojmov a termínov. Každý edukátor by mal do edukačného procesu zaraďovať inovačné informácie, čo nie je možné bez prezentácie nových pojmov a termínov.
- Časť pojmov a termínov sa v dôsledku vedeckotechnického pokroku stáva neaktuálnou. Aj zastarané termíny môžu mať obmedzené využitie a niektoré v edukácii využiteľné aspekty, napr. etymologické a medzikultúrne aspekty.
- Napriek platnosti princípu ustálenosti sa niektoré termíny z rôznych dôvodov z pojmoslovných systémov vyradujú. Takto bol napr. vyradený termín *dielektrická konštanta*, keďže nejde o konštantu.
- Niektoré termíny sa spresňujú. Napr. termín *váha* bol mnoho rokov homonymný a mal dva významy, ktoré sú v súčasnosti pokryté termínmi *hmotnosť* a *ťaž*.
- V priebehu dlhšieho času sa môže zmeniť pravopis niektorých termínov. Pôvodný pravopis termínov hardware, pracovná zmena, sjazd, software sa zmenil, a ich súčasná spisovná forma je *hardvér*, *pracovná zmena*, *zjazd*, *softvér*.
- Z podobných dôvodov môže získať ustálený termín nový tvar. Napr. termín *e-mail* získal aj alternatívnu (synonymnú) formu *email*, ktorá je však homonymná (dva ďalšie významy). Obidve formy majú (na rozdiel od homonymných foriem) rovnakú výslovnosť imejl.

7 Záver a odporúčania

Na základe uvedených informácií a skúseností získaných počas niekoľkodesaťročnej edukačnej praxe na rôznych vysokoškolských pracoviskách doma aj v zahraničí možno konštatovať, že súčasná úroveň terminologickej gramotnosti, terminologickej kultúry a terminologickej disciplíny edukátorov nezodpovedá súčasným požiadavkám. Preto navrhujeme zaradiť do študijných programov na všetkých vysokých školách osobitný študijný predmet zameraný na vedecko-odbornú komunikáciu a terminológiu. Navrhujeme tiež, aby sa pri posudzovaní kvalifikačných prác povinne hodnotila aj ich terminologická správnosť a jazyková kultúra.

Pod'akovanie

Príspevok bol podporený projektom KEGA 012TTU-4/2018: Interaktívne animačné-simulačné modely vo vzdelávaní.

Zoznam bibliografických odkazov

- [1] STOFFA, J. – STOFFOVÁ, V.: Terminológia – celoživotný údel každého edukátora. In: *Eruditio – Educatio*, roč. 6, 2011, č. 1, s. 3 – 14. ISSN 1336-8893
- [2] STOFFA, J. – STOFFOVÁ, V.: Hodnoty terminológie v doplnkovom pedagogickom štúdiu. In: *Schola 2004*. Zost. A. Hambalík a K. Krelová. 1. vyd. Bratislava : Slovenská technická univerzita v Bratislave, 2004, s. 255 – 260. ISBN 80-227-2143-3
- [3] STOFFA, J.: Cognitive values of terminology in education. In: *Proceedings of XXVIII. DidMatTech 2020 : New Methods and Technologies in Education, Research and Practice*. Ed. A. Abonyi-Toth, V. Stoffová a L. Zsako. Budapest : Eötvös Lorand University in Budapest – Trnava University in Trnava, 2020, s. 119 – 126. ISBN 978-963-489-244-1
- [4] STOFFA, J. – STOFFOVÁ, V.: Special aspects of scientific terms. In: *XXIXth DIDMATTECH 2016 : New methods and technologies in education and practice : 1st Part*. Ed. V. Stoffová, L. Szakó a P. Szlávi. 1. vyd. Budapest : Eötvös Loránd University in Budapest, Faculty of Informatics, 2016, s. 275 – 279. ISBN 978-963-799-3
- [5] MARTINCOVÁ, O. a kol.: *Nová slova v češtině : Slovník neologizmů*. 1. vyd. Praha Academia, 1998. 357 s. ISBN 80-200-0640-0
- [6] MARTINCOVÁ, O. a kol.: *Nová slova v češtině : Slovník neologizmů 2*. 1. vyd. Praha Academia, 2004. 568 s. ISBN 80-200-1168-4
- [7] STOFFA, J. – STOFFOVÁ, V.: Terminologické minimum edukátorov. In: *Informatika v škole a v praxi*. Ed. I. Černák, E. Mašek a J. Majherová. 1. vyd. Ružomberok : Pedagogická fakulta Katolíckej univerzity v Ružomberku, Katedra informatiky, 2008, s. 98 – 106. ISBN 978-80-8084-362-5
- [8] STOFFA, J. – STOFFOVÁ, V.: Časté terminologické chyby v oblasti informatiky a informačných a komunikačných technológií. In: *Informatika v škole a v praxi : 3. ročník konferencie s medzinárodnou účasťou*. 1. vyd. Ružomberok : Pedagogická fakulta, Katolícka univerzita v Ružomberku, 2007. ISBN 978-80-8084-222-2 [CD-ROM]
- [9] STOFFA, J. – STOFFOVÁ, V.: O jazykovej a terminologickej kultúre odborných prezentácií v oblasti informatiky a informačných a komunikačných technológií. In: *INFOTECH 2007 Moderní informační a komunikační technologie ve vzdělávání : Díl 2*. Ed. J. Dostál. 1. vyd. Olomouc : Votobia Olomouc, 2007, s. 524 – 527. ISBN 978-80-7220-301-7
- [10] STOFFA, J. – STOFFOVÁ, V.: *Terminológia informatiky a IKT*. 1. vyd. Trnava : TYPI Universitas Tynaviensis, 2017. 251 s. ISBN 978-80-568-0065-2
- [11] STOFFA, J.: Terminological literacy of young scientist in the branch of instructional technology. In: *Technology of Education : Volume 1 : Technológia vzdelávania : Zväzok 1*. 1. vyd. Nitra : Slovidiac, 1998, s. 138 – 148. ISBN 80-967746-1-1 (tamže je štúdia aj v slovenskom znení).
- [12] STOFFA, J.: Terminologická gramotnosť a terminologická kultúra. In: *Terminologické fórum II : na tému socioterminológia, textová a prekladová terminológia*. Ed. I. Cibíková. 1. vyd. Trenčín : Trenčianska univerzita A. Dubčeka, 2009. [CD-ROM] ISBN 978-80-8075-375-7
- [13] STOFFA, J. – STOFFOVÁ, V.: Teacher's terminological competence. In: *Edukacja wczoraj – dziś – jutro Edukacja w dialogu pokoleń i budowaniu lepszej przyszłości*. Red. E. Sałata, M. Mazur a J. Bojanowicz. 1. vyd. Radom : Radomskie Towarzystwo Naukowe, 2015, s. 457 – 464. ISBN 978-83-88100-38-3
- [14] STOFFA, J. – STOFFOVÁ, V.: A new component of teacher's terminological competence. In: *Przygotowanie nauczycieli do nowych wyzwań edukacyjnych : Problemy współczesnej edukacji*. Red. J. Bojanowicz a K. Ziębakowska-Cecot. 1. vyd. Radom : Uniwersytet Technologiczno-Humanistyczny im. Kazimierza Pułaskiego w Radomiu, 2018, s. 204 – 297. ISSN 1642-5271, ISBN 978-83-7351-861-2
- [15] ĎURIŠ, M.: *Krátky technický výkladový a naučný slovník : pre učiteľov technických odborných predmetov*. 1. vyd. Banská Bystrica : Univerzita Mateja Bela, Fakulta prírodných vied, 2004. 188 s. ISBN 80-8055-918-X
- [16] STOFFA, J.: Osvojiteľnosť termínu. *XVIII. Mezinárodní kolokvium o řízení osvojovacího procesu : Sborník příspěvků II*. Vyškov : Vysoká vojenská škola pozemního vojska ve Vyškově, 2000, s. 307 – 310. ISBN 80-7231-059-3
- [17] ŠTOFA, J.: Terminologická kultúra učiteľa technickej výchovy. In: *Současné celosvětové otázky alternativního školství...* Olomouc : Universitas Palackianae Olomucensis, Facultas paedagogica, 1992, s. 28 – 29. ISBN nemá
- [18] ŠTOFA, J.: K zvyšovaniu terminologickej kultúry učiteľov technickej výchovy. In: *Sborník z 2. konference o současných celosvětových otázkách alternativního školství*. 1. vyd. Olomouc : Universitas Palackianae Olomucensis, Facultas paedagogica, 1992, s. 202 – 204. ISBN nemá
- [19] STOFFA, J.: Terminologická kultúra ako súčasť jazykovej kultúry v systéme prípravy učiteľov technickej výchovy. In: *Cesty k tvořivé škole : Sborník příspěvků z konference k 50. výročí založení Pedagogické fakulty Masarykovy univerzity Brno* 13. – 14. listopadu 1996. 1. vyd. Brno : Masarykova univerzita, 1998, s. 248 – 252. ISBN 80-210-1938-7
- [20] STOFFA, J. – KROBOTOVÁ, M.: Jazyková a terminologická kultura učitele 21. století. In: *Připravujeme učitele pro 21. století a vstup do Evropy? : (Pregraduální a postgraduální příprava učitelů) : Sborník z konference*. 1. vyd. Olomouc : Univerzita Palackého, 1998, s. 169 – 173. ISBN 80-7067-871-2
- [21] STOFFA, J. – KROBOTOVÁ, M.: Jazyková a terminologická kultura při osvojovacím procesu. In: *XX. mezinárodní kolokvium o řízení osvojovacího procesu*. Vyškov : Vysoká vojenská škola pozemního vojska ve Vyškově, 2002, s. 202 – 206. ISBN 80-7231-090-9
- [22] STOFFA, J. – STOFFA, V.: Terminology and Language Culture in Modern Communication Technologies. *Technology for mobile society*. Ed. M. Muraskiewicz. Warsaw : MOST Press, 2003, s. 446 – 451. ISBN 83-87091-42-1
- [23] STOFFA, J. – STOFFOVÁ, V.: Terminology in the training of university teachers. In: *Kształcenie praktyczne nauczycieli w szkole wyższej*. Red. E. Sałata, A. Zamkowska a S. Oško. Ryki – Radom : Politechnika Radomska im. Kazimierza Pułaskiego..., 2003, s. 92 – 96. ISBN 83-7351-077-X

- [24] ŠALING, S. – IVANOVÁ-ŠALINGOVÁ, M. – MANÍKOVÁ, Z.: *Veľký slovník cudzích slov*. 4., revid. a dopl. vyd. Prešov : SAMO, 2006. 1392 s. ISBN 80-89123-05-8
- [25] *Slovník súčasného slovenského jazyka ag*. Red. K. Buzássyová a A. Jarošová. [Údaj o vyd. neuvedený] Bratislava : Veda, 2006. 1132 s. ISBN 80-224-0932-4
- [26] STOFFA, J. – STOFFOVÁ, V.: Terminological discipline and terminological creativity of the techno-educator. In: *Journal of Technology and Information Education*, 2/2018, vol. 10, Issue 2, s. 100 – 107. ISSN 1803-537X
- [27] *Pravidlá slovenského pravopisu*. Red. M. Považaj. 4., nezmenené vyd. Bratislava : Veda 2013. 592 s. ISBN 978-80-224-1331-2
- [28] *Krátky slovník slovenského jazyka*. 4., doplnené a upravené vyd. Bratislava : Veda, 2003. 985 s. ISBN 80-224-0750-X
- [29] *Slovník súčasného slovenského jazyka H – L*. [Údaj o vyd. neuvedený] Ed. A. Jarošová a K. Buzássyová. Bratislava : Veda, 2011. 1088 s. ISBN 978-80-224-1172-1
- [30] *Slovník súčasného slovenského jazyka M – N*. Ed. A. Jarošová. [Údaj o vyd. neuvedený] Bratislava : Veda, 2015. 1104 s. ISBN 978-80-224-1485-2
- [31] KRÁĽ, Á.: *Pravidlá slovenskej výslovnosti*. 3. vyd. Martin : Matica slovenská, 2016. 423 s. ISBN 978-80-8128-159-4
- [32] <https://www.oecd.org/pisa/publications/pisa-2018-results.htm> (cit.24.2.2021)
- [33] http://zsss.stranka.info/stranky/zsss/F/subory/cg_narodnastrategia.pdf (cit.24.2.2021)

Synchronní distanční výuka informatiky s využitím problémově orientovaných úloh

Synchronous online lessons of Informatics with the usage of problem-based tasks

Václav Šimandl
KIN PF JU
Jeronýmova 10
371 15 České Budějovice
Česká republika
simandl@pf.jcu.cz

ABSTRACT

The paper looks at learning to work in the Linux operating system in a group of pre-graduate students of Informatics during the COVID-19 pandemic. Lessons were based on the principles of problem based learning while using a set of tasks with increasing difficulty. The synchronous distance lessons were chosen to secure a teacher-student interaction. During these lessons it was required from the teacher to continuously watch students' screens and by doing so, help with a solution for tasks if necessary. For this purpose, the open source Jitsi Meet platform was installed on the faculty server which allows videoconferences to share multiple screens simultaneously. Instructions for students were inserted into LMS Moodle to secure the individualization of lessons. Forums were also created there where students could submit their answers to certain tasks. In the paper we present our findings from the evaluation of such lessons, both from the teacher's and students' perspective. The limit for such lessons is the quality of the Internet access of students and the restricted opportunity for communication among students.

Keywords

Informatics. Linux. Problem based learning. Distance education. Synchronous lessons. LMS Moodle. Jitsi Meet.

ABSTRAKT

Článek se zabývá výukou práce v operačním systému Linux na skupině pregraduálních studentů informatiky během pandemie COVID-19. Výuka byla založena na principech problémového vyučování a byly během ní využity sady gradovaných úloh. Pro zabezpečení interakce mezi učitelem a studenty byla zvolena synchronní distanční výuka. Při ní bylo požadováno, aby učitel mohl průběžně sledovat obrazovky studentů a dle potřeby jim díky tomu s řešením úkolů pomoci. Pro tento účel byla na fakultní server nainstalována open source platforma Jitsi Meet, která umožňuje videokonference se sdílením více obrazovek současně. Pro zajištění individualizace výuky byly studentům do LMS Moodle vloženy pokyny pro práci a zároveň zde byla vytvořena fóra, do nichž studenti vkládali řešení dílčích úkolů. V článku uvádíme zjištění z evaluace takové výuky, a to z pohledu učitele i studentů. Limitem takové výuky byla kvalita internetového připojení jednotlivých studentů a omezená možnost vzájemné komunikace mezi studenty.

Klíčová slova

Informatika. Linux. Problémová výuka. Distanční výuka. Synchronní výuka. LMS Moodle. Jitsi Meet.

1 ÚVOD

Distanční formy výuky jsou u studentů kombinovaného studia vysokých škol dlouhodobě poměrně běžnou záležitostí [1]. V případě výuky informatických a ICT předmětů se však obvykle jedná o asynchronní výuku, při níž je každý student časově nezávislý na ostatních studentech [2, 3]. Pokud potřebuje pomoc učitele, kontaktuje jej skrze vhodné prostředky asynchronní komunikace (např. email, diskuzní fórum [4]) nebo se dostaví na osobní konzultaci. Kromě toho jsou součástí kombinované formy studia také prezenční setkání [1], během nichž je možné nejen konzultovat vyvstálé problémy, ale také přímo trénovat řešení vybraných úkolů. Tehdy výuka probíhá podobně jako u studentů prezenční formy studia.

Přechod vysokých škol na distanční výuku v důsledku pandemie COVID-19 se však od výše uvedené kombinované formy studia do značné míry liší. Především není možné dopředu naplánovat prezenční setkání, na nichž by bylo možné trénovat řešení vybraných příkladů. Tuto situaci lze řešit prostřednictvím synchronní výuky, při níž dochází k online setkávání účastníků výuky v reálném čase pomocí videokonferencí či chatů [4]. Díky tomu lze předcházet frustraci studentů, neboť studenti mohou klást dotazy, které jsou okamžitě zodpovídaný [4].

Vzhledem k výše uvedeným skutečnostem jsme se rozhodli během pandemie COVID-19 naši distanční výuku některých informatických předmětů nerealizovat jako asynchronní, ale jako synchronní. Současně jsme požadovali, aby stejně jako při prezenční výuce byla zachována aktivita studentů i okamžitá zpětná vazba učitele. Cílem tohoto článku je na příkladu problémově orientované výuky práce v operačním systému Linux a jeho správy popsat návrh a realizaci takové synchronní distanční výuky a představit zjištění z její evaluace z pohledu učitele i studentů.

Výuka popisovaná v tomto článku byla realizována v rámci předmětu Správa operačních systémů, který je určen pro studenty bakalářského studijního oboru Informační technologie a e-learning. Jeho absolventi mohou působit například jako odborníci v oblasti programování edukačních aplikací, projektování a správy informačních a e-learningových systémů nebo webu. Obsahem popisované výuky je základní ovládání operačního systému Linux v textovém prostředí (orientace v souborovém systému, manipulace se soubory a jejich vyhledávání, hromadné zpracování textových dat, správa uživatelů a správa procesů), tvorba skriptů v jazyce Bash, instalace a konfigurace služeb (webového, databázového, souborového serveru). Pro práci, při níž nebyla potřeba práva správce (roota), byl využit fakultní linuxový server.

Během činnosti, při níž tato práva byla potřeba (instalace a konfigurace služeb), studenti pracovali ve virtualizovaném operačním systému. Ten spouštěli ve virtuálním počítači prostřednictvím aplikace VmWare Workstation Player z pevného disku svých počítačů.

2 ONLINE NÁSTROJE UMOŽŇUJÍCÍ SYNCHRONNÍ VÝUKU

Základním předpokladem synchronní distanční výuky je volba vhodného nástroje, který umožňuje komunikaci studentů a učitele v reálném čase. Ačkoliv by bylo možno využít klasický textový chat, nedomníváme se, že je tato varianta příliš vhodná. Při jejím použití u studentů dochází oproti videokonferencím k nižšímu pocitu zapojení se do procesu vzdělávání [5] a k pomalejší interakci mezi účastníky výuky. Nástrojů určených pro videokonference, které lze využít při výuce, existuje celá řada. Mezi ně patří například Microsoft Teams, Zoom, Google Meet, Skype Meet Now, Cisco Webex nebo Jitsi Meet. Mnohé z těchto nástrojů však neumožňují současně sdílení obrazovek několika účastníky videokonference [6, 7, 8], a proto se příliš nehodí k výuce informatiky a ICT. Prostřednictvím těchto nástrojů by jistě bylo možné realizovat výuku orientovanou na učitele, kdy učitel prezentuje výukový obsah a případně formou videotutoriálu ukazuje studentům, jak ovládat určitý software. Tyto nástroje jsou však obtížně použitelné v situacích, kdy je výuka orientována na aktivitu studentů, tedy studenti pracují samostatně a učitel skrze nasdílené obrazovky dozoruje jejich práci a případně jim pomáhá v situacích, kdy si sami neví rady.

Z nástrojů, které umožňují současně sdílení obrazovek několika účastníky videokonference, jsme zvolili Jitsi Meet [9]. Tento nástroj je dostupný pod licencí Apache License 2.0, díky čemuž jej lze nejen zdarma používat, ale také nainstalovat na vlastní linuxový server [10], což je využitelné i ve školách [11]. Účastníci videokonference se připojují přes webový prohlížeč bez nutnosti se do služby přihlašovat.

3 NÁVRH DISTANČNÍ VÝUKY

Při návrhu synchronní distanční výuky jsme stejně jako při prezenční výuce vycházeli z předpokladu, že studenti budou pracovat individuálně, každý svým vlastním tempem. Naše výuka je založena na principech problémového vyučování, které podle Maňáka [12] zahrnuje soubor takových činností, jako je organizování problémových situací, formování problémů, poskytování nezbytné pomoci žákům při řešení problémů a při ověřování těchto řešení a řízení procesu systematizace a upevňování takto získaných poznatků.

Protože je cílem představované výuky být schopen pracovat s operačním systémem Linux jako jeho pokročilý uživatel a správce a „umět si poradit“ v různých situacích, není účelem osvojit si určitý předem stanovený výčet příkazů či postupů. Důraz je naopak kladen na schopnost analyzovat zadaný úkol, za použití Internetu vyhledat nejvhodnější postup vedoucí k jeho řešení, ten použít a ověřit jeho správnost. Z hlediska revidované Bloomovy taxonomie kognitivních cílů [13] naše výuka nezahrnuje pouhou schopnost aplikovat (tj. aplikovat určitý nalezený postup), ale také analyzovat (např. analyzovat předložený úkol), hodnotit (tj. posoudit relevantnost nalezených postupů a na základě toho zvolit z nich ten nejvhodnější; otestovat správnost svého řešení) a částečně též tvořit (např. formulovat správný dotaz pro vyhledávání).

Uvedený přístup vychází z našeho dřívějšího zjištění, že studenti daného studijního oboru nejsou zvyklí řešit odborné informatické problémy za použití Internetu. Zejména v úvodu semestru jsou dotazy pro vyhledávání mnohdy nepřiliš konkrétní, preferovaným jazykem je čeština (ačkoliv studenti během studia absolvovali dva semestry anglického jazyka) a z nabídnutých výsledků nejsou studenti často schopni vybrat ten nejvíce relevantní. Uvedené zkušenosti odpovídají názoru Morrisona a Bartona [14], podle nichž je potřeba rozvíjet strategie vyhledávání na internetu.

Výjimkou z uvedeného způsobu práce je tvorba skriptů, kdy jsou studentům při přednáškách na ukázkových příkladech představeny řídicí struktury jazyka Bash a jeho syntaxe. Studenti mají při tvorbě vlastních skriptů podklady z těchto přednášek, a tudíž nemusí způsoby zápisu řídicích struktur hledat na Internetu. Pokud však z dřívějších cvičení zapomenou příkazy používané v rámci skriptů, musí si je umět vyhledat.

K případné spolupráci mezi studenty jsme zaujali neutrální stanovisko, tedy jsme ji studentům nezakázali, ale ani nedoporučili. Ačkoliv si uvědomuje přínos skupinového učení pro rozvoj odborných i sociálních dovedností studentů [15], nepovažujeme jej v tomto případě za příliš reálné. Textový chat je dle našeho názoru pro daný účel poněkud neefektivní a při realizaci soukromého hlasového chatu souběžně s hlavní videokonferencí by bylo pro studenty obtížné vnímat současně řeč spolužáků v hlasovém chatu i případný komentář učitele ve společné videokonferenci. Snaha o párovou či skupinovou diskuzi nad způsoby řešení vybraných úkolů (např. jak úkol vyřešit, jaké z alternativních řešení je výhodnější) v tomto předmětu během dřívější prezenční výuky nebyla příliš úspěšná. Studenti se do diskuze zapojovali jen neradi, řešením představovaným ostatními studenty při skupinové diskuzi nevěnovali mnoho pozornosti (tedy je vyposlechli, ale nereagovali na ně) a bylo zřejmě očekávání konce diskuze a zadání dalších úkolů. Z uvedených důvodů jsme během návrhu distanční výuky diskuzi studentů upozadili.

Pro podporu naší výuky jsme využili LMS Moodle, v němž byl již dříve vytvořen kurz reprezentující tento předmět. Tento kurz byl v předchozích letech pro účely klasické prezenční výuky rozdělen do témat a v každém tématu byla vytvořena speciální stránka s pokyny pro práci studentů. Základem těchto pokynů byla zadání problémově orientovaných úloh, které měly postupně se zvyšující náročností. Pro účely distanční výuky byly tyto pokyny zpřesněny, avšak stále se jedná o pokyny, co mají studenti dělat, a nikoliv tutoriál, jak mají postupovat. Mezi takové pokyny patří například: „*Nalezňte všechny soubory, které se nachází v adresáři vecko, mají příponu jpg nebo JPG a jejichž velikost je větší než 1MB*“ nebo „*Změňte port, na kterém naslouchá OpenSSH server, na 66*“.

Využití nástrojů synchronní výuky naplňuje požadavek, aby učitel mohl průběžně sledovat práci studentů. Oproti prezenční výuce však při této výuce studenti obvykle nejsou tolik komunikativní, zejména je znatelná neochota studentů informovat učitele o splnění dílčích úkolů. Proto je pro každé téma v LMS Moodle vytvořeno jedno fórum, do nějž studenti vkládají svá řešení dílčích úkolů. V každém fóru je vytvořeno několik vláken (obvykle 6 až 12) obsahující otázky, které slouží pro ověření správnosti řešení zadaného úkolu. Tato fóra jsou nastavena tak, aby se studentům zobrazily odpovědi ostatních studentů až 30 minut poté, co sami na danou otázku odpoví. Ve stránce s pokyny se na vhodných místech nachází informace, že je potřeba do fóra vložit řešení daného úkolu. Například k úkolu „*Nalezňte všechny soubory, které se nachází v adresáři vecko, mají příponu jpg nebo JPG a jejichž velikost je větší než 1MB*“ se váže pokyn „*Pomocí vhodného příkazu*“

spočítejte, kolik takových souborů je. Svě zjištění spolu s použitým příkazem zapíše do fóra Poznanky.*

Tento postup byl zvolen na základě zkušeností z prezenční výuky předmětu v dřívějších letech. Tehdy byla výuka více synchronizovaná v tom ohledu, že studenti měli pro splnění několika dílčích úkolů vyhrazený určitý čas (ten nebyl pevně stanoven, ale odvíjel od podílu studentů, kteří již měli hotovo). Po jeho uplynutí učitel požádal vybrané studenty, aby představili svá řešení. Někteří studenti však místo intenzivní snahy o samostatné vyřešení úkolů čekali, až budou možná řešení prozrazena, a pak je jen zapsali a vyzkoušeli.

Využití fóra pro odevzdávání řešení úkolů je výhodné v případě, že řešením je určitý příkaz, krátký textový výsledek nebo vysvětlení určitého problému a studenty toto vkládání odpovědí příliš časově nezatěžuje. V případě tvorby skriptů, instalace a konfigurace služeb je využito jiného přístupu. Tehdy mají studenti učitele informovat o splnění úkolů a učitel pak pohledem na jimi sdílenou obrazovku ověří, zda je úkol vyřešen správně. Pokud vzhledem ke složitosti problému není správnost řešení okamžitě zcela zřejmá nebo je-li učitel na pochybách, nechá studenta předvést funkčnost daného řešení. Příkladem budiž výše uvedená změna portu, na němž naslouchá OpenSSH server. Tehdy učitel studenta požádá, aby se pomocí programu Putty ke svému serveru ještě jednou připojil za použití portu 66, a je-li toto připojení úspěšné, lze dané řešení považovat za správné. Pokud učitel zaznamená ve studentově řešení chybu, může jej vybědnout k provedení určitého úkonu (např. zadání cesty k neexistující složce), aby se chyba projevila. Na základě toho si student sám uvědomí, že jeho řešení je nesprávné, v optimálním případě chybu nalezne a informuje učitele, že aktualizované řešení již funguje správně.

4 REALIZACE DISTANČNÍ VÝUKY

Výše vytvořený model výuky byl pilotně ověřen v zimním semestru 2020/2021 na skupině 15 studentů. Tito studenti byli rozděleni do dvou dílčích podskupin. Protože byla výuce vyhrazena časová dotace 90 minut týdně, bylo dle časových možností studentů a učitele rozhodnuto o realizaci výuky v délce cca 135 minut jednou za dva týdny (obě podskupiny se ve výuce po týdnu střídaly).

4.1 Technické hledisko realizované výuky

Pro realizaci distanční výuky byla zvolena platforma Jitsi Meet. Vzhledem k výpadkům spojení při testování komunikace bylo přikročeno k instalaci této platformy na fakultní server. Pro tento účel byl na serveru zřízen virtuální počítač s operačním systémem Ubuntu, kterému bylo dáno k dispozici 12 sdílených jader procesoru, vyhrazeno 8 GB operační paměti a 20 GB prostoru na pevném disku. Následně byla v rámci této platformy vytvořena místnost, v níž probíhaly samotné videokonference.

Studenti byli vždy na začátku výuku instruováni, aby aktivovali sdílení svých obrazovek, vypnuli webkamery a ztlumili mikrofony, aby nedocházelo k ozvěnám a nebyl šířen ruch z místa, kde se studenti nacházeli. Pokud chtěl student hovořit (požádat o pomoc s řešením úkolu či nějakého problému technického rázu, informovat učitele o splnění dílčího úkolu apod.), nemusel se hlásit, ale mohl rovnou zapnout mikrofon a začít mluvit. Sám učitel nesdílel obrazovku ani obraz z webkamery, ale měl po většinu času zapnutý mikrofon, aby mohl průběžně komentovat práci studentů.

Během výuky docházelo k občasným výpadkům spojení některých studentů. Na začátku distanční výuky měla s těmito výpadky problémy až čtvrtina účastníků (včetně učitele připojeného ze své kanceláře na fakultě). Bylo zjištěno, že problém je v řadě případů způsoben nevhodným webovým prohlížečem, který daný účastník pro videokonferenci využívá. Tento problém se týkal především prohlížeče Mozilla Firefox, naopak jako nevhodnější se ukázal prohlížeč Google Chrome.

U několika studentů používání vhodného webového prohlížeče k eliminaci výpadků nestačilo a na vině bylo patrně méně kvalitní připojení daných studentů k internetu. Ačkoliv tito studenti neměli zapnuté webkamery, jako problematické se ukázalo sdílení obrazovky. Dokud daný student nesdílel svou obrazovku (a neměl zapnutou webkameru), bylo možné s ním hlasově komunikovat. Po zapnutí sdílení obrazovky však bylo spojení s ním přerušeno. Během semestru jsme ověřili, že tyto problémy mají stále ti sami studenti a že oni sami hodnotí své připojení k internetu jako problematické.

Vzhledem k vysoké míře individualizace výuky mohli tito studenti pracovat zcela samostatně bez dohledu učitele a v případě potřeby s ním mohli komunikovat hlasově nebo pomocí textového chatu.

Poměrně nečekaným problémem se stala průběžná kontrola činnosti prostřednictvím sdílených obrazovek u těch studentů, kteří používají obrazovky s vysokým rozlišením (např. s rozlišením Ultra HD). Protože učitel využíval 19" monitor s rozlišením 1280 x 1024 pixelů, bylo pro něj písmo z nasdíleného Ultra HD monitoru příliš malé, nečitelné. Jako řešení se ukázalo v takových případech nesdílet celou obrazovku počítače, ale pouze okno dané aplikace (např. Putty).

4.2 Pedagogické hledisko realizované výuky

4.2.1 Synchronní část výuky

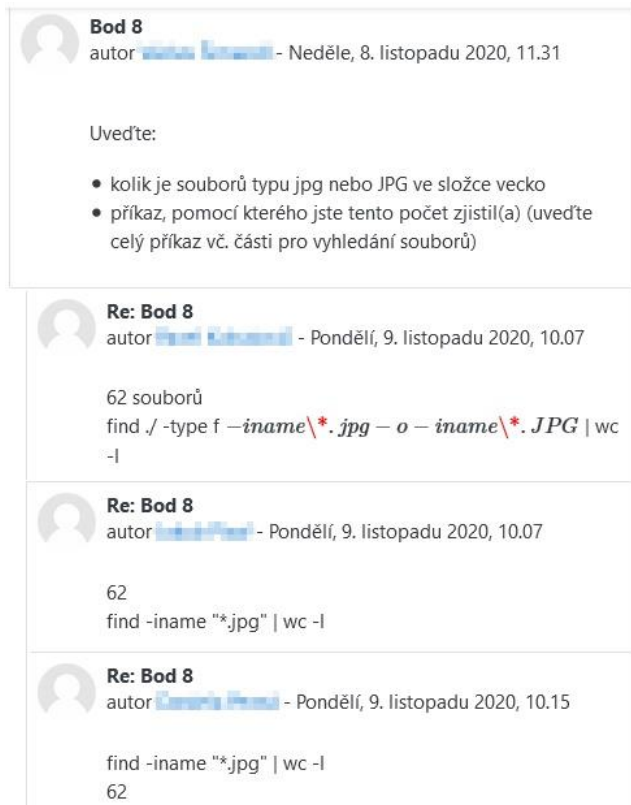
Z pedagogického hlediska výuka probíhala dle připraveného návrhu. Studenti pracovali samostatně, případně v kooperaci se spolužáky. Ačkoliv učitel tuto možnost explicitně nedoporučil ani nezakázal, pokud některý student takovou spolupráci zmínil, učitel ji hodnotil pozitivně. Hlavní činností učitele bylo průběžně prohlížet nasdílené obrazovky studentů. Pokud si všiml, že daný student delší dobu řešil určitý dílčí problém, potýkal se s neočekávanou technickou komplikací nebo když o to student sám aktivně požádal, učitel mu v řešení poradil stejně jako při běžné prezenční výuce.

Ochota požádat učitele o pomoc se do značné míry mezi studenty lišila. Zatímco někteří studenti po určité době řešení problému učitele o pomoc požádali, jiní se snažili problém řešit sami nebo v kooperaci se spolužáky a v některých případech svou snahu vzdali a úkol ponechali nevyřešený. Přitom se zdá, že o pomoc v případě nesnázi dříve žádali rychlejší studenti, zatímco ti pomalejší o ni žádali později, případně o ni nepožádali a snahu o vyřešení vzdali¹. K tomuto jevu docházelo, přestože učitel studenty opakovaně nabádal, že jej mohou v případě potřeby požádat o pomoc. Tento rozpor lze chápat tak, že někteří studenti nechtějí učitele svou žádostí „obtěžovat“ nebo nechtějí před ostatními studenty či před učitelem přiznat, že si s úkolem neví rady. Vzhledem k tomu se ukazuje jako přínosné průběžné

¹ Toto tvrzení vychází ze subjektivního náhledu učitele na průběh výuky a není podloženo analýzou záznamu výuky.

prohlížení nasdílených obrazovek učitelem, neboť učitel může studentovi pomoci bez jeho předchozí aktivní žádosti².

Tam, kde to bylo v pokynech určeno, studenti odpovídali prostřednictvím fóra v LMS Moodle, jak je zobrazeno na Obrázku 1. Učitel odpovědi studentů průběžně kontroloval, a pokud shledal některou odpověď nesprávnou, daného studenta na problém ústně upozornil. Student se obvykle k tomuto úkolu vrátil a svou odpověď upravil, popř. ji vložil znovu. Když téměř všichni (v ideálním případě všichni) studenti odpověděli na danou otázku, učitel ještě jednou zkontroloval jejich odpovědi, ústně zhodnotil jejich správnost a případně vysvětlil, proč je konkrétní odpověď nesprávná.



Obrázek 1: Odpovědi studentů s řešením dílčího úkolu ve fóru LMS Moodle

Při kontrole odpovědi studentů vložených do fóra se ukázalo být náročné věnovat se odpovědím vloženým do fóra a průběžně sledovat obrazovky studentů. Situace byla komplikovanější také tím, že studenti pracovali rozdílnou rychlostí, a tak nové odpovědi přibývaly současně na několik různých otázek v různých vláknech fóra. Tato situace nastávala i přes to, že byly vytvořeny a použity nepovinné složitější úkoly pro rychlejší studenty.

Výhodou tohoto přístupu bylo aktivní zapojení všech studentů do výuky a značný přehled učitele o tom, kdo jaký úkol splnil. Přínosem průběžného kontrolování odeslaných odpovědí byla

relativně rychlá zpětná vazba, díky níž se studenti dozvěděli správnost svého řešení během několika minut a nikoliv až po odeslání odpovědi ostatními studenty popř. na konci výuky.

4.2.2 *Asynchronní část výuky*

Protože časová dotace 135 minut jednou za dva týdny nenahrazovala dostatečně rozvrhovanou časovou dotaci 90 minut týdně, měli studenti úkoly, které nestihli vyřešit při synchronní výuce, dopracovat individuálně. V případě úkolů, jejichž řešení odevzdávali prostřednictvím fóra v LMS Moodle, učitel před začátkem další synchronní výuky odevzdaná řešení zkontroloval a v rámci tohoto fóra okomentoval jejich správnost.

Úkoly na tvorbu skriptů studenti odevzdávali do speciální složky v rámci fakultního serveru, na němž pracovali. Tato složka byla nastavena tak, že studenti do ní mohli sice soubory vkládat, ale nemohli si zobrazit soubory vložené jinými studenty. Učitel následně skripty zkontroloval a asynchronním způsobem (zejména pomocí zpráv v Microsoft Teams³) studentům poskytl zpětnou vazbu. Před začátkem další synchronní výuky učitel změnil nastavení uvedené složky tak, aby si studenti mohli prohlédnout (a případně spustit) skripty vytvořené ostatními studenty. Výhoda tohoto přístupu spočívala v poměrně rychlém odevzdání úkolu ke kontrole (studenti pouze zkopírovali soubor do správné složky) a snadné kontrole ze strany učitele (učitel mohl nejen pročíst zdrojový kód skriptu, ale také jej okamžitě spustit).

Úkoly týkající se instalace a konfigurace služeb byly učitelem kontrolovány na začátku další synchronní distanční výuky.

Mimo synchronní výuku mohli studenti řešení úkolů konzultovat s učitelem během synchronní výuky druhé skupiny, během jeho online konzultačních hodin nebo jej kontaktovat asynchronním způsobem (pomocí emailu, zpráv v prostředí Microsoft Teams či LMS Moodle). Studenti z těchto možností využili zprávy v Microsoft Teams a někteří požádali o konzultaci během synchronní výuky druhé skupiny. Naopak konzultaci během online konzultačních hodin učitele nevyužil nikdo.

5 EVALUACE REALIZOVANÉ VÝUKY

5.1 *Evaluace výuky z pohledu učitele*

Dle názoru učitele, který je současně autorem tohoto článku, byla výuka poměrně úspěšná, neboť splnila vytčené cíle. I přes komplikace spojené s uzavřením vysokých škol a přechodu na distanční výuku se studenti během semestru naučili pracovat s operačním systémem Linux jako jeho pokročilí uživatelé či správci. Při závěrečné zkoušce byla naprostá většina studentů schopna plnit zadané úkoly a prokázala tak dostatečné kompetence v problematice⁴. Jelikož byly při ústní části zkoušky použity též problémově orientované úkoly, byla prověřena schopnost studentů analyzovat tyto úkoly, za použití Internetu vyhledat vhodné postupy vedoucí k jejich řešení, ty použít a ověřit jejich správnost.

Základní výhodou synchronní distanční výuky v prostředí Jitsi Meet byla možnost průběžně prohlížet nasdílené obrazovky studentů a v tomto ohledu tak pracovat podobně jako při prezenční výuce. V určitém ohledu toto řešení prezenční výuku překonalo, neboť učitel měl možnost práci studentů pozorovat, aniž by byl

² Pokud učitel nabídl studentovi pomoc a on ji odmítl, učitel samozřejmě toto rozhodnutí respektoval.

³ Zasilání zpráv prostřednictvím Microsoft Teams bylo iniciováno samotnými studenty. V případě požadavku používat pouze

volně dostupné nástroje by bylo možné využít například zprávy v prostředí LMS Moodle.

⁴ Z 15 studentů zkoušku nezvládl jediný student; dva studenti splnili písemnou část zkoušky, ale k ústní zkoušce se prozatím nepřihlásili.

student stresován tím, že učitel je v jeho blízkosti a sleduje každý jeho úkon. Zároveň nenastával ani problém s narušení osobní zóny studenta ve chvíli, kdy na jeho obrazovce učitel delší dobu pročítal zapsané texty a například mu pomáhal nalézt chybu. Protože studenti mohli sdílet pouze okno používané aplikace (např. Putty), bylo marginalizováno narušování soukromí studenta učitelem.

Odevzdávání řešení dílčích úkolů do fóra v LMS Moodle se ukázalo být také jako výhodné, neboť všichni studenti byli nuceni být aktivní. To je v kontrastu s dřívější prezenční výukou, při níž bylo po určité době ukázáno možné řešení daného úkolu a studenti, kteří úkol nevyřešili, jej pouze opsali a vyzkoušeli. Při využití fór se proto nestávalo, že by pomalejší studenti čekali, až jim bude řešení prozrazeno.

Za zásadní nevýhodu synchronní distanční výuky oproti prezenční výuce považujeme vzniklé technické problémy, především problémy s kvalitou připojení k Internetu některých studentů. Tito studenti nemohli sdílet své obrazovky, a proto byli odkázáni pouze na hlasovou nebo textovou komunikaci s učitelem. Učitel proto neměl přehled o jejich práci a nemohl jim ani efektivně poradit s dílčím řešením úkolu. Tato situace byla pro učitele poměrně nekomfortní. Domníváme se proto, že využití videokonferenčních nástrojů, které umožňují v určitý okamžik sdílet obrazovku jen jednoho účastníka, by značně limitovalo kvalitu takové výuky.

Za další nevýhodu realizované výuky oproti prezenční výuce považujeme absenci sociálního kontaktu a neformální komunikace mezi studenty. Protože studenti nemohli v rámci místnosti, v níž probíhala synchronní výuka, mezi sebou ústně komunikovat, byli odkázáni buď na využití integrovaného soukromého chatu, nebo jiných komunikačních kanálů. Z tohoto důvodu si vzájemně nemohli pomáhat tak efektivně jako při prezenční výuce a byli více odkázáni na případnou pomoc učitele.

5.2 Evaluace výuky z pohledu studentů

Abychom zjistili názor studentů na tuto výuku, realizovali jsme mezi studenty dotazníkové šetření. Prostřednictvím Formulářů Google jsme vytvořili krátký dotazník obsahující tři otevřené otázky:

- Stručně zhodnoťte svůj celkový dojem z distanční výuky předmětu.
- Napište, co se Vám na formě výuky **libilo**, co by bylo vhodné při distanční nebo i prezenční formě výuky využít i v jiných předmětech.
- Napište, co se Vám na formě výuky **nelíbilo**, co by bylo vhodné příště udělat jinak.

Žádost o vyplnění dotazníku byla po skončení semestru odeslána emailem všem 15 studentům předmětu s ujištěním, že dotazník je zcela anonymní. Dotazník vyplnilo 7 studentů, návratnost byla tedy přibližně 46 %.

Z odpovědí studentů vyplývá, že realizovaná výuka byla dostatečnou náhradou za prezenční výuku, avšak prezenční výuka by byla vhodnější: „*Vše probíhalo v pohodě a způsob, jakým byla výuka vedena, byl vhodný. Samozřejmě nepopírám, že by to prezenčně mohlo být lepší a zároveň i pro všechny snadnější*“ nebo „*Převedení předmětu do distanční výuky bylo provedeno na výbornou*“. Jednomu studentovi dokonce výuka vyhovovala více než klasická prezenční výuka: „*Tento styl výuky mi vyhovoval více než prezenční, v případě rychlejšího splnění bylo možné se věnovat domácí aktivitě. Navíc jsme ušetřili čas, který bychom věnovali cestě do budovy školy*“.

Dílčí způsoby realizace výuky byly studenty hodnoceny taktéž vesměs kladně: „*Libil se mi styl pojetí výuky - samostatné úkoly, vlastní tempo práce, pomoc v případech potřeby*“. Ačkoliv je samostatné nalézání řešení problému s pomocí Internetu pro studenty poměrně náročné, hodnotí je jako přínosné: „*Vyhledávání informací na internetu při cvičeních bylo velmi naučné, lepší než plnit úkoly ve třídě a informace dostávat od spolužáků (lépe si jednotlivé příkazy zapamatují)*“.

Studenti oceňovali sdílení obrazovek a související možnost pomoci učitele: „*Určitě je super okamžitá reakce díky sdílení obrazovek*“. Jako úspěšné lze hodnotit i využití fór v LMS Moodle pro odevzdávání řešení dílčích úkolů: „*(Libilo se mi) následně odesílání poznatku do fóra - bylo hned jasné zda je to dobře či špatně. Nyní při učení na zkoušku oceňuji i to, že po žácích chtěl lektor zapisování jednotlivých příkazů do fóra (nemusím je znovu složitě hledat na internetu)*“. Jeden ze studentů zmínil jako pozitivum i možnost online konzultací mimo dobu synchronní výuky: „*Libil se mi Váš přístup, nebyl problém přijít na hodinu druhé skupiny s dotazem*“.

Za negativum distanční výuky studenti označovali především nedostatek sociálního kontaktu s ostatními studenty: „*(...) bohužel ztrácíme fyzický kontakt, ale ten věřím, že lze dohnat mimo školu*“.

Jako problematické označili někteří studenti poměrně velké množství úkolů, které měli vyřešit: „*Každopádně celkový čas je příliš krátký. Distanční výuka a celkově samostatná práce by chtěla více času během cvičení*“. Tento názor kontrastuje s názorem studenta, který se zmiňoval o možnosti opustit distanční setkání po splnění všech úkolů ještě před koncem výuky (viz citace výše). Dělení studentů na dvě skupiny a realizace synchronní distanční výuky v délce tří navazujících vyučovacích hodin bylo podrobeno kritice i dalšího ze studentů, podle jehož názoru byla taková výuka příliš dlouhá: „*Jediné co bych vytkl je délka cvičení. 150 minut je opravdu velmi dlouhá doba na to se soustředit na jednu věc*“.

Někteří studenti uvedli taktéž nápady, jak příště výuku zlepšit. Jeden student doporučil mimo hlavní místnost synchronní výuky vytvořit dílčí pracovní skupiny, v jejichž rámci by si studenti mohli bez přítomnosti učitele vzájemně radit a spolupracovat: „*Možná zkusit využít i na cvičení teams v tom hledisku, že by se tam daly vytvořit týmy, kde by mohli rychlejší žáci pomoci pomalejším v případě potřeby a snížilo by se tím vytížení učitele. Navíc by si tím rychlejší studenti mohli vyzkoušet pozici toho, kdo vysvětluje a učí*“. Další student zmínil možnost využít během semestru průběžně cvičné písemné testy: „*Myslím, že průběžné, byť nehodnocené, testy by hodně ovlivnily výsledky hlavního testu*“.

6 ZÁVĚR

Navrhli a realizovali jsme distanční výuku informatiky, která byla koncipována jako synchronní (vyjma dopracování některých úkolů asynchronním způsobem). Jejím základem se stala problémově orientovaná výuka a videokonference pomocí platformy Jitsi Meet, při nichž studenti sdíleli obrazovky svých počítačů. Díky tomu mohli studenti pracovat individuálně a současně jim byl k dispozici učitel, se kterým mohli efektivně konzultovat svá řešení a který mohl kontrolovat správnost řešení některých úkolů pohledem na jejich obrazovky. Do výuky byly zařazeny prvky asynchronní komunikace – odevzdávání dílčích řešení vybraných úkolů prostřednictvím fór v LMS Moodle. Na základě toho mohl učitel průběžně kontrolovat vytvořená řešení a student nemusel čekat, až toto řešení učitel zkontroluje na jeho obrazovce.

Limitem námi představené výuky je kvalita internetového připojení jednotlivých studentů a také počet studentů účastnících se této

výuky. Protože mají studenti při této výuce omezenou možnost vzájemně komunikovat (radit si, diskutovat o možných řešeních), zvyšuje se oproti prezenční výuce čas, který by měl učitel pro zachování určité kvality výuky věnovat jednomu studentovi. Řešením by mohlo být vytvoření řady dílčích diskuzních online místností, v nichž by se mohly dobrovolně, souběžně se synchronní výukou, scházet dvojice či trojice studentů. Ačkoliv by tito studenti byli současně přítomni i v hlavní online místnosti synchronní výuky, zde by mohli v případě potřeby diskutovat o řešení zadaných úkolů podobně jako při prezenční výuce. Role učitele by zůstala velmi podobná jako při námi představené výuce, avšak lze předpokládat, že by vzhledem k umožnění vzájemné spolupráce studentů kleslo množství času, které je potřeba věnovat jednomu studentovi. Díky této relativně drobné změně výukové strategie by bylo možné realizovat distanční výuku s větší skupinou studentů při zachování obdobné kvality výuky.

BIBLIOGRAFICKÉ ODKAZY

- [1] POLÁKOVÁ, T. Kombinované studium: Vzdělání při práci i velká sebekázeň. In *VysokeSkoly.cz* [online]. 2019 [cit. 2021-01-17]. <https://www.vysokeskoly.cz/clanek/kombinovane-studium-vzdelani-pri-praci-i-velka-sebekazen>
- [2] JOHNSON, G. M. Synchronous and Asynchronous Text-Based CMC in Educational Contexts: A Review of Recent Research. In *TechTrends*. 2006, vol. 50, no. 46, p. 46–53. DOI=<https://doi.org/10.1007/s11528-006-0046-9>
- [3] ROMISZOWSKI, A., MASON, R. Computer-mediated communication. In *Handbook of Research for Educational Communications and Technology*. D. Jonassen (ed.). New York : Macmillan, 2001, p. 397–431. Dostupné též na http://www.communicationcache.com/uploads/1/0/8/8/10887248/computer-mediated_communication_-_chapter_15.pdf
- [4] HRASTINSKI, S. Asynchronous and synchronous e-learning. In *Educause quarterly*. 2008, vol. 31, no. 4, p. 51–55. Dostupné též na: <https://er.educause.edu/-/media/files/article-downloads/eqm0848.pdf>
- [5] CLARK, C., STRUDLER, N., GROVE, K. Comparing asynchronous and synchronous video vs. text based discussions in an online teacher education course. In *Online Learning*. 2015, vol. 19, no. 3, p. 48-69. DOI=<https://doi.org/10.24059/olj.v19i3.668>
- [6] Sharing multiple screens. In *Microsoft Teams UserVoice* [online]. 2020 [cit. 2021-01-17]. <https://microsoftteams.uservoice.com/forums/555103-public/suggestions/40195825-sharing-multiple-screens>
- [7] How to share two screens by hangout?. In *Google Meet Community* [online]. 2020 [cit. 2021-01-17]. <https://support.google.com/meet/thread/52752403>
- [8] Can multiple participants share their screen during a Webex meeting?. In *Cisco Community* [online]. 2020 [cit. 2021-01-17]. <https://community.cisco.com/t5/adopting-webex/can-multiple-participants-share-their-screen-during-a-webex-m-p/4096149>
- [9] Jitsi Meet. <https://meet.jit.si/>
- [10] Self-Hosting Guide. In *Jitsi Meet Handbook* [online]. 2021 [cit. 2021-01-17]. <https://jitsi.github.io/handbook/docs/devops-guide/devops-guide-start>
- [11] DVOŘÁK, D. Jitsi meet: zkušenosti s vlastní videokonferenční platformou. In *Root.cz* [online]. 2020 [cit. 2021-01-17]. <https://www.root.cz/clanky/jitsi-meet-zkusenosti-s-vlastni-videokonferencni-platformou/>
- [12] MAŇÁK, J. a kol. *Alternativní metody a postupy*. Brno : Masarykova univerzita v Brně, 1997. 94 s. ISBN 80-210-1549-7.
- [13] BYČKOVSKÝ, P., KOTÁSEK, J. Nová teorie klasifikování kognitivních cílů ve vzdělávání: revize Bloomovy taxonomie. In *Pedagogika*. 2004, roč. 54, č. 3, s. 227-242. ISSN 2336-2189. Dostupné též na: <https://pages.pedf.cuni.cz/pedagogika/?p=1821&lang=cs>
- [14] MORRISON, R., BARTON, G. Search engine use as a literacy in the middle years: The need for explicit instruction and active learners. In *Literacy Learning: The Middle Years*. 2018, vol. 26, no. 3, p. 37-47.
- [15] PRŮCHA, J., WALTEROVÁ, E., MAREŠ, J. *Pedagogický slovník*. Praha : Portál, 2009. 402 s. ISBN 978-80-7367-647-6.

Modelové školní vzdělávací programy pro novou informatiku v Česku

Model School Educational Programmes of the New Informatics in Czechia

Jiří Vaniček

Jihočeská univerzita v Českých
Budějovicích, Pedagogická fakulta
Jeronýmova 10
37115 České Budějovice
Česká republika
vanicek@pf.jcu.cz

Jan Berki

Technická univerzita v Liberci, Fakulta
přírodovědně-humanitní a pedagogická
Univerzitní náměstí 1410/1
46117 Liberec
Česká republika
jan.berki@tul.cz

ABSTRACT

Model school educational programmes of the new informatics were prepared and published in the framework of the project PRIM – “Support for development of computational thinking”. They are based on the revision of the Frame educational programme for basic education. These model programmes build on a set of textbooks for informatics at basic schools, which have been created as a support for new educational content. The article describes reasons, methodology, approach and individual variants of these model school programmes. Authors explain also purpose of the use of these variants and acquaint with the their structure. Model school educational programmes could help schools that do not orient themselves in the reform and need to put the new educational area into practice.

Keywords

Informatics. School educational programme. Support. Computational thinking. Educational content. New content. Basic School.

ABSTRAKT

V rámci projektu PRIM – „Podpora rozvíjení informatického myšlení“ byly připraveny a publikovány modelové školní vzdělávací programy pro novou informatiku. Vycházejí z revize rámcového vzdělávacího programu pro základní vzdělávání. Navazují na sadu učebnic informatiky pro ZŠ, které byly vytvořeny v rámci stejného projektu jako podpora pro nový informatický obsah. Článek popisuje důvody pro vznik modelových programů a metodu práce při jejich vyvíjení. Vysvětluje též účel použití jednotlivých variant a seznamuje s jejich strukturou. Modelové školní vzdělávací programy mohou posloužit školám, které se v nové reformě informatiky neorientují, se snadnějším zavedením nové vzdělávací oblasti do praxe.

Klíčová slova

Informatika. Školní vzdělávací program. Podpora. Informatické myšlení. Vzdělávací obsah. Nový obsah. Základní škola.

1 DŮVODY VYTVOŘENÍ MODELOVÝCH ŠVP

Právě probíhající reforma výuky informatiky v České republice klade na české školy výrazné nároky. Požadavky stávajícího dobíhajícího Rámcového vzdělávacího programu pro základní vzdělávání (dále jen RVP) [1] ve vzdělávací oblasti *informační a komunikační technologie* (dále jen ICT) jsou založeny na

uživatelském přístupu k technologiím a rozvíjení digitální gramotnosti uvnitř jednoho izolovaného, navíc nedostatečně časově dotovaného předmětu. Inovovaný RVP [2] rozlišuje základy informatiky od uživatelského přístupu, jehož cílem je rozvíjet digitální gramotnost. Cílem nové informatiky je především rozvíjet informatického myšlení žáků [3] a tomu je uzpůsobena náplň nové vzdělávací oblasti *informatika*. K rozvoji digitální gramotnosti by měly přispívat všechny vzdělávací oblasti, resp. obory.

Tato reforma staví školy, které dosud ve svém kurikulu neměly žádnou informatickou pasáž, do složité situace, kde vzít vzdělávací materiály, učitele, kteří budou učit informatický obsah, a finance na nákup odpovídajících učebních pomůcek. Velkým potenciálním problémem je také sestavení informatické pasáže školního vzdělávacího programu (dále jen ŠVP).

ŠVP v České republice stanoví zejména konkrétní cíle vzdělávání, délku, formy, obsah a časový plán vzdělávání [4]. ŠVP musí být v souladu se státním RVP a učitel se při své výuce musí řídit ŠVP své školy. Touto dvojstupňovostí kurikula je školám dána jistá autonomie a možnost profilovat se, např. využít zkušenosti pedagogického sboru. K tomuto účelu také slouží tzv. disponibilní hodiny, kterými může škola rozšířit hodinovou dotaci libovolného předmětu.

Na českých školách je málo učitelů, kteří mají odpovídající informatické vzdělání. V oblasti ICT nebyla ve školním roce 2019/2020 výuka na téměř 60 % základních škol pokryta aprobovanými učiteli, což je nejvíce po výtvarné výchově [5]. Dá se předpokládat, že ještě méně je pedagogů, kteří již mají zkušenosti s výukou informatických témat. Je tak velmi problematické po každé škole chtít, aby sestavila kvalitní ŠVP tak, aby naplnil požadavky nového RVP.

Tento stav byl důvodem pro úvahy, zda vytvořit nějaký vzorový ŠVP z informatiky, který by škola mohla převzít a usnadnil tak svým učitelům náběh reformy informatiky. Stály zde ovšem významné argumenty proti. Ty vycházely z přesvědčení, že školy mají vytvářet svůj ŠVP samy, neboť diskuse při vytváření či úpravách mezi ředitelem a učiteli navzájem je důležitá. Neméně důležitá byla negativní zkušenost z minula, kdy řada škol svůj ŠVP od někoho pouze převzala bez snahy jej sama vystavět s ohledem na svoji situaci a zaměření a vyladit podle svých možností. Na druhou stranu hrozilo riziko, že škola svůj ŠVP v této nově pojaté vzdělávací oblasti sestaví bez hlubší znalosti cílů a zkušeností z pilotního ověřování a tedy nekvalitně. V případě, že by modelové ŠVP nebyly k dispozici, mohli by se učitelé a ředitelé cítit „necháni

na pospas“. Takový pocit by mohl zvýšit jejich negativní postoj k celé reformě.

Diskuse o ŠVP probíhala jednak na úrovni vedení projektu Podpora rozvíjení inženýrského myšlení (dále jen PRIM) [6], který mohl alokovat prostředky na vývoj takovýchto kurikulárních dokumentů, jednak se zástupci Národního ústavu pro vzdělávání, resp. Národního pedagogického institutu, a Ministerstva školství, mládeže a tělovýchovy. Nakonec zvítězil návrh vytvořit sadu podpůrných, tzv. modelových ŠVP, které budou odpovídat různým situacím, ve kterých se základní školy aktuálně vyskytují. To má za cíl umožnit snadnější náběh jak školám, které aktuálně nejsou připraveny a vybaveny, tak školám, které chtějí svůj ŠVP aktivně vybudovat a mají zkušenosti.

Cílem přitom bylo maximálně využít dat a zkušeností získaných z tvorby a pilotního ověřování sady učebnic, vytvořených v rámci projektu PRIM, a synergicky je využít jako vzdělávací zdroje. Při jejich tvorbě byl kladen důraz na konstruktivistický přístup k výuce a postupné získávání zkušeností učícího se. V rámci projektu vznikly vzdělávací materiály nejen na tradiční inženýrská témata programování nebo populární práce s robotickými pomůckami, která jsou na některých školách v současnosti vyučovány, ale i ke zcela novým inženýrským tématům. Tato témata jako např. data a jejich kódování, modelování pomocí schémat nebo informační systémy [7] nejsou ještě v základním vzdělávání běžná a doposud byla přítomna až v terciálním vzdělávání s diametrálně odlišnými nároky na učícího se.

Záměr vytvořit modelová ŠVP z informatiky uspil požadavek vyjádřený MŠMT, aby pro chystané vyhlášení revize RVP z informatiky byla připravena co největší podpora. Modelové ŠVP prezentované tímto článkem jsou koncipovány pouze pro novou vzdělávací oblast Informatika.

2 METODA A HARMONOGRAM PRÁCE

Na těchto ŠVP se pracovalo ve dvou etapách. První etapa proběhla v únoru a březnu 2020. Protože během ní neměl nový RVP definitivní kontury, úzká skupina autorů ŠVP vycházela z pracovní verze RVP, kterou v tu dobu vytvářel ICT panel pro inovaci výuky ICT [8]. Tato skupina byla součástí panelu a na nových RVP se podílela. V této etapě se vycházelo z předpokládané hodinové dotace pro informatiku 3 + 4, tedy 3 hodiny na 1. stupni a 4 hodiny na 2. stupni, v každém ročníku od třetího od jedné hodiny týdně.

Jednotlivé tematické celky popsané v RVP byly konfrontovány s učebnicemi z řady Informatika PRIM z pohledu, nakolik jsou jimi pokryty. Konkretizací očekávaných výstupů RVP byly vytvořeny očekávané výstupy ŠVP tak, aby čerpaly v maximální míře z již připravených učebnic. K nim bylo sestaveno odpovídající učivo tak, aby bylo pomocí konkrétní učebnice dobře realizovatelné. Takto připravené tabulky s očekávanými výstupy a učivem pro každý tematický blok byly předloženy již zmíněnému ICT panelu, na jehož online jednáních byly obhajovány a na základě jeho připomínek upravovány.

Učivo pak bylo prolinkováno na konkrétní kapitoly učebnice pro dané téma, které jsou publikovány online [9] a byly vytvořeny jednotlivé varianty ŠVP. V odkazech na zdroje byly především použity učebnice vytvořené v rámci projektu PRIM proto, že byly recenzovány odbornými recenzenty a byly na školách dlouhodobě ověřovány.

Ve druhé etapě v listopadu a prosinci 2020 se pracovalo na harmonizování jednotlivých ŠVP s finální podobou již schválených RVP a mezitím změněnou hodinovou dotací na 2 + 4 (2 h na

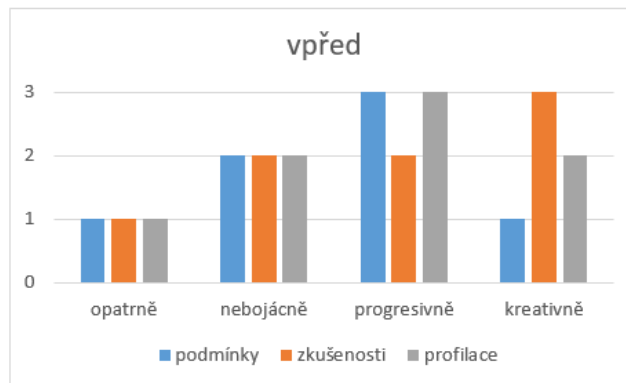
1. stupni, 4 h na 2. stupni). Byla též finalizována preambule, tedy charakteristika předmětu a jeho obsahové a organizační vymezení. Informatika je zde prezentována jako předmět, který dává prostor všem žákům porozumět tomu, jak fungují digitální technologie a informační systémy, přičemž hlavní důraz je kladen na rozvíjení žákova inženýrského myšlení. Škola se snaží rozvíjet digitální gramotnost v ostatních předmětech. Z organizačního pohledu je v řadě činností preferována práce žáků ve dvojicích u jednoho počítače, některá témata probíhají bez počítače. Výuka je orientována činnostně, s žákem, který objevuje, experimentuje, diskutuje, tvoří, řeší problémy, spolupracuje, konstruuje své poznání. Není kladen naprosto žádný důraz na pamětné učení a reprodukci.

3 VARIANTY ŠVP

Pro vytvoření jednotlivých variant modelových ŠVP [10] byly brány v potaz následující charakteristiky:

- podmínky – Jedná se především o vybavení a pomůcky pro výuku. Patří sem pracovní digitální zařízení, robotické hračky a stavebnice. Pokud je školy v danou chvíli nevlastní, počítají s jejich pořízením.
- zkušenost – Především jde o zkušenost s aktivním přístupem k tvorbě ŠVP a to jak na úrovni školy, tak na úrovni jednotlivých učitelů.
- profilace – Vyjadřuje, do jaké míry vnímá škola informatiku, resp. inženýrské myšlení jako svou profilující část, tedy do jaké míry klade důraz na témata jako robotika a programování či zda dokonce posiluje výuku z disponibilních hodin.

Minimální míra naplnění jednotlivých charakteristik u škol, jimž jsou jednotlivé varianty určeny, je zjednodušeně znázorněna na grafu níže. Osa y zobrazuje úroveň naplnění těchto charakteristik.



Obrázek 1: Charakteristiky škol pro varianty ŠVP

3.1 ŠVP Opatrně vpřed

Tento program je cílen na školy, které přistupují k reformě informatiky pozvolna, nemají s informatikou zkušenosti, jednak si netroufají vytvořit vlastní návrh ŠVP, jednak neplánují nebo nemohou investovat do nákupu učebních pomůcek v podobě různých robotů, robotických stavebnic či softwaru.

ŠVP pracuje s časovou dotací 2 + 4 (jedna hodina týdně od 4. do 9. ročníku) a je sestaven z témat, která vedou k naplnění RVP se stávajícím vybavením školy, tedy počítačová učebna a připojení k internetu.

3.2 ŠVP Nebojácně vpřed

Tento program je připraven pro školy, které mají zkušenosti, orientaci na informatiku a technologie vnímají jako jednu ze svých priorit a plánují investovat do nákupu učebních pomůcek v podobě různých robotů, robotických stavebnic či softwaru, tak aby výuka byla pestřejší a pro žáky atraktivnější. Netroufají si však na vytvoření vlastního ŠVP a tento návrh využijí s nějakými úpravami podle aktuální situace školy a nakoupených pomůcek.

ŠVP pracuje s časovou dotací 2 + 4 a je sestaven z tematických celků, které vedou k naplnění RVP pomocí nákupů robotických pomůcek a s využitím odpovídajících metodik a učebnic vytvořených v projektu PRIM. Příkladem rozšiřujících tematických celků je práce s robotickou hračkou Beebot a stavebnicí Lego WeDo, na 2. stupni práce se stavebnicí Lego Mindstorms a programovatelnou deskou Micro:bit.

3.3 ŠVP Progresivně vpřed

Tento program je určen školám, které berou informatiku jako jednu ze svých priorit, což chtějí kromě nákupu robotických pomůcek podpořit navýšením dotace hodin na informatiku nad minimální rámec stanovený RVP z disponibilních hodin a posunutím začátku výuky informatiky do nižších ročníků. Současně si škola není jistá se sestavením vlastního ŠVP a bere tento program jako návrh, který případně upraví podle svých podmínek a priorit.

ŠVP pracuje s posílenou dotací 3 + 5 s tím, že výuka začíná ve 3. ročníku a v 8. nebo v 9. ročníku je dotace informatiky 2 h týdně. Podle tohoto modelového programu se škola rozhodne, zda toto rozšíření realizuje v 2. pololetí 8. ročníku, nebo v 1. pololetí 9. ročníku. Rozšiřujícími činnostmi jsou zde práce s pomůckami Beebot, Emil 3 a 4, Lego WeDo, Lego Mindstorms, Micro:bit.

3.4 ŠVP Kreativně vpřed

Šablona pro školy, které si plánují sestavit svůj ŠVP samy. Obsahuje všechny tematické celky, použité v ostatních variantách. Pokud se škola rozhodne použít některou z učebnic vytvořených v rámci projektu PRIM, může převzít kompletní text, týkající se daného tematického celku.

Jednotlivé tematické celky nejsou přiřazovány konkrétním ročníkům, pouze jednotlivým stupňům vzdělávání. Tabulka každého tematického celku je uvedena jako celek; některé rozsáhlejší tematické celky, jako např. programování, jsou v ostatních ŠVP rozděleny na několik částí, aby mohly být poskládány do časového sledu tak, aby se střídaly s jinými tématy. Výuka tak může být tematicky pestrá a naplňovat znaky spirálového kurikula.

4 UČEBNÍ PLÁN

Skladba témat a jejich časová dotace vychází z několika premis a požadavků:

1. Vývoj má být kontinuální, proto při dotaci 2 hodiny na 1. stupni a 4 hodiny na druhém stupni jsou v učebním plánu témata rozložena od 4. do 9. ročníku vždy po jedné vyučovací hodině týdně.
2. Průměrný počet týdnů výuky je stanoven na 33, čemuž odpovídá i součet hodin v jednotlivých ročnících.
3. Doporučený počet hodin pro jednotlivá témata vychází z pilotního ověřování vzdělávacích materiálů.
4. Na základě zpětné vazby od učitelů jsou rozsáhlejší témata rozdělena na více částí, které se střídají s jinými tématy.
5. Je-li to možné, pak jsou témata realizovaná na počítači prostrídána tématy realizovatelnými tzv. unplugged.

Učební plán v modelových ŠVP má také podobu tabulky. Jednotlivé řádky představují ročníky, v nichž jsou vždy názvy témat a počet hodin, které doporučujeme tématu věnovat. Dále je za tématem identifikátor, zda je téma nutné k naplnění RVP. Tato informace slouží především těm školám, které si budou modelové ŠVP upravovat. Pokud totiž toto téma vynechají, musí ho nahradit jiným, pomocí něhož naplní očekávané výstupy k tomuto tématu přiřazené. Následující příznak rozlišuje témata, k jejichž realizaci je nutný počítač (resp. odpovídající digitální zařízení). Pokud v tomto sloupci příznak A uveden není, pak k realizaci tématu počítač nezbytný není, ale nevylučuje se jeho použití jako doplňku. Poslední sloupec pak indikuje témata, k jejichž realizaci je obvykle nutné pořídit pomůcky. Jedná se o témata spojená s robotikou či nákupem licence specifického softwaru.

Tabulka 1: Příklad části učebního plánu

roč.	téma	hodin	nutné k naplnění RVP	je třeba počítač	nutný nákup pomůc.
...
6.	Kódování a šifrování inf. Práce s daty Informační systémy Programování 1	9 10 3 11	A A A A	A A A	
...

4.1 POPIS TÉMATU

Struktura tabulky byla využita také pro vymezení jednotlivých témat. Tabulka každého tématu obsahuje očekávané výstupy, učivo, edukační zdroje a výukové metody a formy.

Tabulka 2: Příklad rozpisu tématu

Úvod do modelování pomocí grafů a schémat

Tematický celek RVP Data, informace a modelování	
Očekávané výstupy RVP Žákyně/žák: <ul style="list-style-type: none"> • popíše konkrétní situaci, určí, co k ní již ví, a znázorní ji • vyčte informace z daného modelu 	Očekávané výstupy ŠVP Žákyně/žák: <ul style="list-style-type: none"> • pomocí grafu znázorní vztahy mezi objekty • pomocí obrázku znázorní jev • pomocí obrázkových modelů řeší zadané problémy
Zdroje Metodika Základy informatiky pro 1. stupeň ZŠ https://imysleni.cz/ucebnice/zaklady-informatiky-pro-1-stupen-zs	
Učivo Graf, hledání cesty Schémata, obrázkové modely Model	Odkaz na učivo ve zdrojích Grafové modely Další grafové modely Řešení problémů pomocí modelů
Výukové metody a formy Diskuse, badatelské aktivity, problémová výuka, práce ve dvojicích či skupinách	

Název tematického celku obsaženého v RVP jednak umožňuje lepší identifikaci příslušné skupiny očekávaných výstupů v dokumentu, jednak napovídá učivo, ke kterému se dané téma váže.

Očekávané výstupy jsou rozděleny na dvě úrovně. Při formulaci těchto školních byl kladen důraz na konkretizaci a zároveň hrála důležitou roli snaha neopakovat formulace z RVP. Výstupy nejsou párovány. Očekávané výstupy nevnímáme izolovaně každý zvlášť, ale vždy jako provázané prvky. U některých lze vysledovat jejich příslušnost, některé procházejí více tematickými celky. Výchozí inspirací formulací byly mimo jiné zdroje uvedené o řádek níže.

Pokud bylo nalezeno a doporučeno více zdrojů pro jedno téma, pak jsou tyto rozlišeny velkými písmeny. A stejná písmena jsou pak použita v následujícím řádku, který přiřazuje identifikovanému učivu přímo kapitoly či části v jednotlivých zdrojích.

V neposlední řadě pro zdůraznění konstruktivistického přístupu k výuce jsou pojmenovávány také metody a formy, k nimž by měly vybrané zdroje učitele vést.

5 VZDĚLÁVACÍ ZDROJE

Vzdělávací materiály i modelové ŠVP byly vytvářeny jako metodická podpora pro školy právě v rámci revize RVP. Projekt PRIM měl za cíl vytvořit podporu pro rozvíjení inforatického myšlení, ovšem ne každý tematický celek nového RVP je takto zaměřen. Některá témata proto nejsou zdroji z PRIM pokryta. Jde o témata:

- Ovládání digitálního zařízení (4. ročník),
- Počítače (6. ročník),
- Hromadné zpracování dat (8. ročník),
- Digitální technologie (9. ročník).

Tato témata jsou ve výuce ICT tradiční, obsažená v dosavadních RVP. Je tedy předpoklad, že učitelé pro tato témata již edukační zdroje mají. Nicméně pokud jsme našli učebnici nebo materiál, který by odpovídal základnímu přístupu k výuce, předpokládající aktivního žáka, který není založen na výkladu, uvedli jsme ho mezi zdroji také, i když nebyl vytvořen v rámci projektu PRIM.

6 SHRNUŤÍ NA ZÁVĚR

Všechny vytvořené varianty ŠVP jsou umístěny na webu imysleni.cz v sekci ŠVP [10]. U každé varianty je uveden její účel tak, aby si školy mohly správně vybrat. Všechny materiály jsou k dispozici ke stažení i v editovatelné formě proto, aby školy, které je chtějí převzít, je nemusely přepisovat.

Vytvořené ŠVP dávají dobrou představu, ke kterým tematickým celkům ještě nejsou odpovídající materiály vytvořeny a jakým směrem je třeba napnout síly k jejich doplnění, aby celé kurikulum nové informatiky bylo plně pokryto. Ani modelové ŠVP, ani vzdělávací zdroje nepovažujeme za dogma. Věříme, že se najde nemálo dostatečně kreativních učitelů s dostatečným vhladem a porozuměním tématu, kteří si budou vytvářet vlastní.

Současně si uvědomujeme, že pestrost především robotických pomůcek narůstá a budou vznikat další alternativní témata pro použití konkrétních pomůcek. Věříme, že zdroje uvedené v těchto ŠVP povedou autory nových materiálů k tomu, aby se držely linie didaktické kvality již zveřejněných materiálů nebo se ji snažily překročit.

BIBLIOGRAFICKÉ ODKAZY

- [1] MŠMT. Rámcový vzdělávací program pro základní vzdělávání, aktualizovaný text s účinností od 1. 9. 2017 [online]. Praha: MŠMT, 2017. [vid. 10. 1. 2021]. Dostupné z https://www.msmt.cz/file/43792_1_1/.
- [2] MŠMT. Rámcový vzdělávací program pro základní vzdělávání, vzdělávací oblast Informatika, s účinností od 1. 9. 2021. Praha: MŠMT, 2021.
- [3] PRIM. *Inforatické myšlení* [online]. Web projektu PRIM. České Budějovice: Pedagogická fakulta Jihočeská univerzita, © 2018. [vid. 9. 1. 2021]. Dostupné z <https://imysleni.cz/>.
- [4] Zákon č. 561/2004 Sb. o předškolním, základním, středním, vyšším odborném a jiném vzdělávání (školský zákon) v aktuálním znění
- [5] ČŠI. *Kvalita a efektivita vzdělávání a vzdělávací soustavy: Výroční zpráva 2019/2020* [online]. Praha: Česká školní inspekce, 2020. ISBN 978-80-88087-43-4. [vid. 12. 1. 2021]. Dostupné z https://www.csicr.cz/Csicr/media/Prilohy/PDF_el._publikace/V%c3%bdro%c4%8dn%c3%ad%20zpr%c3%a1vy/Vyrocn%C3%AD-zprava-Ceske-skolni-inspekce-2019-2020_zm.pdf.
- [6] PRIM. *O projektu* [online]. České Budějovice: Pedagogická fakulta Jihočeská univerzita, © 2018. [vid. 9. 1. 2021]. Dostupné z <https://imysleni.cz/o-projektu>.
- [7] BERKI, J. a J. DRÁBKOVÁ. *Základy inforaticky pro 2. stupeň základní školy* [online]. Liberec: Technická univerzita v Liberci, 2020. [vid. 11. 1. 2021]. ISBN 978-80-7494-521-2. Dostupné z <https://imysleni.cz/ucebnice/zaklady-inforaticky-pro-zakladni-skoly>.
- [8] NÚV. *RVP v oblasti Inforaticky a ICT* [online]. Národní ústav pro vzdělávání, © 2011–2021. [vid. 12. 1. 2021]. Dostupné z <http://www.nuv.cz/t/revize-rvp-ict>.
- [9] PRIM. *Učebnice a vzdělávací materiály pro školy* [online]. Web projektu PRIM. České Budějovice: Pedagogická fakulta Jihočeská univerzita, © 2018. [vid. 9. 1. 2021]. Dostupné z <https://imysleni.cz/ucebnice>.
- [10] PRIM. *Modelové školní vzdělávací programy* [online]. Web projektu PRIM. České Budějovice: Pedagogická fakulta Jihočeská univerzita, © 2018. [vid. 9. 1. 2021]. Dostupné z <https://imysleni.cz/svp>.

Ked' učiteľ vyrastie na žiaka – vzdelávanie učiteľov ZŠ a SŠ v regióne Banskej Bystrice

When teacher grows into pupil – education of secondary school teachers in region of Banská Bystrica

Patrik Voštinár

Katedra informatiky FPV UMB
Tajovského 40
974 09 Banská Bystrica
Slovensko
patrik.vostinar@umb.sk

Miroslav Melicherčík

Katedra informatiky FPV UMB
Tajovského 40
974 09 Banská Bystrica
Slovensko
miroslav.melichercik@umb.sk

ABSTRACT

Computer science belongs to the subjects that are continuously evolving. Teachers who teach it in primary and secondary schools have to constantly learn new technologies, tools and programming languages to engage attention of their students. For this reason, we have established a Computer Science Teachers' Club at the Department of Computer Science FNS UMB, which organizes educational meetings for computer science teachers within the Banská Bystrica region. In this article, we summarize activities, which were organized in the first year of its activity in real and online space.

Keywords

Teaching computer science. Programming. MakeCode. Micro:bit. Arduino. Computer Science Teachers' Club.

ABSTRAKT

Informatika patrí medzi predmety, ktoré sa neustále vyvíjajú. Učelia, ktorí ju učia na základných a stredných školách sa musia neustále učiť nové technológie, nástroje a programovacie jazyky, aby dokázali zaujať svojich žiakov. Z tohto dôvodu sme na Katedre informatiky FPV UMB založili Klub učiteľov informatiky, v rámci ktorého sme začali robiť vzdelávacie aktivity pre učiteľov informatiky v rámci banskobystrického kraja. V príspevku sumarizuje činnosti, ktoré boli zrealizované počas prvého roku jeho činnosti v prezenčnej alebo online forme.

Kľúčové slová

Vyučovanie informatiky. Programovanie. MakeCode. Micro:bit. Arduino. Klub učiteľov informatiky.

1 ÚVOD

Na vzdelávanie za účelom zvyšovania digitálnej gramotnosti žiakov na základných ako aj na stredných školách sa v poslednej dobe kladie veľký dôraz. V zoznamoch kompetencií či zručností v 21. storočí (OECD, UNESCO, Európska komisia, prípadne rôznych národných a medzinárodných učiteľských profesijných spoločností) sa vyskytuje ako jedna z položiek digitálna gramotnosť, prípadne informačná gramotnosť alebo zručnosti v práci s informáciami a IKT [1]. Dôležitosť tejto témy sa ukázala ako obzvlášť aktuálna po vypuknutí pandémie COVID-19, kedy žiaci zostali doma odkázaní na diaľančné a online vzdelávanie. Avšak popri tom nemožno zabúdať na samotných učiteľov, ktorí zohrávajú v tomto procese významnú a nenahraditeľnú úlohu. Čerství absolventi učiteľského štúdia majú dobrý prehľad v moderných pedagogických metódach a pomôckach, ale pri tak dynamicky sa vyvíjajúcej oblasti, akou je informatika, môže rýchlo

nastať situácia, že prestanú stíhať sledovať nové trendy, a tak sa pripravujú o mnohé atraktívne spôsoby vzdelávania a zvyšovanie motivácie žiakov k štúdiu v oblasti STEM.

Na základe podnetov od učiteľov informatiky zo základných a stredných škôl v banskobystrickom regióne sa ukázalo, že učelia nemajú k dispozícii dostatok prostriedkov a času, aby sa mohli kontinuálne oboznamovať s novými trendmi vo vzdelávaní informatiky a s modernými didaktickými pomôckami. Preto sme sa po vzore viacerých pracovísk, ako aj z iných aprobačných predmetov, na Katedre informatiky, Fakulty prírodných vied, Univerzity Mateja Bela v Banskej Bystrici, ako nositelia moderného informatického vzdelávania v regióne podujali na vytvorenie takej platformy, ktorá by tento stav dokázala zlepšiť. Začiatkom roku 2020 sme založili Klub učiteľov informatiky [2] na FPV UMB, ktorého hlavnými cieľmi sú:

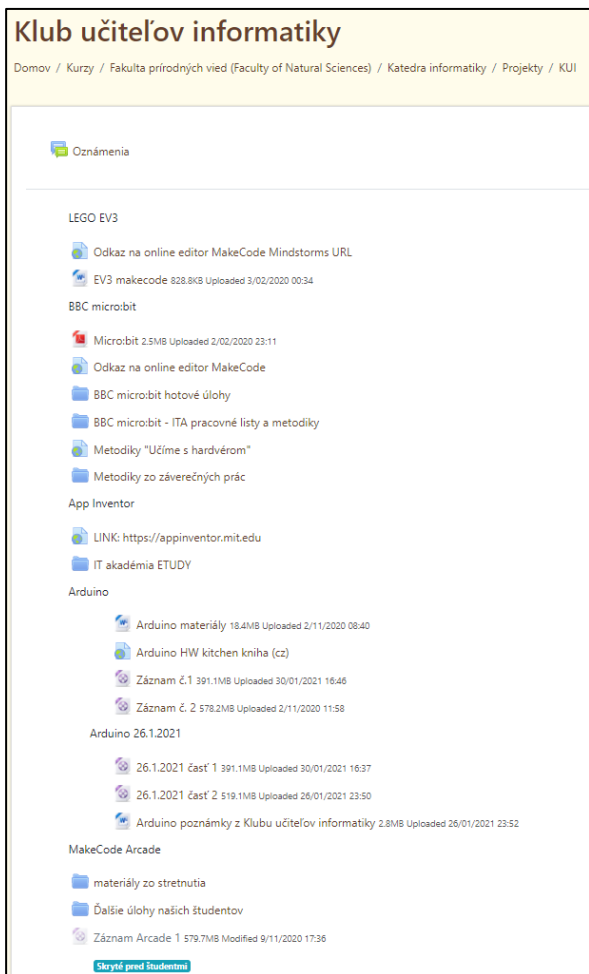
- podporovať učiteľov informatiky pri ich pedagogickom raste,
- podporovať rozširovanie informácií o nových trendoch v oblasti informatického vzdelávania na Slovensku a vo svete,
- združovať učiteľov informatiky na ZŠ a SŠ a vytvoriť priestor pre diskusiu a výmenu poznatkov a skúseností s využívaním inovatívnych didaktických prístupov a didaktických pomôcok v oblasti informatiky.

2 ČINNOSŤ KLUBU UČITEĽOV INFORMATIKY

Počas prvého roku svojej činnosti sa klub zamerával najmä na poskytovanie podpory učiteľov informatiky vo forme pravidelných stretnutí, inšpirovaných Klubom učiteľov informatiky UPJŠ v Košiciach, ktorí organizovali stretnutia v rámci národného projektu IT Akadémia – Vzdelávanie pre 21. storočie [3]. Doposiaľ sa uskutočnilo 7 stretnutí, na ktorých sa celkom zúčastnilo 76 učiteľov informatiky z banskobystrického regiónu. Konkrétne boli venované týmto témam (uvádzame v chronologickom poradí):

- programovanie v prostredí MakeCode (micro:bit, LEGO EV3),
- programovanie v prostredí MakeCode (micro:bit, LEGO EV3) – opakované pre veľký záujem,
- rôzne didaktické pomôcky zamerané na programovanie,
- programovanie mobilných aplikácií v App Inventore,
- programovanie mikrokontroléra Arduino – zvuk,
- programovanie 2D hier v prostredí MakeCode Arcade,
- programovanie mikrokontroléra Arduino – displeje.

Keďže nás pandémie prinútila plne prejsť do online priestoru, niektoré stretnutia nebolo možné konať prezenčne, ale formou videokonferenčného hovoru s použitím rôznych simulátorov. Záznamy z týchto stretnutí sú pre ďalších asi 20 dodatočných záujemcov spoločne aj s ďalšími materiálmi dostupné v LMS Moodle¹ (Obrázok 1). Všetkým účastníkom je na konci stretnutia vystavený certifikát potvrdzujúci absolvovanie školenia v rozsahu 4 vyučovacích hodín.



Obrázok 1: LMS kurz s materiálmi zo stretnutí Klubu učiteľov informatiky na UMB

3 PREZENČNÉ STRETNUTIA

Prvé štyri stretnutia sa uskutočnili prezenčne, pričom v troch prípadoch sa školenie konalo priamo na Katedre informatiky FPV UMB v Banskej Bystrici. Jedno stretnutie bolo zorganizované na požiadanie v Žiari nad Hronom.

Po skončení každého stretnutia bola získavaná spätná väzba od účastníkov formou neformálneho dialógu.

3.1 Micro:bit a Lego EV3

Prvé stretnutie Klubu učiteľov informatiky v Banskej Bystrici bolo zamerané na čoraz populárnejšie prostredie MakeCode od spoločnosti Microsoft. MakeCode poskytuje používateľom

prostredie na základy programovania vo vizuálnom blokovom programovaní, skriptovacím jazyku JavaScript a Python [4].

Hlavným cieľom školenia bolo zoznámiť učiteľov s edukačnou pomôckou micro:bit a tiež ukázať alternatívne prostredie k zariadeniu Lego EV3. V rámci školenia sme chceli tiež ukázať vytvorené metodiky a návody, ktoré môžu učители používať v rámci vyučovacieho procesu.

V rámci prvých dvoch hodín sa učители zoznámili s edukačnou pomôckou micro:bit a jeho rôznymi rozšíreniami – Neopixel LED pásik, pripojenie reproduktora a herného ovládača. V závere boli spomenuté ďalšie zaujímavé rozšírenia, ktoré sú vhodné pre pokročilejšiu prácu s edukačnou pomôckou micro:bit.

V druhej časti stretnutia sa učители zoznámili počas 2 hodín s prostredím MakeCode pre populárnu didaktickú pomôcku LEGO EV3. Prostredie MakeCode pre EV3 obsahuje simulátor LEGO kocky a možnosť programovania v blokoch kódu, JavaScripte a jazyku Python [5]. Všetci učители uviedli, že majú skúsenosti s programovaním LEGO EV3 v oficiálnom prostredí LEGO, nikto z nich však nepracoval s prostredím MakeCode, ktoré je veľmi podobné blokovému prostrediu Scratch.

Prvého stretnutia sa zúčastnilo 16 učiteľov z banskobystrického kraja – väčšinou so základných škôl, alebo osemročných gymnázií (Obrázok 2).

Počas školenia sme dostali viacero pozitívnych spätných väzieb od učiteľov na vytvorené aktivity a tiež na použité pomôcky. Niektorí učители poznali micro:bity, všetci poznali LEGO EV3, avšak nie prostredie MakeCode. Viacerí učители sa vyjadrili, že sa pokúsia zaobstarať pomôcky micro:bity do škôl.



Obrázok 2: Prvé stretnutie zamerané na LEGO EV3

Na základe veľkého záujmu sa zrealizovalo opakované stretnutie pre ďalších 11 učiteľov vo februári 2020 (Obrázok 3).

¹ <https://lms.umb.sk/course/view.php?id=2772>



Obrázok 3: Programovanie LEGO EV3 počas druhého stretnutia

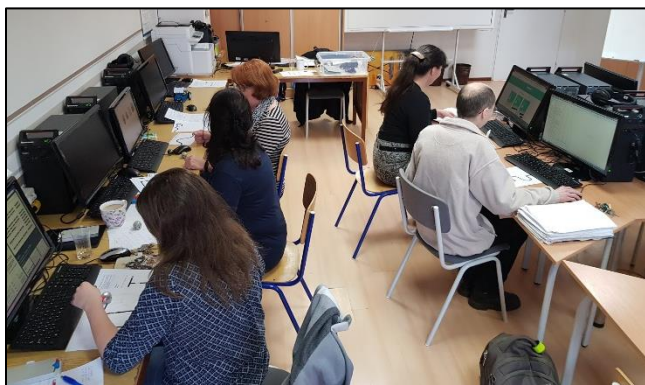
Spätná väzba na základe rozhovorov na konci stretnutia bola od učiteľov veľmi pozitívna, niektorí sa vyjadrili dodatočne aj mailovou komunikáciou.

3.2 Rôzne didaktické pomôcky zamerané na programovanie

Na základe pozitívnej odozvy z prvého stretnutia bol organizátor Dr. Patrik Voštinár požiadaný o zrealizovanie Klubu učiteľov informatiky na Gymnáziu Milana Rúfusa v Žiari nad Hronom pre učiteľov zo základných škôl v okrese (Obrázok 4).

Cieľom stretnutia bolo predstaviť inovatívne pomôcky pre výučbu informatiky, ktoré sú aktuálne používané vo svete. Na tomto stretnutí sa učitelia zoznámili s inovatívnymi pomôckami zameranými na programovanie – edukačnou pomôckou micro:bit a jeho rozšíreniami (LED pásik, reproduktor, herný ovládač, micro:bit autíčko, roboti Phiro Pro, mBot, Ozobot EVO a dosková hra Scottie Go!).

Stretnutie v Žiari nad Hronom malo pozitívne ohlasy od učiteľov, všetci sa vyjadrili, že sa im pomôcky páčili, len si nevedeli vybrať, čo by z toho najviac chceli.



Obrázok 4: Klub učiteľov informatiky v Žiari nad Hronom

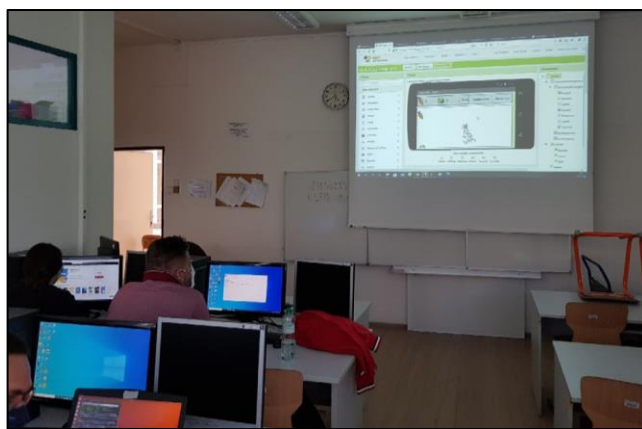
3.3 Programovanie v App Inventor

Počas pandémie Covid-19 a uzatvorenia škôl sa stretnutia Klubu učiteľov informatiky v BB prerušili do septembra 2020, kedy sa zorganizovalo posledné prezenčné stretnutie na programovanie mobilných aplikácií a hier v online prostredí App Inventor. Tohto

stretnutia sa vzhľadom na začínajúcu druhú vlnu pandémie COVID-19 nakoniec zúčastnili iba 4 učitelia z pôvodne prihlásených 15 učiteľov (na poslednú chvíľu sa odhlasovali, či už z dôvodu hroziaceho nakazenia chorobou alebo na pokyn riaditeľov škôl).

Cieľom školenia bolo ukázať online nástroj App Inventor ako možný nástroj na výučbu programovania mobilných aplikácií a tiež predstaviť inovatívne metodiky, ktoré v rámci projektu IT akadémie vznikajú. V rámci tohto stretnutia sa učitelia zoznámili s prostredím App Inventor, vytvorili si viacero zaujímavých aplikácií a hier, ktoré boli pripravené špeciálne pre toto stretnutie a taktiež aj metodiky, ktoré boli vytvorené počas riešenia národného projektu IT akadémie (Obrázok 5).

Na základe rozhovorov a pozorovania môžeme skonštatovať, že prostredie App Inventor sa učiteľom páčilo a podľa ich vyjadrení ho budú používať vo vyučovaní.



Obrázok 5: Programovanie mobilných aplikácií

Všetky materiály sú dostupné na stránke Klubu učiteľov informatiky BB², vybrané metodiky z projektu IT akadémie v dobe použitia neboli ešte publikované.

4 ONLINE STRETNUTIA

Ostatné tri stretnutia v tomto školskom roku sa konali v online priestore prostredníctvom aplikácie MS Teams, pričom účastníci sa k nim pripájali zo svojho domova alebo školy. Vzhľadom k tejto skutočnosti bolo potrebné prispôbiť obsah školení s ohľadom na to, že účastníci nemusia mať pri sebe potrebné hardvérové vybavenie. Ako vhodné riešenie sa ukázalo použitie rôznych online simulátorov.

Spätnú väzbu od účastníkov po skončení školenia sme získavali formou online dotazníka.

4.1 Programovanie mikrokontroléra Arduino

V októbri 2020 sa zrealizovalo prvé online stretnutie Klubu učiteľov informatiky na tému programovanie mikrokontroléra Arduino. Toto stretnutie bolo zamerané hlavne na stredoškolských učiteľov. Stretnutie sa zúčastnilo 11 učiteľov online a ďalším 6 bol poslaný záznam zo stretnutia, nakoľko v danom čase sa nemohli zúčastniť stretnutia kvôli ich výučbe.

Jedným z dôvodov vytvorenia veľmi populárnej pomôcky – mikrokontroléra Arduino bolo jeho využitie pri vzdelávaní. Arduino je hardvérové zariadenie, ku ktorému sa pripájajú rôzne

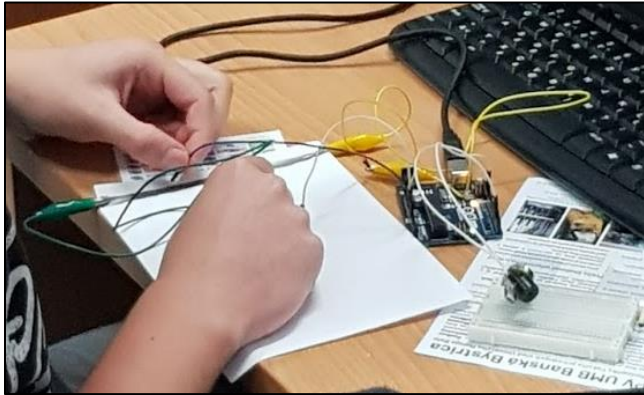
² <https://lms.umb.sk/course/view.php?id=2772>

zariadenia ako LED svetlá, tlačidlá, senzory [6,7]. Vzhľadom na to, že Arduino je fyzické zariadenie, ktoré nemusia mať všetci učители k dispozícii doma, rozhodli sme sa pri realizácii školenia použiť online simulátor tinkercad.com, ktorý je možné použiť aj na modelovanie 3D objektov. Okrem tejto funkcionality však obsahuje aj možnosť zapojiť, naprogramovať a odskúšať rôzne súčiastky a moduly s mikrokontrolérom Arduino. Vytvorené projekty na tejto stránke je možné následne otestovať v online simulátore. Tým pádom je možné vyučovať túto oblasť informatiky aj dištančne, bez nutnosti mať Arduino sadu fyzicky k dispozícii.

Pre tieto školenie sme si stanovili niekoľko cieľov – oboznámiť učiteľov s mikrokontrolérom Arduino a možnosťami jeho uplatnenia vo výučbe, prezentovať učiteľom vybrané úlohy pre začiatočníkov a pokročilých študentov vo väzbe na STEM a predstaviť simulátor na stránke Tinkercad, ako vhodnú alternatívu pre prácu s mikrokontrolérom Arduino počas dištančnej formy výučby.

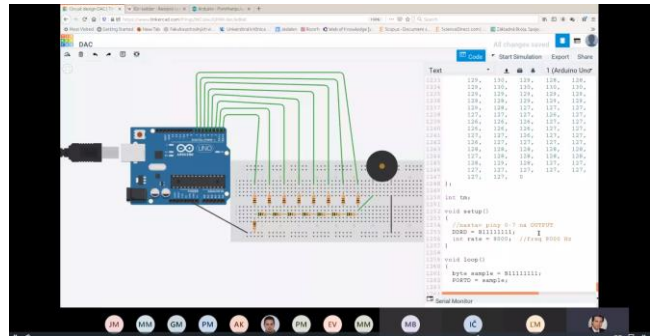
4.1.1 Generovanie zvuku

Počas prvého stretnutia si učители v prvých 2 hodinách vyskúšali prostredie Tinkercad, zoznámili sa s Arduino, ako ho používať – zapájať LED svetlá, bzučiak a potenciometer. Princíp potenciometra sme ukázali ako ukážku zaujímavej aktivity na samostatnom projekte, kde sme na obyčajný papier nakreslili ceruzkou (grafitom) čiaru. K jednému koncu sme pripli krokosvorkový kábel napojený k Arduino pinu GND a druhý koniec papiera sme prepojili krokosvorkovým káblom k pinu 5V na Arduino. Tretím vodičom pripojeným k analógovému vstupu Arduino sme pohybovali po nakreslenej odporovej dráhe. Načítaná hodnota na analógovom vstupe zodpovedala frekvencii tónu vydávaného bzučiakom (Obrázok 6).



Obrázok 6: Vytvorenie vlastného potenciometra

V ďalších dvoch hodinách sme ukázali úlohu vhodnú pre pokročilejších používateľov – 8-bitový DA prevodník, pomocou ktorého sme prehrávali nahratý a digitalizovaný zvuk (Obrázok 7).

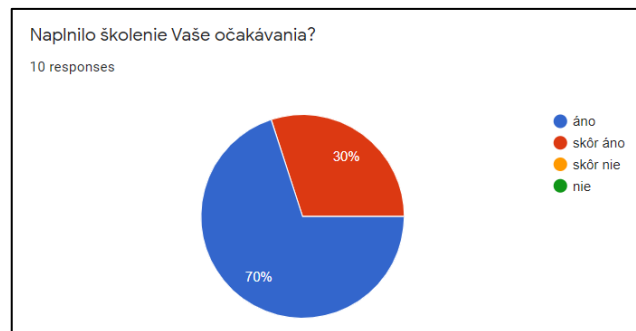


Obrázok 7: Stránka Tinkercad – Arduino simulátor

4.1.2 Vyhodnotenie online dotazníka

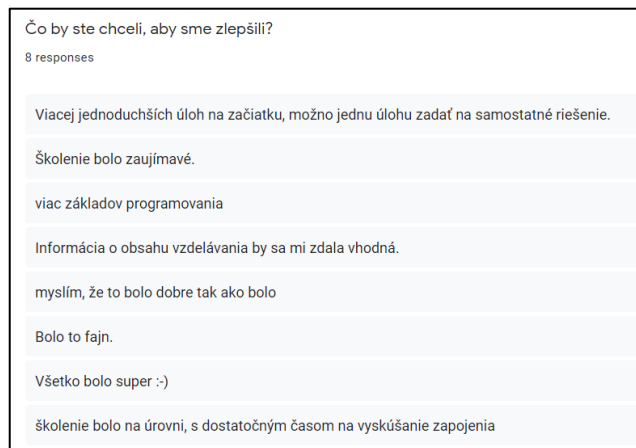
Po stretnutí sme dali učiteľom vyplniť dotazník, v ktorom sa mohli vyjadriť, či by chceli pokračovať v ďalšom stretnutí s ďalšími úlohami na Arduino a taktiež sme pomocou dotazníka získavali spätnú väzbu k stretnutiu.

Dotazník obsahoval 6 otázok. Na otázku, či školenie naplnilo očakávanie (Obrázok 8) odpovedalo 10 respondentov (jeden učiteľ neodpovedal), väčšina učiteľov odpovedala, že naplnilo, 3 skôr naplnilo.



Obrázok 8: Dotazníková otázka zo školenia na Arduino

Na otázku, čo by chceli zlepšiť pre ďalšie stretnutia odpovedalo iba 8 učiteľov (Obrázok 9).



Obrázok 9: Odpovede učiteľov, čo by chceli zlepšiť na školení s Arduino

Ďalšie otázky boli zamerané na to, či by chceli mať ďalšie pokračovanie na Arduino – 10 respondentov odpovedalo áno a či by chceli niečo odkázať lektorom – všetci ďakovali za stretnutie, dvaja odpovedali, že posledný príklad bol zaujímavý, ale náročný.

4.1.3 Programovanie LED displejov

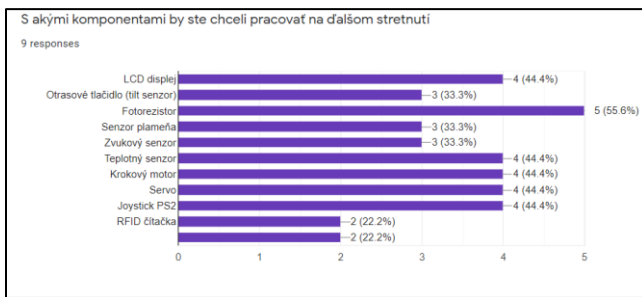
Na základe pozitívnej odozvy sme v januári 2021 zorganizovali ďalšie stretnutie – pokračovanie v programovaní mikrokontroléra Arduino. Na ďalšom stretnutí sme ukázali pokročilejšiu prácu s mikrokontrolérom – RGB LED, 7-segmentový displej, 8-bitový posuvný register 74HC595 a IR senzor s diaľkovým ovládačom. Hlavným cieľom bolo pomocou jednoduchých praktických príkladov oboznámiť účastníkov s princípom fungovania zobrazovacích LED modulov. Získané poznatky je možné využiť pri zostavovaní veľkoplšnej obrazovky z RGB LED panelov. V závere stretnutia boli načrtnuté možnosti vzájomného prepojenia použitých komponentov pre skúsenejších študentov v podobe realizácií zložitejšieho projektu, ako napríklad stopky ovládané diaľkovým ovládačom.

4.1.4 Vyhodnotenie online dotazníka

Po skončení stretnutia sme dali učiteľom opäť dotazník so 7 otázkami – či naplnilo školenie ich očakávania (odpovede boli identické ako pri rovnakej otázke po prvom stretnutí), či si myslia, že úlohy použijú vo vyučovacom procese, čo by chceli zlepšiť, či by chceli mať ďalšie pokračovanie s Arduino a ak áno, tak čo konkrétne by chceli vyskúšať. Dotazník vyplnilo 9 učiteľov.



Obrázok 10: Dotazníková otázka, či si učители myslia, že môžu použiť úlohy zo školenia vo vyučovaní



Obrázok 11: Otázka zameraná na ďalšie témy školenia s Arduino

Napriek náročnosti niektorých prezentovaných projektov s použitím mikrokontroléra Arduino je možné školenia považovať za úspešné. Pričom náročnejšie projekty je vhodné použiť pre nadaných študentov alebo počas krúžkovej činnosti.

Na základe dotazníka môžeme skonštatovať, že sa učiteľom školenia páčili a chcú pokračovať v podobných školeniach.

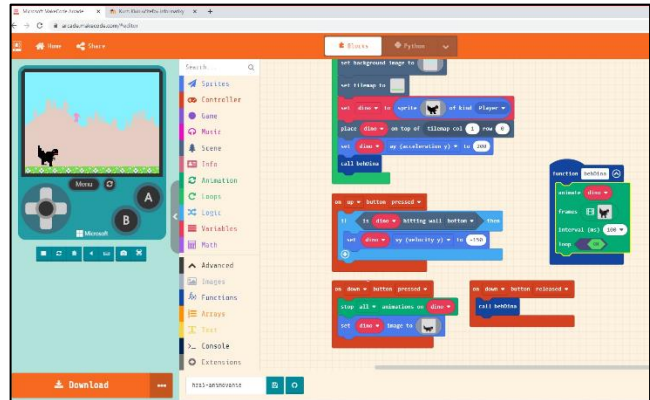
4.2 Programovanie 2D hier v MakeCode Arcade

Začiatkom novembra 2020 sa uskutočnilo ďalšie online stretnutie zamerané hlavne na učiteľov základných škôl na tému

programovanie 2D hier. Stretnutie sa zúčastnilo 15 učiteľov a ďalším 5 bol poslaný záznam zo stretnutia.

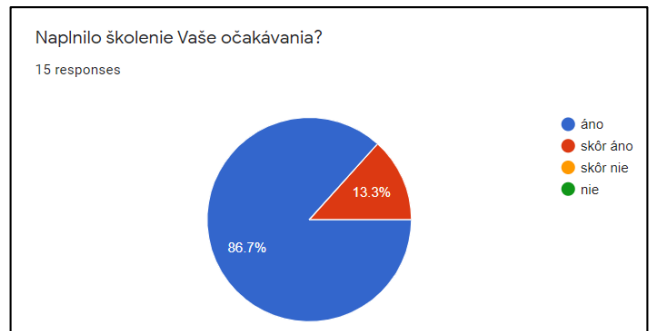
V rámci stretnutia sa učители zoznámili s prostredím MakeCode Arcade, ktoré umožňuje vytvárať jednoduché 2D hry v online prostredí. Cieľom školenia bolo predstaviť online nástroj MakeCode Arcade, ktorý umožňuje programovanie jednoduchých 2D hier aj počas dištančného vzdelávania.

Počas stretnutia si vytvorili 4 zaujímavé hry, ktoré naprogramovali v blokovom prostredí podobnom ako v prostredí Scratch. Na testovanie aplikácii sa používal simulátor hernej konzoly, niektorí učители si zdieľali vytvorené hry cez URL odkaz do svojich mobilných zariadení, ktoré použili na testovanie.



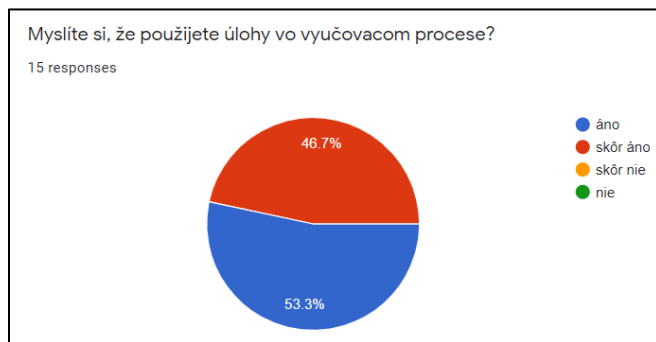
Obrázok 12: Prostredie MakeCode Arcade s ukážkou vlastnej hry

Po skončení stretnutia sme učiteľom dali vyplniť dotazník s 5 otázkami na získanie spätnej väzby. V prvej otázke sme sa pýtali, či naplnilo školenie ich očakávania. Všetci odpovedali, že áno, alebo skôr áno (2 ja respondenti)



Obrázok 13: Dotazníková otázka ku stretnutiu MakeCode Arcade

Druhou otázkou sme zisťovali, či použijú prezentované úlohy vo vyučovacom procese. 8 učiteľov odpovedalo áno a 7 skôr áno.



Obrázok 14: Dotazníková otázka, či si učители myslia, že použijú prostredie MakeCode Arcade vo vyučovacom procese

Otázkou „Čo by ste chceli, aby sme zlepšili?“ sme sa pýtali na možné vylepšenia stretnutia do budúcnosti. Z odpovedí vyplývalo, že 3 učители by chceli zvoľniť tempo (v odpovedi 2 učители uviedli, že im chýbal druhý monitor na sledovanie) a 9 učiteľov by nemienili nič.

Obrázok 15: Dotazníková otázka, čo by chceli zlepšiť učители v školení MakeCode Arcade

Predposledná otázka bola zameraná na záujem o pokračovanie školenia – všetci 15 učители odpovedali, že by chceli pokračovať.

Posledná otázka „Ak chcete odkázať niečo lektorom, môžete napísať sem (napr. k výberu úloh).“ bola informačná – zaujímalo nás, či chcú lektorom niečo odkázať.

Obrázok 16: Dotazníková otázka, či chcú učители niečo odkázať lektorom

Z odpovedí dotazníka vyplýva, že učiteľom sa prostredie páčilo, chceli by pokračovať v ďalších školeniach na programovanie 2D hier.

5 ZÁVER

V príspevku sme zhrnuli prehľad aktivít, ktoré sme zrealizovali na Katedre informatiky FPV UMB v rámci aktivít Klubu učiteľov informatiky pre učiteľov informatiky na ZŠ a SŠ v rámci banskobystrického kraja. V rámci Klubu učiteľov informatiky sme počas roka 2020 zrealizovali 7 stretnutí pre učiteľov informatiky. Témou týchto stretnutí boli inovatívne prostredia na výučbu programovania. Na základe spätnej od zúčastnených učiteľov sa ukázalo, že činnosť Klubu učiteľov informatiky má veľké opodstatnenie a prispieva k zvyšovaniu motivácie k štúdiu informatiky a zvyšovaniu digitálnej gramotnosti žiakov. V týchto a ďalších aktivitách plánujeme pokračovať aj v ďalšom období prezenčnou aj dištančnou formou.

POĎAKOVANIE

Tento článok bol vytvorený v rámci národného projektu IT Akadémia – vzdelávanie pre 21. storočie, ktorý sa realizuje vďaka podpore z Európskeho sociálneho fondu a Európskeho fondu regionálneho rozvoja v rámci Operačného programu Ľudské zdroje.

BIBLIOGRAFICKÉ ODKAZY

- [1] KALAŠ, I. a kol. *Premeny školy v digitálnom veku*. Bratislava: Slovenské pedagogické nakladateľstvo – Mladé letá, s.r.o., 2013, 256 s. ISBN 978-80-10-02409-4.
- [2] Klub učiteľov informatiky. <https://www.fpv.umb.sk/katedry/katedra-informatiky/klub-ucitelov-informatiky-6793.html>
- [3] IT AKADÉMIA. <http://itakademia.sk/>
- [4] DEVINE, J. and col. MakeCode and CODAL: Intuitive and Efficient Embedded Systems Programming for Education. In *ACM SIGPLAN Notices*. 2018, vol 53, no. 6, p. 19-30. VOŠTINÁR, P. MakeCode for Lego Mindstorms EV3. In *iJOE*. 2020, vol. 16, no. 14, p. 42-53.
- [5] VOŠTINÁR, P., MELICHERČÍK, M., VAGAČ, M. Teaching programming through the Arduino workshops at secondary schools. In *ICETA 2020 : 18th IEEE international conference on emerging eLearning technologies and applications*, New Jersey : IEEE, 2020, p. 749-755. ISBN 978-0-7381-2366-0.
- [6] SARIK, J., KYMISSIS, I. Lab kits using the Arduino prototyping platform. In: *2010 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2010. p. T3C-1-T3C-5.

Korona - akcelerátor informatizácie výučby

Korona - accelerator of computerization of education

Patrik Voštinár, Dana Horváthová, Ľudovít Trajtel', Jarmila Škrinárová, Alžbeta Michalíková

Katedra informatiky Fakulty prírodných vied Univerzity Mateja Bela

Tajovského 40

974 01 Banská Bystrica

Slovensko

patrik.vostinar@umb.sk, dana.horvathova@umb.sk, ludovit.trajtel@umb.sk, jarmila.skrinarova@umb.sk,
alzbeta.michalikova@umb.sk

ABSTRACT

In this article, we summarize all the activities that took place under the patronage of the IT Academy at the Department of Informatics, FPV UMB in Banská Bystrica. In the last demanding period, marked by the coronavirus pandemic, there were done a number of innovative methodologies for primary and secondary schools, several textbooks, several educational activities for teachers and school principals, as well as many other motivational and popularizing activities for pupils and students in cooperation with other IT departments in Slovakia at all levels of education. We want to highlight here the fact that, despite (or perhaps thanks to) difficult conditions, many teachers have been able to mobilize and take a drastic step towards better preparing the young generation for the knowledge society and the digital world.

Keywords

IT academy. Innovative methodologies. Informatics teacher education. Motivation of pupils and students.

ABSTRAKT

V tomto článku sa venujeme sumarizácii všetkých aktivít, ktoré sa pod patronátom IT Akadémie odohrávali na katedre informatiky FPV UMB v Banskej Bystrici. V uplynulom náročnom období, poznačenom pandemiou koronavírusu vzniklo v spolupráci s ďalšími informatickými pracoviskami na Slovensku množstvo inovatívnych metodík pre ZŠ a SŠ, niekoľko učebníc, viacero vzdelávacích aktivít pre učiteľov a riaditeľov škôl, ako aj veľa ďalších motivačných a popularizačných aktivít pre žiakov a študentov na všetkých stupňoch vzdelávania. Chceme tu vyzdvihnúť fakt, že aj napriek, (alebo možno vďaka) sťaženým podmienkam sa množstvo učiteľov dokázalo zmobilizovať a urobiť razantný krok smerom k lepšej príprave mladej generácie na vedomostnú spoločnosť a digitálny svet.

Kľúčové slová

IT akadémia. Inovatívne metodiky. Vzdelávanie učiteľov informatiky. Motivácia žiakov a študentov.

1 ÚVOD

Marec minulého roku nám priniesol nový fenomén tejto doby, ktorý radikálne zmenil naše životy, náš pohľad na svet a na naše potreby. Nás, ktorí pracujeme v oblasti vzdelávania, prinútil oveľa rýchlejšie ako kedykoľvek pred tým, reagovať na vzniknutú situáciu a prispôbiť sa novovzniknutým podmienkam. Prechodom výučby na všetkých stupňoch škôl do on-line prostredia, sme boli všetci postavení pred nemalo technických pedagogických, psychologických, didaktických, či iných výziev, ktoré sme museli veľmi rýchlo prekonať, aby sme čo najplynulejšie a najbezbolestnejšie zachránili, čo sa zachrániť dalo.

Zmenil sa náš životný štýl, zmenil sa trh práce a zmenili sa požiadavky zamestnávateľov na mladých absolventov škôl. To všetko sa premieta aj do rastúcej nutnosti meniť formálne vzdelávanie žiakov a študentov, starostlivo prehodnotiť akademický obsah vzdelávania a reagovať na celkom nové potreby na úrovni zručností pre produktívny život v spoločnosti. [1]

S odstupom času, takmer po jednom roku, s menšími, či väčšími prestávkami, sme zistili, že tie prekážky, ktoré na začiatku boli obrovské, sa dnes darí prekračovať s oveľa menším úsilím. Uvedomujeme si však aj katastrofálne dôsledky, aké to bude mať pre tých, ktorí na prekážku on-line vzdelávania narazili a nedokázali ju zdolať bez ujmy na „zdraví“. Pre všetkých, ktorí sa vzdelávaním informatiky zaoberajú, by mali byť práve výstupy z konferencie Didinfo (a snád' aj náš článok), zdrojom čerpania nápadov, inšpirácie a súhrnom aktivít, ktoré sme uskutočnili na pomoc pri prekonávaní tejto neľahkej doby.

Projekt IT akadémie je už od roku 2016 našou hybnou silou v inovácii vzdelávacích programov, metodík, učebných materiálov, ako aj v aktualizácii obsahu, metód a foriem výučby matematiky, informatiky, prírodovedných predmetov a odborných predmetov. [2]

Projekt sa snaží reflektovať na potreby spoločnosti, napomáha eliminovať nedostatok IT odborníkov a pripravuje študentov VŠ pre zamestnanie v IT sektore.

2 INOVATÍVNE METODIKY

Trojčlenný tím zostavovateľov Trajtel', E. (UMB) v spolupráci s Lovászovou, G. (UKF) a Hanesz, A. (UPJŠ) vytvoril svojim spôsobom jedinečné autorské, 507 stranové súborné dielo – zbičku celkom 48 inovatívnych metodík z informatiky, pričom autormi jednotlivých metodík boli: Tkáčová, Z. (18 metodík), Hanesz, A. (12), Tomcsányiová, M./Tomcsányi, P. (6), Trajtel', E./Jacková, J., resp. Michalíková, B. (6), Lovászová, G. (2), Cápaj, M. (2), Michalíková, V. (2 metodiky). [3]

„Zbička inovatívnych metodík z Informatiky pre druhý stupeň ZŠ a pre SŠ“ je koncipovaná ako pracovná kniha – učebnica, ktorú možno používať ako doplnok na vyučovacích hodinách informatiky. Jej úlohou je informovať učiteľa i jeho žiaka, podporiť výučbu a učenie sa všetkých zúčastnených. Základným školám je určených 30 metodík v celkoch: BBC micro:bit (3 metodiky); 3D modelovanie a tlač (3); Video (3); Lego EV3 (5); Kódovanie a šifrovanie (4); Spracovanie geolokačných dát (6); Spoločenské aspekty IKT (6 metodík). Stredným školám je určených 18 metodík v celkoch: Bezpečnosť (3 metodiky); Umelá inteligencia (3); Raspberry Pi (6); Programovanie v prostredí App Inventor 2.0 (6

metodík). Učebnica ponúka širokú paletu aktuálnych úloh, ktoré nenásilnou formou obohacujú hodiny informatiky, pútavo a zaujímavým spôsobom pomáhajú spestriť a obohatiť prácu učiteľa i žiakov. Široká v predkladaných celkoch je nielen samotná ponuka úloh, ale ku každému celku sú k dispozícii pracovné listy so skupinami aktivít, príkladov a postupov na precvičovanie učiva, prípadne jeho zhrnutie a opakovanie, tiež hodnotenie alebo samohodnotenie. Keďže učebnica je učiteľovi k dispozícii digitálne, nie je nevyhnutné, aby sa jednotlivé úlohy z nej používali výhradne iba pri frontálnom, osobnom, kontakte so žiakom a triedou, ale aj na individuálnu prácu žiakov, napríklad formou domácich úloh počas dištančne vedenej výučby.

3 UČEBNICE

V rámci aktivity 1.1 *Inovácia prírodovedného a technického vzdelávania na ZŠ a SŠ so zameraním na informatiku a IKT* sa riešitelia na Katedre informatiky FPV UMB podieľali na vytváraní viacerých učebníc pre základné a stredné školy.

3.1 Aktivita 1.1 – Tvorba predmetu Tvorba a prezentácia dát

Predmet sa zameriava na aktuálne trendy v oblasti tvorby statických webových stránok s dôrazom na analýzu obsahu vytváranej webovej stránky a dodržiavanie aktuálnych štandardov. Pod názvom Tvorba a prezentácia dát vznikla aj učebnica pre SŠ, v ktorej sa autorský kolektív Bezáková, D., Horváthová, D., Hrušecká, A., Hrušecký, R., Jašková, E., Tomcsányiová, M., a Voštinár, P. zaoberá tvorbou webu, obsahom stránky, jej vzhľadom, tvorbou a prezentáciou multimediálnych dát a ďalšími súvisiacimi témami, ktoré najlepšie vystihuje Obsah učebnice Obrázok 1.

OBSAH	
Zopár dôležitých informácií pre učiteľa	6
1 Úvod do tvorby webových stránok	10
2 Štruktúrovanie textu na webovej stránke	18
3 Úvod do kaskádových štýlov	26
4 Obrázky	32
5 Zoznamy s odrážkami a číslované zoznamy	42
6 Odkazy a navigácia	52
7 CSS - vlastnosti písma a textu, pseudotriedy	64
8 Blokové prvky na webovej stránke a ich štýlovanie	74
9 Tabuľky	92
10 Rozmiestňovanie objektov	108
11 Responzívny web a štýly pre rôzne zobrazovacie zariadenia, štýl pre tlač	130
12 Formuláre	148
13 Validita	168
14 Publikovanie a správa obsahu webovej stránky	180
15 Osoby so špecifickými potrebami a WEB	186
16 Prístupnosť INFORMÁCIÍ na webe	194
17 Tvorba prístupných webových stránok	204
18 Použiteľnosť webu	218
19 Multimédiá	224
20 Tvorba vektorovej grafiky a metagrafiky v editore Inkscape	230
21 Tvorba videa	238
22 Multimédiá na webe	250
23 Programovanie multimédií v jazyku Python	274
Index obrázkov, grafov a tabuliek	281
Bibliografia	289

Obrázok 1: Obsah učebnice „Tvorba a prezentácia dát“

3.2 Aktivita 1.1 – Tvorba predmetu Programovanie webových stránok

Do procesu tvorby predmetu Programovanie webových stránok sa v aktivite 1.1 sa zapojili členovia KI (Melicherčík, M., Voštinár P., ako spoluautori). V rámci tejto aktivity vznikla 228 stranová učebnica s názvom *Programovanie webových stránok*. Učebnica,

ktorej obsah je na obrázku 2, sa zameriava na vyučovanie vytvárania dynamických webových stránok v jazyku PHP a skriptovacím jazyku JavaScript.

Obsah	
1. Dynamické webové stránky	10
2. PHP – základná syntax a štruktúra	20
3. PHP – spracovanie formulárov (1)	46
4. PHP – základná syntax a štruktúra	58
5. PHP – spracovanie formulárov (2)	80
6. PHP – vkládanie kódu z rôznych súborov	102
7. PHP- trvalé ukladanie údajov	114
8. PHP – pamätanie údajov	124
9. Úvod do jazyka JavaScript	130
10. JavaScript – základná syntax a štruktúra	138
11. JavaScript – objektový model dokumentu	188
12. JavaScript – udalosti	194
13. JavaScript – formuláre	202
14. JavaScript – testovanie a ladenie	210
15. JavaScript – pokročilé techniky	220

Obrázok 2: Obsah učebnice „Programovanie webových stránok“

3.3 Aktivita 1.1 – Počítačové systémy a siete

Kolektív autorov Švec, P., Koprda, Š., Škrinárová Š. a Siládi, V. vytvoril učebnicu *Počítačové systémy a siete*. Učebnica je určená pre učiteľov a študentov v predmete informatika pre informatické triedy na stredných školách. Obsahuje množstvo príkladov aj výkladových častí, ktoré môžu učitelia priamo použiť na hodinách informatiky. Obsahové zameranie a nosné témy učebnice sú: architektúra počítačov a mikrokontroléry, uchovávanie, spolupráca počítača s periférnymi zariadeniami, zálohovanie a archivácia dát, operačné systémy, virtuálne počítače a virtualizačné nástroje, počítačové siete a protokoly a bezpečnosť v počítačových sieťach.

CVIČENIE 16 – POUŽITE!

Príklad 16: Vytvorme skript, ktorý vypíše obsah aktuálneho adresára pomocou príkazu s pevným počtom opakovaní. Majme skript s názvom príklad4 (pozri obrázok 22).

```
for i in *
do
echo $i
done
```

Obrázok 22
Obsah skriptu s cyklom s pevným počtom opakovaní

V skripte používame cyklus `for`. [6]

Syntax:

```
for premená in zoznam
do
príkazy
done
```

Pomocou príkazu `for` sa prehrádzajú všetky položky definované v zozname. Tieto položky sa postupne (pri každom opakovaní cyklu) priradia do premennej. Príkazy, ktoré sa vykonávajú v cykle, medzi výrazmi `do` a `done`, urobia určitú operáciu s položkou so zoznamu. V našom prípade (pozri obrázok 22) * nahradza zoznam všetkých položiek v aktuálnom adresári a následne urobí príkaz s každou položkou tak, že položku vypíše. Postupne vypíše všetky položky.

Spustíme skript s názvom príklad4 a výpis skriptu, ktorý vypisuje všetky položky aktuálneho adresára vidíme na obrázku 23.

```
príklad1
príklad2
príklad3
príklad4
zoznam4
zoznam5
```

Obrázok 23
Výpis po spustení skriptu s príkazom cyklu

Obrázok 3: Ukážka riešeného príkladu z učebnice „Počítačové systémy a siete“

Časť učebnice, ktorá sa orientuje na oblasť operačných systémov, obsahuje úvod do operačných systémov, riadenie procesov, riadenie pamätí, riadenie súborov v operačnom systéme.

V úvode do operačných systémov sú uvedené hlavné úlohy a funkcie operačného systému, základy práce s operačným systémom Linux a príklady systémových programov v skriptovacom jazyku Bash. V tejto kapitole je uvedených 16 riešených príkladov. Ukážku riešeného príkladu môžeme vidieť na obrázku 3.

4 VZDELÁVANIE UČITEĽOV INFORMATIKY

V rámci aktivity 1.2 *Vzdelávanie učiteľov informatiky, matematiky, prírodovedných a technických predmetov ZŠ a SŠ* sa riešitelia KI podieľali na aktualizáčnom a inovačnom vzdelávaní učiteľov ZŠ a SŠ v rámci rôznych aktivít.

4.1 Aktivita 1.2 – Klub učiteľov informatiky

Od januára 2020 začala KI (Voštinár, P. a Melicherčík, M.) organizovať Klub učiteľov informatiky KI FPV UMB. V rámci aktivity sa počas štyroch hodín preberajú aktuálne novinky vo svete informatiky vo vyučovaní na ZŠ a SŠ. Materiály zo stretnutí majú učitelia kedykoľvek k dispozícii v systéme LMS Moodle¹. Za jeden rok sa uskutočnilo 7 stretnutí Klubu učiteľov informatiky. Pôvodne sa počítalo s viacerými stretnutiami, avšak kvôli pandémie Covid-19 bolo stretnutí iba 7. Prvé štyri stretnutia boli realizované prezenčnou formou, zvyšné tri stretnutia dištančnou formou.

Počas týchto stretnutí sa preberali viaceré témy:

- Programovanie edukačných dosiek micro:bit.
- Programovanie LEGO EV3 v online prostredí MakeCode.
- Rôzne didaktické pomôcky zamerané na motivačné programovanie.
- Výučba programovania mobilných aplikácií v App Inventore.
- Programovanie 2D hier v prostredí MakeCode Arcade.
- Výučba mikrokontrolérov Arduino (2 stretnutia).

4.2 Aktivita 1.2 – Transformácia vzdelávania a školy pre digitálnu dobu

V rámci aktivity 1.2 sa Voštinár, P. zapojil do vzdelávania riaditeľov a IT koordinátorov. V rámci online webinárov od konca novembra 2020 do začiatku februára 2021 sa zrealizovalo 10 stretnutí, kde sa viac ako 100 riaditeľov a IT koordinátorov ZŠ a SŠ zoznámilo s mikrokontrolérom Arduino a edukačnou doskou micro:bit. Stretnutia mali názov témy:

- Rôzne možnosti využitia edukačnej dosky micro:bit a mikrokontroléra Arduino vo vyučovaní (STEM ZŠ).
- Rôzne možnosti využitia mikrokontroléra Arduino vo vyučovaní (STEM).

5 MOTIVÁCIA ŽIAKOV A ŠTUDENTOV PRE ŠTÚDIUM IKT, PRÍRODNÝCH A TECHNICKÝCH VIED

Svet, v ktorom žijeme, sa mení závažnou rýchlosťou vpred smerom k digitalizácii spoločnosti, a preto každá aktivita, zvyšujúca záujem o vzdelávanie v oblasti informatiky, je v tejto digitálnej dobe veľmi prospešná a vítaná. V celom svete chýbajú IT

odborníci. Jedným z cieľov projektu IT akadémie je motivácia žiakov a študentov pre štúdium informatiky a iných prírodovedných a technických vied. V rámci tejto popularizácie sa riešitelia z KI snažia o motiváciu k štúdiu informatiky v rámci organizovania rôznych aktivít, ako sú krúžok programovania pre žiakov ZŠ, workshopy pre žiakov, online IT čajovňa, denný IT tábor, či Girl's Day.

5.1 Krúžok programovania pre žiakov ZŠ

Od akademického roku 2018/2019 organizuje krúžok programovania pre žiakov ZŠ Voštinár, P. V rámci krúžku, ktorý prebieha dvakrát do týždňa po 1,5h sa žiaci z rôznych základných škôl zoznávajú s rôznymi didaktickými pomôckami zameranými hlavne na programovanie – edukačné dosky micro:bit, roboti Edison, mBot, Sphero, Lego EV3, Pro-Bot, Phiro Pro, CodeyBot, dosková hra Scottie Go!, Ozobot EVO, Kano Harry Potter. V rámci troch rokov si žiaci mali možnosť vyskúšať aj programovanie mobilných aplikácií v App Inventore, modelovanie 3D objektov a tvorbu jednoduchých webových stránok. Na obrázku 4 sú zobrazení žiaci pri programovaní mobilných aplikácií na krúžku.



Obrázok 4: Programovanie mobilných aplikácií na ovládanie robotov mBot

Počas dištančnej výučby (uzatvorenia škôl kvôli pandémie Covid-19) krúžok pokračuje on-line s aktivitami programovanie 2D hier v prostredí MakeCode Arcade (Obrázok 5) a programovaním vo svete Minecraft: Education Edition.



Obrázok 5: Programovanie vo svete Minecraft

¹ <https://lms.umb.sk/course/view.php?id=2772#section-1>

5.2 Workshopy pre žiakov ZŠ a SŠ

KI FPV UMB začala realizovať od akademického roka 2019/2020 bezplatné workshopy pre žiakov ZŠ a SŠ v rámci Banskobystrického kraja. Voštinár, P., Melicherčík, M., a Vagač, M. realizovali viacero prezenčných workshopov na školách v Žiari nad Hronom, Zvolene, Novej Bani, Brezne a Banskej Bystrici. Školy zapojené do projektu IT akadémia si majú možnosť vybrať z viacerých tém, ktoré realizujeme na školách, alebo ak žiaci chcú, môžu prísť k nám na katedru. Témy sú dostupné na stránke katedry².

Na obrázku 6 je zobrazený workshop v Žiari nad Hronom, v ktorom mali možnosť si žiaci vyskúšať programovanie robotov Phiro Pro a Scottie Go!



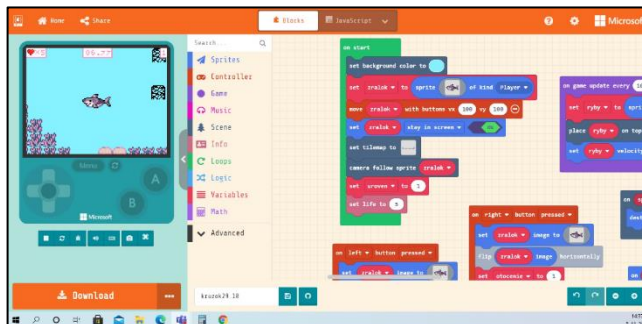
Obrázok 6: Workshop v Žiari nad Hronom

Počas dištančnej výučby workshopy pokračovali online s hlavnou témou programovanie 2D hier v prostredí MakeCode Arcade. Viaceré dištančné workshopy robil Voštinár, P. pre osemročné gymnázia hlavne v Brezne a Žiari nad Hronom.

5.3 IT čajovňa

Koncom roka 2020 dostali riešitelia projektu IT akadémia možnosť urobiť online workshopy/webináre v rámci aktivity IT čajovňa. Jedného stretnutia sa môže zúčastniť 100 žiakov. Voštinár, P. zrealizoval v rámci mesiaca január 2021 online workshopy pre žiakov ZŠ (2x) a pre žiakov SŠ (1x) na tému *Programovanie 2D hier v prostredí MakeCode Arcade*. Na obrázku 7 je zobrazená ukážka hry z prostredia MakeCode Arcade.

V rámci dvojhodinového workshopu si žiaci mali možnosť vyskúšať vytvoriť jednoduché 2D hry v online prostredí MakeCode Arcade, ktoré si mohli vyskúšať zahrať v hernom simulátore, alebo v mobilnom tablete/telefóne. V aktivitách IT čajovne plánuje KI pokračovať aj v ďalších mesiacoch s rôznymi témami.



Obrázok 7: IT čajovňa - programovanie 2D hier

5.4 IT tábor

Ďalšou plánovanou a tradične každoročne realizovanou aktivitou je denný, letný IT tábor, ktorého sa zúčastňuje 40 detí rôznych vekových kategórií. Posledný IT tábor, ktorý bol zorganizovaný ešte pred pandemiou, sa venoval robotike so zostavami LEGO Mindstorms a programovaniu v prostredí Codu Game Lab.

5.5 Siedmy ročník Girl's Day

Šesť predchádzajúcich ročníkov Girl's Day sa konalo vždy štvrtý štvrtok v štvrtom mesiaci a vlani sa ho zúčastnilo vyše 1200 dievčat zo 132 škôl. Siedmy ročník tohto podujatia sa uskutočnil vzhľadom na aktuálnu situáciu s COVID-19 v náhradnom termíne 8.10. 2020. V tomto ročníku takmer 40 firiem a inštitúcií v piatich slovenských mestách ponúko svoje priestory (niektoré v online prostredí), aby prezentovali svoje aktivity v oblasti IT sektora a pomohlo tak prilákať nežné pohlavie ku vzdelávaniu informatiky. Organizátorom akcie bolo vždy občianske združenie Aj Ty v IT, tento rok s podporou Ministerstva investícií, regionálneho rozvoja a informatizácie Slovenskej republiky a pod záštitou UNESCO (<https://girlsday.sk/>).

Naša katedra informatiky FPV UMB si pripravila pre dievčatá zaujímavý program, v rámci ktorého sa mohli zoznámiť v skratke so štúdiom učiteľstva informatiky a aplikovanej informatiky. Predstavili sme im najnovšie didaktické pomôcky na vyučovanie informatiky a mohli si vyskúšať viacero robotíkov a robotických hračiek naživo. Na obrázku 8 a 9 je zobrazená aktivita na programovanie robotov Ozobot Evo v rámci akcie Girl's Day.



Obrázok 8: Programovanie Ozobotov

² <https://www.fpv.umb.sk/katedry/katedra-informatiky/workshopy-pre-zs-a-ss-6957.html>



Obrázok 9: V laboratóriu VR a používateľského zážitku

Ukázali sme im naše laboratória: laboratórium robotov a didaktických pomôcok, geoinformatiky a digitálnych technológií, historickú expozíciu počítačov a číslicovej techniky, laboratórium virtuálnej reality a používateľského zážitku a priblížili sme im aplikácie, ktoré vytvorili študenti v rámci výučby počítačovej grafiky, počítačového videnia, rozpoznávania obrazu, programovania, robotiky, virtuálnej reality a umelej inteligencie. Dievčatá najviac ocenili, že si mohli naprogramovať malých Ozobotíkov a zažili na vlastnej koži vnorenie sa do rôznych prostredí virtuálnej reality.

ZÁVER

Výstupy z projektu sú k dispozícii pre vyše 350 ZŠ a SŠ, 33 000 žiakov základných a stredných škôl, 2 100 pedagogických a odborných zamestnancov, ktorí sú zapojení do projektu *IT*

Akadémia – vzdelávanie pre 21. storočie. Veríme, že spomenuté inovatívne učebné metodiky, učebnice a aktivity, môžu byť užitočné a atraktívne aj pre ďalších žiakov a študentov modernej školy na Slovensku a poslúžia ako inšpirácia aj pre širšiu komunitu učiteľov, ktorí premýšľajú o tom, ako modernizovať svoje pedagogické postupy. Takýmto spôsobom sa nám snáď podari prebudovať vzdelávací systém smerom k dôslednej a zmysluplnej implementácii IKT nástrojov do vzdelávania.

POĎAKOVANIE

Tento článok bol vytvorený v rámci národného projektu IT Akadémia – vzdelávanie pre 21. storočie, ktorý sa realizuje vďaka podpore z Európskeho sociálneho fondu a Európskeho fondu regionálneho rozvoja v rámci Operačného programu Ľudské zdroje.

BIBLIOGRAFICKÉ ODKAZY

- [1] KALAŠ, I. Učebné aktivity žiakov pre 21. storočie. [ed.] *DidInfo 2012* [CD]. Banská Bystrica : Univerzita Mateja Bela, Fakulta prírodných vied, 2012. s. 35-46. ISBN: 978-80-557-0342-8.
- [2] IT Akadémia. IT Akadémia - vzdelávanie pre 21. storočie. [Online] [Dátum: 31. 01 2021.] <http://itakademia.sk/zakladne-informacie>.
- [3] ADAMOVÁ Z., ŠKREKO A. *Autorské právo v školách*. Bratislava : UIPS, 2012.

Programovanie robotov vo virtuálnom prostredí

Robot programming in a virtual environment

PaedDr. Martin Zboran

Univerzita Komenského v Bratislave
FMFI KDMFI, Mlynská dolina
942 48 Bratislava
Slovenská republika
mazboran@gmail.com

ABSTRACT

The use of robotic kits in teaching creates real preconditions for students to acquire new knowledge in a playful way and acquire the necessary technical and construction skills. An insufficient number of robotic kits available to schools may be an obstacle to their widespread introduction into the teaching of computer science at primary school. A possible way to solve this problem is to use a suitable virtual environment to simulate robots.

The article deals with the use of simulation environments in teaching the construction and programming of robots, their characteristics and setting the criteria for their selection for use in the second stage of primary school. It also presents some findings on the issue based on information in the available literature.

Keywords

Simulation. Virtual environment. Robot programming. Educational robotics.

ABSTRAKT

Používanie robotických stavebníc vo vyučovaní vytvára reálne predpoklady, aby si žiaci hrovou formou osvojovali nové poznatky a získali potrebné technické a konštrukčné zručnosti. Prekážkou ich celoplošného zavedenia do vyučovania informatiky na základnej škole môže byť nedostatočný počet robotických stavebníc, ktorými školy disponujú. Možným spôsobom riešenia tohto problému je použitie vhodného virtuálneho prostredia na simuláciu robotov.

Článok sa zaoberá s využitím simulačných prostredí vo vyučovaní konštruovania a programovania robotov, ich charakteristikou a stavením kritérií ich výberu na použitie na druhom stupni základnej školy. Uvádza tiež niektoré zistenia o danej problematike na základe informácií v dostupnej literatúre.

Kľúčové slová

Simulácia. Virtuálne prostredie. Programovanie robotov. Edukačná robotika.

1 ÚVOD

S rozvojom nových technológií je potrebné pripraviť žiakov na ich implementáciu v praxi. Hrovou formou tejto prípravy môže byť programovanie robotov. Takáto hra so stanovenými pravidlami má svoje opodstatnenie aj vo vyučovaní. Učiteľ vyberá jednotlivé aktivity tak, aby napomáhali plniť vzdelávacie ciele. Pritom môže vyučovacie proces realizovať hrovou formou, ktorá je pre žiakov zaujímavejšia, poskytuje im príjemný zážitok a získavajú nové vedomosti a skúsenosti spontánne bez toho, aby si to uvedomovali.

2 CHARAKTERISTIKA A KRITÉRIÁ VÝBERU SIMULAČNÝCH PROSTREDÍ NA PROGRAMOVANIE ROBOTOV

2.1 Hra a jej dôležitosť v živote dieťaťa

Hra je v živote detí veľmi dôležitou funkciou. Pomáha vyplniť voľný čas, ale môže aj pomôcť získať nové zručnosti, rozvíjať tvorivosť, psychomotorické schopnosti, zlepšiť zdravotný stav, zahnať stres, získať nové skúsenosti [1].

Najlepšou alternatívou je, ak hra nielen zabáva, ale aj vzdeláva. Dieťa prostredníctvom hry objavuje, experimentuje, hľadá rôzne riešenia reálnych problémov v bezpečnom prostredí hry. Vhodnou formou takýchto hier môžu byť aj edukačná robotika, digitálne hry, napríklad hra s programovateľnými robotmi, ktorá okrem zábavy a rozptýlenia podporuje aj logické myslenie, rozvoj technických zručností a získavanie nových vedomostí pomocou gamifikácie aktivít [2].

2.2 Simulácia

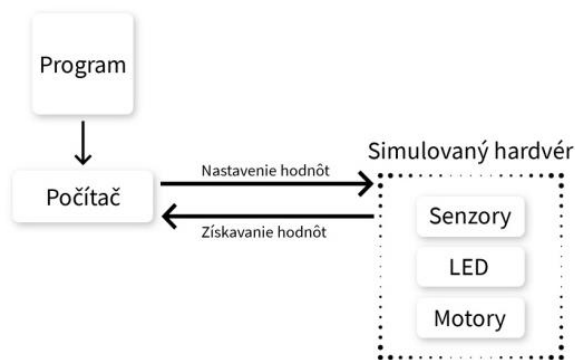
Niekedy nie je k dispozícii reálny robot. V takom prípade ho môže nahradiť jeho simulácia vo virtuálnom prostredí, ktoré umožňuje programovať roboty bez potreby zostavenia reálneho robota. Je vhodnou alternatívou pre školy, ktoré z finančných dôvodov nemôžu zakúpiť dostatočné množstvo robotických stavebníc pre všetkých žiakov.

Simulácia je experimentálna metóda, v ktorej sa imitujú reálne stavy a procesy pomocou počítačových modelov implementovaných na základe exaktného matematického modelu. Tento model umožňuje realizovať množstvo simulačných experimentov, vyhodnotiť ich a na základe výsledkov vybrať optimálny model (model s optimálnymi parametrami) a potom ho reálne použiť [3].

V našom prípade po napísaní programu počítač komunikuje so simulačným hardvérom vo vybranom simulačnom prostredí, ktorý zadané parametre realizuje a posiela späť dáta na vyhodnotenie správnosti operácie (pozri Obrázok 1).

Na trhu je v ponuke široká paleta simulačných prostredí robotov. Niektorí autori uprednostňujú prostredia, v ktorých nie je hlavným účelom napodobňovať reálne mechanizmy, ale hľadať konštrukčné riešenia, iní simulácie totožné s konkrétnymi mechanizmami [4].

Simulačné prostriedky a prostredia napomáhajú žiakom pochopiť ako pracuje robot, ktorý simulačný softvér ponúka formou už pripraveného funkčného modelu alebo dáva možnosť zostaviť vlastný model. V našom prípade simulácia, bola iba prostriedkom na naplnenie cieľov vyučovacieho procesu.



Obrázok 1: Interakcia simulátora a programu [5]

3 KRITÉRIÁ VÝBERU SIMULAČNÉHO PROSTREDIA PRE ŽIAKOV ZÁKLADNÝCH ŠKÔL

3.1 Kritériá výberu podľa niektorých autorov

Pri výbere simulačných prostriedkov a prostredí odborníci nemajú jednotný názor. Uvedieme aké kritériá stanovili traja vybraní odborníci na výber simulačného prostriedku pre žiakov základných škôl.

Tselegradis [6] vybral nasledovné kritériá:

- typ robota (dron, robotické rameno, dynamický s kolesami, multifunkčný),
- vek používateľov (najčastejšie od 6 do 18 rokov, prípadne aj nad 18 rokov),
- programátorské poznatky a úroveň používateľov (je prepojená s vekom aj programovacími skúsenosťami užívateľa),
- rok vývoja prostredia (môže mať vplyv na vizuálnu stránku),
- programovací jazyk, ktorý používa (Python, MATLAB a iné),
- účel použitia (na vzdelávanie, na experimentovanie, na súťaže),
- či umožňuje simuláciu reálneho robota,
- aké vývojové prostredie možno použiť (Unity3D, Gazebo, V-REP),
- operačný systém (Windows, Linux)

Autor skonštatoval, že pre veľmi široký záber a rôzny vek žiakov je ich porovnanie ťažko realizovateľné.

Cañas [7] hodnotil simulačné prostredia a emulátory (softvér umožňujúci priebeh počítačových programov na inej platforme, než pre ktorú boli pôvodne vytvorené a ktorú samy od seba podporujú, je špeciálnym prípadom virtualizácie).

Jeho kritériá okrem predchádzajúcich zahŕňajú aj:

- cenu (platené alebo zadarmo),
- platforma (spustenie s inštaláciou alebo spustenie iba cez webový prehliadač),
- emulátor (jeho prítomnosť alebo neprítomnosť),
- virtuálne prostredie (2D, 3D, nie),
- realizácia (plnenie predefinovaných úloh pre postup do vyššej úrovne alebo nepredefinované úlohy).

Jedným z kritérií Molnára [8,9] bolo, či má virtuálne prostredie zakúpenú licenciu od firmy LEGO na používanie dizajnu robota LEGO Mindstorms alebo či je simulačné prostredie voľne dostupné bez zakúpenia licencie.

Haak [10] medzi kritériá zaradil aj:

- čas potrebný na sfunkčnenie programu,
- úplnosť simulačného prostredia,
- existencia príručiek a dokumentácie,
- podporu zo strany komunity používateľov a výrobcu,
- jednoduchosť používania prostredia.

3.2 Naše kritériá výberu simulačných nástrojov pre základné školy

Autori, ktorí sa venujú problematike simulácii robotických stavebníc, prispôbujú výber kritérií svojim potrebám. Naš záujem je nájsť kritériá vhodné pre druhý stupeň základných škôl, aby boli pre žiakov nie iba hrou, ale aj prínosom k vzdelávaniu a alternatívou pre reálne robotické stavebnice. Naším cieľom je nastaviť vhodné kritériá pre výber simulačného prostredia.

V nadväznosti na kritériá predchádzajúcich autorov a vzhľadom na naše požiadavky sa nám zdajú byť vhodné brať do úvahy tieto:

- dosiahnutie edukačných cieľov – naplnenie štátneho vzdelávacieho programu,
- stupeň znalostí programovania, ktoré prostredie vyžaduje - predošlé skúsenosti žiaka s programovaním – vhodnosť pre základnú školu,
- finančné náklady – toto kritérium môže byť veľmi dôležité, pretože vzhľadom na finančnú situáciu, môže byť pre základné školy vhodnejšou alternatívou zakúpenie simulačného prostredia namiesto reálnej robotickej stavebnice,
- platforma – potreba nainštalovania a možnosť použitia len na jednom zariadení alebo možnosť spustenia na hociktorom zariadení cez webový prehliadač,
- programovací jazyk – grafický alebo textový, prípadne sú možné oba,
- virtuálne prostredie – 2D, 3D,
- aktualizácie – spätná väzba prípadne aktívna komunita,
- návody pre žiaka (textové príručky, video-návody a pod.)
- rozpracovaná metodika pre učiteľa,
- možnosť pripojenia aj reálneho robota.

Za dôležitú podmienku považujeme predošlé skúsenosti žiaka s programovaním. Preto je potrebné určiť, pre aký stupeň znalostí programovania je určený daný prostriedok, aby programovací jazyk a ovládanie simulačného prostredia nebolo pre žiaka náročné. Z hľadiska jednotlivých základných škôl môže byť vzhľadom na ich ekonomickú situáciu cena jedným z hlavných kritérií. Výsledky porovnania cien simulačného prostredia s cenami reálnych stavebníc z dôvodov efektívneho využívania finančných prostriedkov môže byť rozhodujúcim faktorom pre rozhodnutie.

V dobe dištančného vzdelávania je prednosťou jednoduchšia implementácia, ktorú ponúka webový prehliadač a zároveň pri prezenčnom vyučovaní má žiak možnosť pracovať na probléme nielen v škole, ale aj v domácom prostredí.

Vhodnejšou alternatívou pre žiakov druhého stupňa základnej školy je na programovanie použiť grafický jazyk, lebo je pre nich jednoduchší. Výhodou je možnosť prepnúť na textový jazyk pre žiakov, ktorí dosiahli pokročilejšiu úroveň alebo budú pokračovať na stredných školách.

Vizualizácia prostredia plošná alebo priestorová je tiež jeden z faktorov výberu. Priestorová je reálnejšia, môže byť však náročnejšia na hardvér počítača, ale poskytuje dokonalejšiu grafiku, ktorá je pre žiaka zaujímavejšia [11].

Aktualizácia poskytuje funkčné vylepšenia a aktívna komunita môže byť pre žiaka aj učiteľa inšpiráciou a pomocou pri riešení rôznych problémov.

Textové návody, prípadné video-návody v rôznych internetových službách sú veľmi dobrým doplnkom pre žiaka.

Pre učiteľa, ktorý nemá skúsenosti so simulačným prostredím je dobrou pomôckou aj vypracovaná metodika.

Pripojenie reálneho robota je vhodnou kombináciou práce vo virtuálnom a reálnom prostredí. Kritéria výberu prostredia sú zhrnuté v tabuľke 1.

Tabuľka 1: Kritériá výberu simulačného prostredia

KRITÉRIÁ
Dosiahnutie edukačných cieľov
Stupeň znalostí programovania
Finančné náklady
Platforma
Programovací jazyk
Virtuálne prostredie (2D / 3D)
Aktualizácie
Návody pre žiaka
Metodika pre učiteľa
Možnosť pripojiť reálny robot

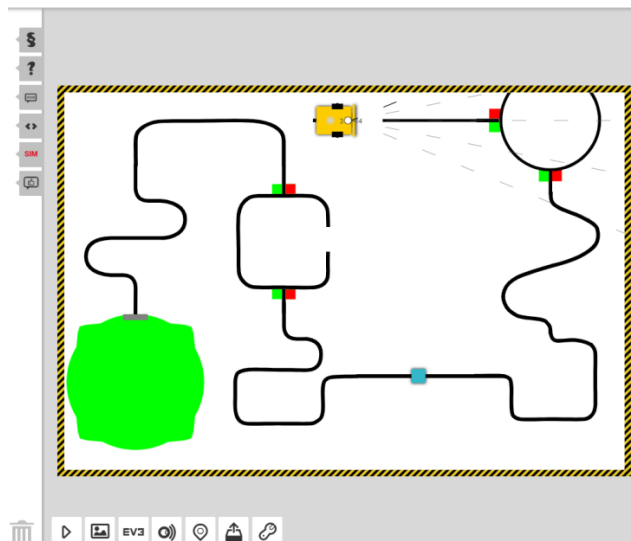
Predpokladáme, že zohľadnenie týchto kritérií pomôže vybrať virtuálne prostredie pre druhý stupeň základnej školy a vhodne ho implementovať do vyučovacieho procesu. Jedným z najdôležitejších kritérií je splnenie stanovených vzdelávacích štandardov. Prostredie by malo byť flexibilné, aby s ním mohli pracovať žiaci s rôznym stupňom znalosti programovania. S tým súvisí aj výber programovacieho jazyka. Pre začiatočníkov je vhodnejší grafický jazyk. Pokročilejší môžu pracovať aj s textovým jazykom. Dôležité je zhodnotiť aj cenovú dostupnosť pre jednotlivé školy. 3D prostredie názornejšie predstavuje reálnu skutočnosť, čím sa stáva pre žiaka zaujímavejším. Aktualizácie, vypracované návody pre žiaka a metodika pre učiteľa tiež môže zvýšiť záujem učiteľov o využívanie takýchto prostredí vo vyučovaní.

3.3 Vybrané simulačné prostredia

Ponuka simulačných prostredí je bohatá. Predstavíme z nich tri. Vybrali sme prostredia, v ktorých simulácie prebiehajú v 2D a 3D v priestore. Dve z nich sú dostupné zadarmo, a jedno je platené. Pre porovnanie sú na ukázkových fotografických záberoch vybrané prostredia zamerané najmä na pohyb robota po čiare.

A) Open Roberta Sim

Toto simulačné prostredie je zadarmo, vhodné pre žiakov druhého stupňa základných škôl. Implementácia pre používateľa je jednoduchá a ľahko použiteľná, možno ho spustiť bez inštalácie cez webovú aplikáciu spustenú v prehliadači. Nie je dostupná v slovenskom jazyku, ale z ponuky rôznych jazykov je možnosť vybrať si český jazyk. Využíva grafický programovací jazyk založený na programovacích blokoch Blockly s názvom NEPO®. Programovať sa dá iba v grafickom jazyku, textový sa dá zobrazíť bez možnosti úpravy. Jeho súčasťou sú senzory (dotykový, svetelný, ultrazvukový, optický). Virtualizácia prebieha v 2D prostredí, na výber ich je niekoľko. Je možnosť vložiť aj vlastné prostredie. Open Roberta Sim obsahuje aj emulátor kocky EV3. Použiteľný je aj pre rôzne druhy reálnych robotov (EV3, micro:bit, Edison a ďalšie) [12] (pozri Obrázok 2).



Obrázok 2: Prostredie Open Roberta Sim [12]

B) GearsBot

GearsBot má prepracovanejšie simulačné 3D prostredie. Je k dispozícii zadarmo. Je spustiteľná vo webovom prehliadači, nie je potrebné prihlasovanie. Slovenské jazykové prostredie nie je k dispozícii. Na komunikáciu možno použiť anglický, prípadne nemecký jazyk.

Programovanie je založené na grafickom jazyku a vytvorený program sa automaticky konvertuje na Python alebo sa priamo programuje cez programovací jazyk Python. Simulácia je realistejšia ako v predchádzajúcom prostredí. Simulovaný robot sa správa ako reálny. To znamená napríklad, že sa nepohybujú „ideálne“ rovno, lebo preklzávajú pneumatiky. Je určený pre pokročilejších žiakov.



Obrázok 3: Prostredie GearsBot [13]

Umožňuje vložiť vlastné prostredie alebo využiť preddefinované a plniť úlohy zamerané na sledovanie čiar, vyhýbanie sa rôznym prekážkam. V poslednom čase prešiel veľkou inováciou. V tomto prostredí sa dá pracovať s optickým, ultrazvukovým, laserovým, gyroskopickým a dotykovým senzorom a tiež GPS. Nedá sa použiť pre reálny robot [13] (pozri Obrázok 3).

C) Virtual Robotics Toolkit

Simuluje reálny fyzický robot LEGO Mindstorms EV3. Je vhodný pre druhý stupeň základnej školy. Je potrebné si ho zakúpiť a nainštalovať. Jedna zakúpená licencia je pre jeden počítač. V porovnaní s reálnou robotickou stavebnicou je cenovo

výhodnejší. Zatiaľ je k dispozícii v anglickom, prípadne nemeckom jazyku.

Pracuje v 3D prostredí, ktoré je veľmi dobre prepracované. Robot je preddefinovaný alebo sa dá v programoch LEGO Digital Designer a LDraw vytvoriť vlastný model. Obsahuje emulátor kocky EV3. Na výber sú dve programovacie prostredia.

- LEGO MINDSTORMS Education EV3 Software, ktorý sa bežne používa. A na programovanie je k dispozícii jazyk EV3-G. Je to grafický jazyk, kde sa programuje metódou drag and drop. Pri tomto programovacom prostredí je potrebná inštalácia.
- Microsoft MakeCode dostupné online a je možné programovanie pomocou blokov alebo aj v textovom programovacom jazyku JavaScript [14] (pozri Obrázok 4).



Obrázok 4: Prostredie Virtual Robotics Toolkit [14]

4 REÁLNE A VIRTUÁLNE ROBOTICKÉ STAVEBNICE VO VZELÁVANÍ

S reálnymi robotickými stavebnicami sa môžeme stretnúť na viacerých školách na hodinách informatiky, iných prírodovedných predmetoch alebo na krúžkovej činnosti. Problémom je ich premiestňovanie do odborných učební, následné prestavovanie na požadovaný účel [15,16]. Viacerí autori sa venovali týmto problémom a porovnávaním práce žiakov s fyzickým robotom a simulátorom pozitívnejšie hodnotia prácu vo virtuálnom prostredí.

Využitie virtuálnych robotov má aj určité obmedzenie, pretože sa nepracuje s hmatateľným skutočným robotom. Ale ak je simulačné prostredie správne navrhnuté a programovací jazyk je ľahko použiteľný, pre žiaka začínajúceho s programovaním môže byť ich využitie či už na konštruovanie alebo riadenie robotov menej náročné [17].

Využívanie grafických programovacích jazykov a virtuálnych robotických prostredí na vzdelávanie v oblasti programovania môže uľahčiť získavanie potrebných zručností v informatickom, no hlavne algoritmickej a programátorskej mysli, ktoré sú mimoriadne dôležité v 21. storočí [18].

Zurba [19] porovnával skupiny žiakov, ktorí pracovali len s fyzickým robotom, ďalšia iba so simulátorom a tretia aj s fyzickým robotom aj virtuálnym prostredím. Z výsledkov pozorovania vyplýva, že zmiešaná skupina úlohy riešili tvorivejšie a mali lepšiu interakciu so spolužiakmi.

Liu [20] porovnávaním skupín pracujúcich s virtuálnou alebo reálnou stavebnicou zistil, že obidve skupiny dosiahli rovnaké výsledky, ale skupina so simulátorom ich dosahovala v kratšom čase, nakoľko ju nezdržovalo prestavovanie robotov.

Podľa Gonzalesa-Garcia [21] ak nebudú mať prístup k reálnemu robotovi, môžu si žiaci natrénovať úlohy vo virtuálnom prostredí a zistil, že po prechode na reálne úlohy sa ich vedomosti nezhoršili. Programovanie vo virtuálnom prostredí je vhodnou alternatívou vo vyučovaní, lebo podporuje aj logické myslenie, tvorivosť, kreativitu. Pomocou 3D modelov, ktoré sú identické s časťami robotickej stavebnice LEGO, sa dajú nasimulovať rôzne úlohy skontrolovať ich funkčnosť skôr, ako zostrojíme reálny robot. Autor poukazuje na to, že keď mám virtuálny model robota alebo robotického ramena z výrobného prostredia, môžem ho využiť na nácvik pracovných činností bez potreby prerušovania pracovného procesu [22].

Práca s fyzickými modelmi robotov je motivujúca, ale žiaci ich môžu používať iba keď sú v škole a tak nemajú možnosť precvičovať si úlohy a hľadať rôzne riešenia doma. Tu sa ponúka ako vhodné riešenie využívanie virtuálneho prostredia [10].

Ak simulačné prostredie dokáže čo najpresvedčivejšie prezentovať skutočné prostredie, môže lepšie prepojiť fázu testovania so skutočným používaním [22].

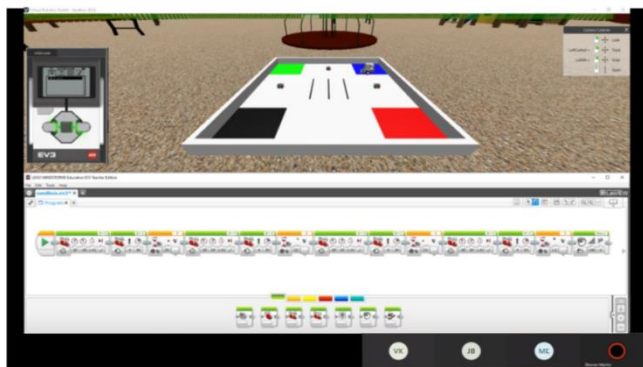
5 NAŠE SKÚSENOSTI A ODPORÚČANIA

Počas dištančnej formy vyučovania bolo programovanie robotov vo virtuálnom prostredí adekvátnou alternatívou programovania reálneho robota.

Na základe predchádzajúcich skúseností sme vybrali Virtual Robotics Toolkit. Zakúpená bola iba jedna licencia a nainštalovaná na jednom počítači. Komunikácia so žiakmi a zdieľanie obrazovky prebiehalo pomocou MS Teams. Žiaci už mali skúsenosti s programovým prostredím LEGO MINDSTORMS Education EV3 Software a jeho grafickým jazykom EV3-G, ktorý mali nainštalovaný vo vlastných počítačoch. Prostredníctvom zdieľania pracovnej plochy učiteľa žiaci mohli upravovať program. Po predvedení vzorovej úlohy učiteľom, mali žiaci priestor na riešenie zadanej úlohy. Nakoľko však nemali k dispozícii virtuálne prostredie, nemohli overovať funkčnosť svojho kódu priamo. Overovanie sa realizovalo postupne cez zdieľanú obrazovku. Počas overovania prebiehala diskusia o správnosti programu ako aj o rôznych alternatívach riešenia (pozri Obrázok 5).

Toto virtuálne prostredie sme si vybrali na základe našich predchádzajúcich skúseností a analýzou pozorovania niektorých autorov. V súčasnej dobe je oveľa viac možností výberu, čo si vyžaduje hlbšiu analýzu týchto virtuálnych prostredí, preto sme si stanovili kritériá uvedené v časti 3.2.

Z nich je dôležitou podmienkou, aby ich mali k dispozícii všetci žiaci vo svojom počítači. Virtuálne prostredie autor využíva vo vyučovacom procese, ako aj v krúžkovej činnosti. V čase dištančného vyučovania, keď nie je možné pracovať s reálnymi



Obrázok 5: Práca s virtuálnym prostredím

robotickými stavebnicami, je vhodnou alternatívou na dosahovanie vyučovacích cieľov. Stanovením kritérií sme chceli prispieť k správne mu výberu takéhoto prostredia.

Autor využíval vo vyučovaní na komunikáciu so žiakmi MS Teams. Bolo by dobré preskúmať aj iné online komunikačné platformy.

Bolo by vhodné porovnať použitie reálnej a virtuálnej stavebnice a overiť, ktorý typ stavebnice je výhodnejší na prezenčné vyučovanie.

6 ZÁVER

Skonstruovanie a programovanie robotov je pre žiakov základných škôl obľúbenou činnosťou. Pripomína im to herné aktivity s klasickými mechanickými ako aj modernejšími a „inteligentnejšími“ stavebnicami – vracia ich to do bezstarostného detstva. Úspechy pri zostavení a sfunkčnení robota môže žiakov motivovať do ďalšej činnosti, k zdravému porovnávaniu sa navzájom, ale aj k vzájomnej komunikácii a riešeniu spoločných projektov. Môžu samostatne hľadať a nachádzať riešenia a akceptovať aj riešenia ostatných spolužiakov. Hravou formou možno takto naučiť žiakov základy programovania, hľadať nové riešenia úloh a odvahu prezentovať svoju prácu. Prostriedkom k dosiahnutiu takéhoto cieľa môže byť aj práca v simulačných prostrediach. Simulácia je v súčasnej dobe neodmysliteľná nielen pre tých, ktorí začínajú programovať, ale aj v priemyselnej robotike. Tomuto vývoju je preto potrebné prispôbiť aj vzdelávanie.

POĎAKOVANIE

Príspevok vznikol za podpory KEGA 012TTU-4/2018: Interaktívne animačno-simulačné modely vo vzdelávaní.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] STOFFOVÁ, V. *Počítačové hry a ich klasifikácia*. In: *Trendy ve vzdělávání*. Roč. 9, 2016a, č. 1, s. 243-252. ISSN 1805-8949 *Trends in Education Information Technologies and Technical Education*. Ročník 9, 2016/1
- [2] STOFFOVÁ, V. *The Importance of Didactic Computer Games in the Acquisition of New Knowledge*. In: *The European Proceedings of Social & Behavioural Sciences EpSBS*. 2016b. pp. 676-688. eISSN 2357-1330 [HTTP://DX.DOI.ORG/10.15405/EPBSBS.2016.11.70](http://dx.doi.org/10.15405/epsbs.2016.11.70)

- [3] STOFFOVÁ, V. *Počítač – univerzálny didaktický prostriedok* 1. vyd. Nitra : Fakulta prírodných vied UKF v Nitre, 2004. 172 s. ISBN 80-8050-450-4
- [4] JAKUBIEC, B. *Application of simulation models for programming of robots*. Society integration education Proceedings of the International Scientific Conference. Volume V. 2018. s. 283-292. <https://pdfs.semanticscholar.org/7674/852073141c31aac8120cba23ae8af484e0dd.pdf>
- [5] CORNELISSEN, L. *Simulating Lego Mindstorms EV3 robots using Unity and Python* Author: Leo Cornelissen – s4606566. 2019. 33 s. https://www.cs.ru.nl/bachelors-theses/2018/Leo_Cornelissen__4606566__Simulating_Lego_Mindstorms_EV3_robots_using_Unity_and_Python.pdf
- [6] TSELEGRADIS, S. *Simulators in Educational Robotics: A Review*. 2021. 12 s. <https://www.mdpi.com/2227-7102/11/1/11>
- [7] CAÑAS, J. *A ROS-Based Open Tool for Intelligent Robotics Education*. 2021. 21 s. <https://www.mdpi.com/2076-3417/10/21/7419/pdf>
- [8] MOLNÁR, G. *The virtual toolkit of digital instruction and its application in digital work forms*. 11th IEEE International Conference on Cognitive Infocommunications – CogInfoCom 2020 . 4 s. https://www.researchgate.net/publication/344388850_The_virtual_toolkit_of_digital_instruction_and_its_application_in_digital_work_forms
- [9] HAAK, V. a kol. *Conception of a Lego Mindstorms EV3 Simulation for teaching C in Computer Science Courses*. 2018 IEEE. 2018. Global Engineering Education Conference. 2018. 6 s. <https://www.semanticscholar.org/paper/Conception-of-a-Lego-Mindstorms-EV3-simulation-for-Haak-Abke/4cb61a82b8bb2ad4fc933ee369e944900194a519>
- [10] HAAK, V. a kol. *Work-in-Progress: Development of a LEGO Mindstorms EV3 Simulation for Programming in C*. Engineering, Computer Science 2018 IEEE Global Engineering Education Conference (EDUCON) . 2018. 8 s. <https://www.semanticscholar.org/paper/Conception-of-a-Lego-Mindstorms-EV3-simulation-for-Haak-Abke/4cb61a82b8bb2ad4fc933ee369e944900194a519>
- [11] HASTINGS, E. a kol. *Young Children's Video/Computer Game Use: Relations with School Performance and Behavior*. *Issues Ment Health Nurs*. 2009 Oct; 30(10): 2009. s. 638–649. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3041171/>
- [12] Open Roberta®. 2021. <https://lab.open-roberta.org/>
- [13] GearsBot. 2021. <https://gears.aposteriori.com.sg/>
- [14] Virtual Robotics Toolkit. 2021. <https://www.virtualroboticstoolkit.com/>
- [15] STOFFOVÁ, V., HAVELKA, M. *Práca s robotickými stavebnicami na 2. stupni ZŠ - Zbierka riešených úloh*. 1. vyd. Olomouc : Pedagogická fakulta UP v Olomouci, 2018. 66. s.
- [16] HAVELKA, M., STOFFOVÁ, V. *Robotika - Stavba a programování robotů (LEGO Mindstorms NXT a EV3) 1*.

- vyd. Olomouc : Pedagogická fakulta UP v Olomouci, 2017. s. 85.
- [17] JOÃO, A. a kol. *A Virtual Robot to support Programming Learning*. Conference: International Symposium on Robotics, 2018. 4 s.
https://www.researchgate.net/publication/327189954_A_Virtual_Robot_to_support_Programming_Learning
- [18] WITHERSPOON, E. B., a kol. *Developing Computational Thinking through a Virtual Robotics Programming*. ACM Transactions on Computing Education, Vol. 18, No. 1. 2017. 20 s. <https://dl.acm.org/doi/10.1145/3104982>
- [19] ZURBA, M. *Progressive education and robotics: a behavioural evaluation of learning with robots and simulator*. 2019. 89 s.
<https://mspace.lib.umanitoba.ca/handle/1993/34302>
- [20] LIU, A. *Students Learn Programming Faster Through Robotic Simulation*. 2013. 4 s.
<https://www.cmu.edu/roboticsacademy/PDFs/Research/LearnProgrammingFasterThroughSimulation.pdf>
- [21] GONZALES – GARCIAL, S. *Teaching forward kinematics in a robotics course using simulations: transfer to a real-world context using LEGO mindstorms™*. International Journal on Interactive Design and Manufacturing. 2020. 14 s.
https://www.researchgate.net/publication/343299203_Teaching_forward_kinematics_in_a_robotics_course_using_simulations_transfer_to_a_real-world_context_using_LEGO_mindstorms
- [22] CHOI, H., a kol. *On the use of simulation in robotics: Opportunities, challenges, and suggestions for moving forward*. 2019. 9 s.
<https://www.pnas.org/content/118/1/e1907856118>
- [23] GUCWA, K. J. a kol. *RoboSim: a simulation environment for programming virtual robots*. 2018. 11 s.
https://www.researchgate.net/publication/321508065_RoboSim_a_simulation_environment_for_programming_virtual_robots

Krátke příspěvky

Short papers

Ukážky úloh z informatických prechádzok

Examples of Tasks from Informatics Walks

Silvia Haringová

Univerzita Konštantína Filozofa
Tr. A. Hlinku 1
949 74 Nitra
Slovenská republika
silviaharingova@atlas.sk

Nika Klimová

Univerzita Konštantína Filozofa
Tr. A. Hlinku 1
949 74 Nitra
Slovenská republika
nika.klimova@ukf.sk

ABSTRACT

Outdoor teaching is used in various subjects. As part of the Comenius Institute project, we decided to implement the approach in the subject of Informatics. We have created Informatics walks, which we consider to be an innovative way to implement Informatics lessons outdoors. During the walks, students develop critical thinking, work with codes and ciphers, security on the Internet and learn about coordinates while using the game Minecraft. The paper describes examples of Informatics walks, which can be implemented in the Informatics education. The aim of Informatics walks is to change the environment in which students learn new knowledge and develop collaboration.

Keywords

Outdoor teaching. Informatics walks. Hoax. Minecraft.

ABSTRAKT

Outdoorové vyučovanie sa využíva v rôznych predmetoch. V rámci projektu Komenského inštitútu sme sa rozhodli zaoberať implementáciou tohto prístupu v predmete informatika. Vytvorili sme informatické prechádzky, ktoré považujeme za inovatívny spôsob, ako realizovať vyučovanie informatiky v exteriéri. Počas prechádzok si žiaci preveria kritické myslenie, prácu s kódmi a šiframi, venujú sa bezpečnosti na internete a naučia sa o súradniciach pomocou hry Minecraft. Príspevok ponúka ukážky informatických prechádzok, ktoré sa môže implementovať do vyučovania informatiky. Cieľom informatických prechádzok je zmeniť prostredie, v ktorom sa žiaci vzdelávajú a rozvíjajú kolaboráciu.

Kľúčové slová

Outdoorové vyučovanie. Informatické prechádzky. Hoax. Minecraft.

1 OUTDOOROVÉ VYUČOVANIE

Outdoorové vyučovanie je pomerne široký pojem. Napriek prvkom anglicizmu sa ustálil v takejto forme, čoho dôkazom sú rôzne slovenské vysokoškolské učebnice [1]. Takéto vyučovanie môže predstavovať množstvo rôznych aktivít, ktoré majú spoločný atribút – realizáciu vo vonkajšom prostredí. Takéto vyučovanie sa stáva pomerne populárnym, hoci nejde o novinku o outdoorovom vyučovaní sa hovorilo už v roku 1957 [2]. Takémuto prístupu sa venovala aj akademická obec vo Veľkej Británii, Austrálii, Dánsku, Singapure, Škótsku či Kanade [3, 4, 5]. Zároveň sa zistilo, že outdoorové vyučovanie prináša benefity ako kolaboratívnosť, sebavedomie, rozvoj tvorivého myslenia či občiansku angažovanosť [3, 6].

2 INFORMATICKÉ PRECHÁDZKY

Informatické prechádzky sú podobné matematickým prechádzkam. Nápad ich vytvorenia vznikol ako projekt realizovaný v rámci ročného vzdelávacieho programu v Komenského inštitúte. Cieľom projektu je vytvorenie sady informatických prechádzok v podobe pracovných listov, ktoré sa využijú na hodinách informatiky so žiakmi ZŠ a SŠ. Prechádzky vytvorili študenti 5. ročníka učiteľstva informatiky v kombinácii na Univerzite Konštantína Filozofa v Nitre. Študenti kolaboratívnou formou na predmete Projekty v školskej informatike vytvorili 4 prechádzky:

- Hoaxy,
- Kódovanie a šifrovanie,
- Minecraft v reálnom svete,
- Bezpečnosť na internete.

2.1 Hoaxy

Dnes je náročné vyhnúť sa dezinformáciám, a preto je dôležité učiť žiakov rozoznávať medzi pravdivými a falošnými informáciami. Žiaci počas prechádzky rozhodujú o pravdivosti jednotlivých správ a následne prebieha diskusia, kde sa rozhoduje o správnom riešení. Úlohou žiakov je kritickým myslením formou argumentov určiť, či je daná správa pravdivá alebo ide o hoax (Obrázok 1). Táto prechádzka je situovaná do Mestského parku v Nitre, no dá sa prispôsobiť do ľubovoľného prostredia, pretože objekty, na ktoré sa viažu vytvorené správy, ako napr. strom, budova, ulica, parkovisko sa nachádzajú kdekoľvek. Cieľom prechádzky je naučiť žiakov, čo je to hoax a ako overovať správnosť informácií. Učiteľ môže do vytvorenej tabuľky zapisovať ich odpovede.

Farská ulica

Farská ulica sa nachádza nad tajnou podzemnou cestou, ktorá viedla do Nitrianskeho hradu. Avšak táto podzemná cesta bola v roku 2015 celoplošne zabetónovaná. Zo záznamov z obdobia socializmu sme sa dočítali, že sa touto podzemnou cestou pašovali na hrad belgické pralinky. Tieto pralinky obsahovali najviac kakaa v porovnaní s ostatnými, a preto boli luxusom, ktorý obyčajní ľudia nemohli okúsiť.

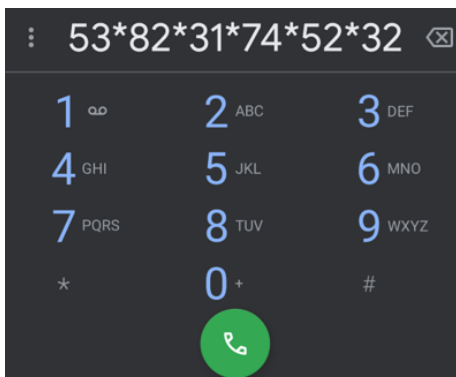


Obrázok 1: Ukážka správ

2.2 Kódovanie a šifrovanie

Žiaci dekodujú alebo dešifrujú úlohy spojené s Morseovou abecedou, piktogramami, ASCII tabuľkou a ďalšími (Obrázok 2). V jednotlivých úlohách sú zakódované alebo zašifrované slová textu. V poslednej úlohe si žiaci môžu pomôcť jednotlivými čiastkovými riešeniami na dešifrovanie celého textu. Po skončení prechádzky si všetci posadajú do kruhu a diskutujú o význame textu, ktorý rozlúštili. Prechádzka je vytvorená formou stanovíšť, aby bola vhodná na realizáciu napr. na školskom dvore. Na každom stanovisku sa nachádza jedna úloha. Cieľom

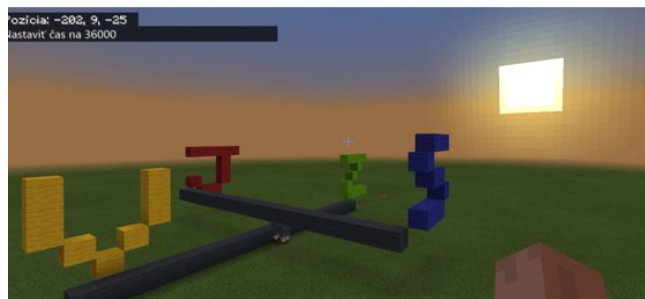
prechádzky je, aby si žiaci osvojili nové vedomosti v oblasti Reprezentácie a nástroje – informácie, rozvíjali svoje komunikačné a praktické zručnosti počas práce v skupinách.



Obrázok 2: Ukážka úlohy na dešifrovanie

2.3 Minecraft v reálnom svete – učíme sa o súradniciach pomocou hry Minecraft

Žiaci sa počas prechádzky učia o pozíciách a ich využití v reálnom svete. Pre 3-4 členné skupiny žiakov sa vytvorila 3 stanovišťa. Úlohy sa nachádzajú na Sway webovej stránke, takže sú potrebné mobily s internetom. Na každom stanovišti dostanú žiaci odkaz v podobe QR kódu. Témy stanovišť sú: kompas, absolútna pozícia, počítanie relatívnej pozície (Obrázok 3).



Obrázok 3: Určovanie vzdialenosti na bloky

2.4 Bezpečnosť na internete

Je opodstatnené venovať sa so žiakmi bezpečnosti na internete, ktorá je zahrnutá aj v ŠVP. Realizácia prechádzky môže prebiehať na ihrisku. Skupiny žiakov sa dozvedia, kde nájdu jednotlivé podtémy, ku ktorým budú vytvárať asociácie a zapisovať ich do myšlienkového mapy. Žiaci budú musieť nájsť podtémy: komunikácia, sociálne siete, správanie na internete a informácie. Cieľom prechádzky je, aby sa žiaci naučili, aké informácie môžu poskytovať na internete, ako komunikovať a zodpovedne sa správať na internete, ale aj o vplyve sociálnych sietí. Po absolvovaní prechádzky každá skupinka odprezentuje svoju myšlienkovú mapu a spoločne s vyučujúcim zreflektujú myšlienkovú mapu. Takáto mapa sa dá spracovať v rôznych editoroch (Obrázok 4).



Obrázok 4: Myšlienková mapa spracovaná v editore

2.5 Realizácia prechádzok

Informatické prechádzky sme mali so spolužiakmi v pláne realizovať vonku so žiakmi základných a stredných škôl. Vzhľadom na pandemickú situáciu sa prechádzky dočasne presunuli do online prostredia. Napriek tomu sa budúcim učiteľom informatiky podarilo zrealizovať informatické prechádzky na 4 školách v 10 triedach s cca 110 žiakmi. Spätná väzba bola od žiakov pozitívna a budúci učitelia ocenili najmä fakt, že žiaci boli na hodine aktívni. Vzhľadom k tomu, že prechádzky boli uskutočnené za online prítomnosti učiteľov z praxe, aj v tomto prípade bola vždy pozitívna spätá väzba. Vyzdvihol sa napr. fakt, že hoci žiaci musia byť doma, pri hoaxoch im evokuje to, že sú vonku, pretože vidia reálne obrázky z Nitry.

3 ZÁVER

Našou snahou bolo vytvoriť informatické úlohy, ktoré sa dajú realizovať vonku. Na základe implementácie úloh na školách počas dištančného vzdelávania s lektormi končiacich študentov učiteľstva informatiky pod vedením učiteľov z praxe mali informatické prechádzky pozitívnu spätú väzbu. Okrem toho aj budúci učitelia postrehli, že žiaci boli aktívni a celkovo sa im vyučovanie páčilo. Vzhľadom k tomu, že projekt informatických prechádzok ešte nie je ukončený, tejto téme sa budeme venovať aj naďalej so zameraním na vyhodnotenie výhod a nevýhod informatických prechádzok na predmete informatika v ZŠ a SŠ.

POĎAKOVANIE

Tento článok bol finančne podporený Komenským inštitútom v rámci projektu Matematické a informatické prechádzky.

BIBLIOGRAFICKÉ ODKAZY

- [1] DURKÁČ, P., CHOVANOVÁ, E. *Outdoorové aktivity v edukácii*. Prešov : Prešovská univerzita, 2013, 185 s.
- [2] SMITH, J. W. *The Good Education of Youth*. UK : Oxford University Press, 1957, p. 411.
- [3] MALONE, K., WAITE, S. *Student Outcomes and Natural Schooling: Pathways from Evidence to Impact Report*. Plymouth : Plymouth University, 2016, p. 50.
- [4] LEARNING AND TEACHING SCOTLAND. *Curriculum for Excellence*. <https://education.gov.scot/documents/All-experiencesoutcomes18.pdf>
- [5] AUSTRALIAN GOVERNMENT. *Living Sustainably: The Australian Government's National Action Plan for Education for Sustainability*. Canberra : Australian Government, 2009, p. 32. ISBN 978-0-646-50992-1.
- [6] WAITE, S. Where Are We Going? International Views on Purposes, Practices and Barriers in School-Based Outdoor Learning. In *Education Sciences*. 2020, vol. 10, no. 11. DOI=<https://doi.org/10.3390/educsci10110311>

Kognitívna náročnosť úloh v súťaži iBobor

Cognitive difficulty of tasks in competition iBobor

Klára Horváthová

KDMFI FMFI

Univerzita Komenského v Bratislave

Slovensko

horvathova115@uniba.sk

ABSTRACT

International informatic competition Beavers (iBobor in Slovakia) is already from 2004 helping to continually develop logical and informatic thinking of pupils [1]. Last year this competition attended more than 55 000 pupils from primary and secondary schools in Slovakia. Thanks to the great attendance, we have every year lot of useful data. Various research in the area of informatics and programming builds on the findings from the Beavers competition. In 2018, we have conducted a research [2] in collaboration with K. Miková (Mayerová) focused on cognitive difficulty of tasks in the Beavers competition. We used a categorization proposed in [3] and with decision tree algorithms (ID3, CHAID) we tried to predict the task difficulty according to given attributes. We succeeded to predict the task difficulty in 84,38%. However, we found out that the proposed categorization needs to be updated, because the same combination of attributes describing various tasks give them various difficulty evaluation, e.g. evaluate them as difficult and easy at the same time. Thus, we suggest using the informatic constructs described in [4] to develop a new categorization and its validation.

Keywords

Cognitive difficulty. Decision trees. ID3. CHAID. Informatic constructs.

ABSTRAKT

Medzinárodná informatická súťaž iBobor už od roku 2004 pomáha kontinuálne rozvíjať logické a informatické myslenie žiakov [1]. Minulý rok sa len na Slovensku zúčastnilo tejto súťaže viac ako 55 000 žiakov zo základných a stredných škôl. Vďaka tomu máme každoročne rozsiahle súbory dát. Viacero výskumov v oblasti informatiky a programovania vychádza práve zo zistení, ktoré ponúka súťaž iBobor. V roku 2018 sme spolu s K. Mikovou (Mayerovou) uskutočnili výskum [2] zameraný na kognitívnu náročnosť úloh v súťaži iBobor, kde sme sa s využitím kategorizácie navrhutej v [3] a s algoritmi rozhodovacích stromov (ID3, CHAID) pokúsili predikovať náročnosť úloh podľa daných atribútov. Podarilo sa nám predikovať náročnosť úloh až na 84,38%. Zistili sme však, že navrhnutá kategorizácia potrebuje upraviť, lebo rovnaká kombinácia atribútov ohodnotí úlohu aj ako ťažkú aj ľahkú. Navrhujeme preto využitie informatických konštruktov popísaných v [4] na vytvorenie novej kategorizácie úloh a ich následné otestovanie.

Kľúčové slová

Kognitívna náročnosť. Rozhodovacie stromy. ID3. CHAID. Informatické konštrukty.

1 ÚVOD

Medzinárodnej informatickej súťaže iBobor sa každoročne zúčastňuje viac ako 80 000 žiakov¹. Táto súťaž sa organizuje od 2. ročníka na ZŠ až po oktávu, či 5. ročník na SŠ. Súťaž pozostáva z ôsmich súťažných kategórií, navrhnutých podľa veku a špeciálnych obmedzení žiakov². Súťažné úlohy sú podľa predpokladanej náročnosti rozdelené na ľahké, stredne ťažké a ťažké. Za správne riešenie úlohy dostane súťažiaci počet bodov podľa zodpovedajúcej náročnosti úlohy. Najviac bodov dostane za správne vyriešenú ťažkú úlohu a najmenej za správne vyriešenú ľahkú úlohu. Podobne sa za nesprávne vyriešené úlohy súťažiacim odpočítavajú body. Úlohy sú ohodnotené podľa náročnosti ešte pred začatím súťaže a pre každú súťažnú kategóriu je rovnaký pomer ťažkých, stredne ťažkých a ľahkých úloh. Správne navrhnutie náročnosti je veľmi dôležité, a preto sa viacero výskumov venuje tomuto procesu. Cieľom našej práce bolo zautomatizovanie určovania kognitívnej náročnosti úloh, ktoré by pomohlo organizátorom súťaže predikovať náročnosť úloh s lepšou presnosťou.

1.1 Kategorizácia

Aby sme mohli úspešne zautomatizovať predikciu náročnosti úloh, potrebovali sme súťažné úlohy efektívne kategorizovať.

Poslúžila nám na to novo navrhnutá kategorizácia od Budinskej, Mayerovej a Veselskej [3]. Táto kategorizácia vznikla z dôvodu náročnej kategorizácie zahraničných úloh, ktoré nezapadali do kategórií ŠVP³. Pri tvorbe kategorizácie sa autorky sústredili na oblasti kognitívnych požiadaviek podľa Ahmed a Politt [5]:

Komplexnosť, Abstrakcia, Zdroje a Stratégia. Autorky teda vytvorili štyri kategórie s ďalšími podkategóriami reflektujúcimi kognitívne požiadavky úloh.

- | | | |
|----|----------------|--|
| 1. | Užívateľské | základné a kombinované |
| 2. | Logické | výrokové, grafové, iné |
| 3. | Algoritmické | pravidlá: dané, identifikuje
odpoveď: tvorí, vyberá |
| 4. | Programátorské | vytvára, interpretuje |

1.2 Rozhodovacie stromy

Na navrhnutie a zostrojenie rozhodovacích stromov, na predikovanie obtiažnosti úloh, sme využili dáta zo súťaže iBobor z rokov 2012 až 2018. Dáta obsahovali okrem iného predikovanú náročnosť úlohy a jej reálnu náročnosť podľa úspešného alebo nesprávneho riešenia danej úlohy súťažiacimi. Zamerali sme sa iba na vekovú kategóriu Drobci⁴ (2. a 3. ročník ZŠ) a Bobriči (4.-

¹ V roku 2020 sa súťaže zúčastnilo iba 50 000 žiakov.

² Vekové kategórie a iné nájdete na stránke súťaže
<https://ibobor.sk/>

³ Štátny vzdelávací program (2014)

⁴ Súťažná kategória Drobci sa do súťaže zaradila až v r. 2017.

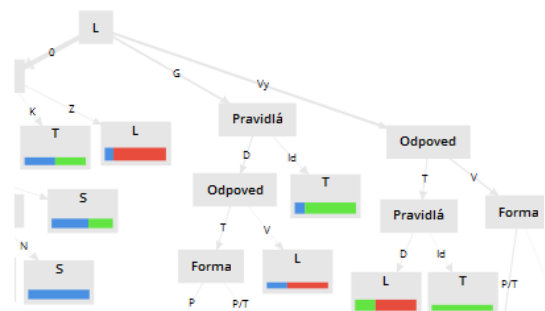
5.ročník ZŠ). Úlohy z uvažovaných rokov a vekových kategórií sme kategorizovali podľa kategorizácie predstavenej v 1.1. Dáta zo súťaží ohľadom úspešnosti riešení úloh sme rozdelili na tréningové (80% dát) a testovacie (20% dát). Pomocou tréningových dát sme skonštruovali dva rozhodovacie stromy použitím dvoch algoritmov – CHAID a ID3. Zvolili sme si tieto dva algoritmy, lebo ich ponúka program RapidMiner⁵, ktorý sme využili na tvorbu rozhodovacích stromov a taktiež preto, že tieto algoritmy využívajú rozdielne metódy rozdelenia atribútov do vrcholov stromov. Po prvých testoch kategorizácie sme zistili, že pre potreby rozhodovacích stromov kategorizáciu musíme upraviť. Finálna kategorizácia použitá na vybudovanie rozhodovacích stromov je v tabuľke 1.2. Na záver sme skontrolovali presnosť predikcie použitím testovacích dát.

Tabuľka 1.2: Finálna kategorizácia úloh použitá na tvorbu rozhodovacích stromov.

L= Logické	Vy=Výrokové, G=Grafové, 0 = nepatrí medzi logické
U= Užívateľské	Z=Základné, K=kombinované, 0 = nepatrí medzi užívateľské
A= Algoritmické	Po=Postupnosť, 0 = nepatrí medzi algoritmické
P= Programátorské	V=Vytváram, In=Interpretujem, 0 = nepatrí medzi programátorské
Pravidlá	D=Dané, In=Interpretujem
Odpoveď	V=Vyberám, T=Tvorím
Forma	P=Obrázok, T=Text, P/T=Obrázok doplnený textom
Interaktívne	I = Interaktívne, N = Neinteraktívne

2 VÝSLEDKY

Podarilo sa nám navrhnúť rozhodovacie stromy s presnosťou predikcie náročnosti úloh na 84,38% s využitím algoritmu ID3 a 73,44% s využitím algoritmu CHAID. Ukážka rozhodovacieho stromu vybudovaného použitím algoritmu CHAID je na obrázku 1.



Obrázok 1: Časť rozhodovacieho stromu vybudovaného algoritmom CHAID.

Síce presnosť predikcie je pomerne zaujímavá, odhalili sme v kategorizovaných dátach veľký nedostatok. Daná kategorizácia umožnila úlohám pri rovnakom ohodnotení atribútov nadobúdať

rôzne náročnosti. Pre niektoré logické úlohy dokonca rovnaké atribúty nadobudli všetky tri úrovne náročnosti.

3 DISKUSIA

Naše výsledky ukázali, že je veľmi dôležité správne navrhnúť kategorizáciu úloh, a to nielen do tematických kategórií ale aj podľa kognitívnych požiadaviek a možných operácií s nimi. Takúto kategorizáciu pre programátorské úlohy navrhuje Kalaš a Tomcsányiová [4]. Pri rôznych úlohách určujú, aké infromatické konštrukty je potrebné použiť na jej úspešné splnenie. Infromatický konštrukt definovali Kalaš a Benton [6], ako infromatický koncept a infromatické operácie, spojené s daným konceptom. Navrhujeme, aby sa ďalší výskum predikcie náročnosti úloh súťaže iBodor venoval detailnejšej kategorizácii s použitím infromatických konštruktov.

4 ZÁVER

Vedieť správne odhadnúť náročnosť úloh môže byť nápomocné aj pri navrhovaní správnej postupnosti gradácie infromatických konštruktov. Kalaš a Gjúberová [7] navrhujú využitie takej postupnosti na zlepšenie vyučovania, aby sa získané vedomosti stávali trvalými poznatkami. Taktiež zdôrazňujú ich úlohu pri navrhovaní národných kurikul, v ktorých by sa nevymenovávali len koncepty, ktoré sa žiaci majú naučiť, ale aj aké operácie s nimi budú vedieť vykonávať.

POĎAKOVANIE

Tento článok môže byť publikovaný na konferencii Didinfo 2021 vďaka agentúre VEGA, prostredníctvom projektu 1/0602/20.

BIBLIOGRAFICKÉ ODKAZY

- [1] DAGIENE, V.: Information Technology Contests – Introduction to Computer Science in an Attractive Way. *Informatics in Education* 5(1), 37–46 (2006)
- [2] HORVÁTHOVÁ, K. 2018. Kognitívna náročnosť úloh v súťaži iBodor. Bakalárska práca. Bratislava: FMFI UK, 2018.
- [3] BUDINSKÁ, L. MAYEROVÁ, K. VESELOVSKÁ, M. Bebras task analysis in category little beavers in Slovakia. In *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, s 91–101. Springer, 2017.
- [4] KALAŠ, I., TOMCSANYIOVA, M. Students' attitude to programming in modern informatics. *Proc. 9th WCCE 2009, Education and Technology for a Better World*, 2009.
- [5] POLLITT, A., AHMED, A. A new model of the question answering process. In: *IAEA conference, Slovenia, May 1999*. 1999.
- [6] KALAŠ, I., BENTON, L. Defining procedures in early computing education. In: *IFIP World Conference on Computers in Education*. Springer, Cham, 2017. p. 567-578.
- [7] GÚBEROVA, M., KALAŠ, I. Designing productive gradations of tasks in primary programming education. In: *Proceedings of the 8th Workshop in Primary and Secondary Computing Education*. 2013. p. 108-117.

⁵ softvér RapidMiner Studio Version 8.0, <https://rapidminer.com/>

BBC micro:bit ako súčasť projektu ENTER

BBC micro:bit as a Part of ENTER Project

Marek Mansell
SPy o.z.
Sibírska 21
83102 Bratislava
Slovenská republika
marek.mansell@pycon.sk

Nika Klimová
Univerzita Konštantína Filozofa
Tr. A. Hlinku 1
949 74 Nitra
Slovenská republika
nika.klimova@ukf.sk

Martin Cápaj
Univerzita Konštantína Filozofa
Tr. A. Hlinku 1
949 74 Nitra
Slovenská republika
mcapaj@ukf.sk

Magdaléna Bellayová
eTeacher o.z.
Štefánikova 57
949 01 Nitra
Slovenská republika
magdalena.bellayova@pycon.sk

ABSTRACT

Programming in schools can take various forms, from using of pre-prepared microworlds, through the programming of virtual characters living in a limited world, to the programming of real devices. The ENTER project aims to develop pupils' digital skills with the use of the BBC micro:bit, a physical device capable of responding to user stimuli through various sensors. By using such an approach in programming, it is possible to connect programming knowledge with the real experiences of students in an engaging way and thus increase their motivation and desire to program. The aim of the project is to support schools with hardware equipment but also methodological support and thus contribute to innovation in Slovak education.

Keywords

Programming. Hardware. Sensors.

ABSTRAKT

Programovanie na školách môže mať rôzne podoby, od využívania predpripravených mikrosvetov, cez programovanie virtuálnych postaviek žijúcich v ohraničenom svete, až po programovanie reálnych zariadení. Projekt ENTER je zameraný na rozvoj digitálnych zručností žiakov prostredníctvom využívania BBC micro:bit, fyzického zariadenia schopného cez rôzne senzory reagovať na podnety používateľa. Využívaním takéhoto prístupu v programovaní je možné pútať formou prepojiť programátorské poznatky s reálnymi zážitkami žiakov a tak zvýšiť ich motiváciu a chuť programovať. Cieľom projektu je podporiť školy hardvérovým vybavením, ale aj metodickou podporou, a tak prispieť k inovácii vo vzdelávaní na Slovensku.

Kľúčové slová

Programovanie. Hardvér. Senzory.

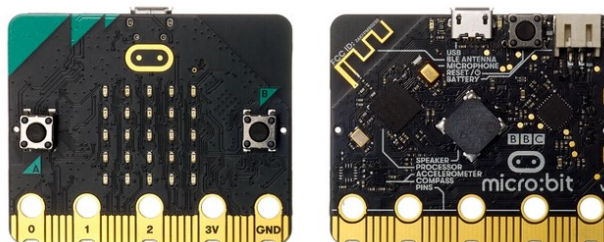
1 BBC MICRO:BIT (v1.5 a v2)

BBC micro:bit bol prvýkrát predstavený 7. júla 2015 ako zariadenie, ktoré je vhodné na edukačné účely. O mesiac na to sa učitelia z Veľkej Británie zapojili do školení a vo februári 2016 sa začali využívať v britských školách [1]. Veľký záujem o BBC micro:bit vyplýval najmä z jeho vlastností. Onedlho Micro:bit Educational Foundation vytvoril koncepty vyučovacích hodín, ktoré sú prepojené s britským kurikulumom (KS2, KS3, KS4) v predmete Computing [2].

BBC micro:bit je postavený na prístupe, kedy poznanie si buduje žiak. Papert z Piagetovho konštruktivismu odvodil konštrukcionizmus, kedy žiak je v roli konštruktéra/tvorcu a má k dispozícii hmatateľnú, reálnu, viditeľnú vec [3,4]. Okrem toho zariadenia, ktoré sa dajú chytiť, podporujú brikolážny prístup, ktorý podľa Levi-Straussa, Turkle, Paperta a Stillera predstavuje kontrast k analytickému prístupu [4]. Ide o spôsob, kedy myšlienkový

postup nezačína na axiómoch, ale poznatky (konštrukty) sa budujú na základe usporiadania, preskupenia a skúmania už známych „vecí“.

BBC micro:bit verzia 1.5 sa skladá z 5x5 červených LED diód na zobrazovanie obrázkov, textu a čísel a dá sa s nimi merať intenzita svetla dopadajúca na LED diódy. Okrem toho má 2 programovateľné tlačidlá, RESET tlačidlo na zadnej strane, anténu, procesor, žltú indikačnú LED diódu, micro USB konektor, konektor na batérie, kompas a akcelerometer [5]. Následne sa tam nachádzajú vstupno/výstupné piny, ktoré sú programovateľné, prip. majú skratku GND (ZEM) a 3V (na napájanie externých senzorov). BBC micro:bit verzia 2 bola ohlásená v októbri 2020 a obsahuje v sebe pár zmien a nové funkcie [6]. Upravili sa aj vizuálne zmeny, napríklad vzhľad vstupno-výstupných pinov či anténa (obr. 1).



Obrázok 1: BBC micro:bit verzia 2

Tieto zmeny sa dotkli aj programovacieho prostredia Microsoft MakeCode, kedy sa do kategórií *Vstup*, *Hudba* a *Koliky* vložili nové príkazy pred sekciou *micro:bit (V2)*. Príkazy sú kompatibilné len s BBC micro:bit v2 (Tab. 1), pričom pri nahratí takéhoto programu do verzie 1 sa zobrazuje chyba 927 (nekompatibilita) a program sa nespustí.

Tabuľka 1: Porovnanie dvoch verzií BBC micro:bit-u [2]

Funkcia	BBC micro:bit v1.5	BBC micro:bit v2
Procesor	nRF51822	nRF52833
Bluetooth	Bluetooth 4.0	Bluetooth 5.0
Šetriaci mód energie	Nie je	Áno
Mikrofón	Nie je	Integrovaný s indikačnou LED diódou
Reproduktor	Nie je	Integrovaný

Kapacitné dotykové tlačidlo	Nie je	Áno (Logo na prednej strane)
Indikátor napájania	Nie je	LED dióda
Elektrický prúd pre pripojené zariadenia	90 mA	200 mA
Flash pamäť	256 kB	512 kB
RAM pamäť	16 kB	125 kB
GPIO	3	4

2 PROJEKT ENTER

Projekt ENTER bol oficiálne spustený v septembri 2020 [7]. Projekt je financovaný Nadačným fondom Slovak Telekom, pričom na projekte sa podieľajú aj občianske združenia SPy, eTeacher, Aj Ty v IT a Nadácia Pontis. Cieľom projektu je pripraviť a podporiť riešenia, ktoré prinesú inovácie vo vzdelávaní a rozvinú digitálne zručnosti žiakov. Na naplnenie tohto cieľa sa realizuje viacero aktivít:

- Grantový program *ENTER Digitálna generácia* pre základné a stredné školy, v rámci ktorého bolo z 388 žiadostí podporených 243 škôl sumou v celkovej výške 229-tisíc eur. Financie boli určené výhradne materiál, t.j. kúpu micro:bitov a rozšírení k micro:bitom.
- Školenia pre učiteľov na prácu s micro:bitmi. Pôvodne boli plánované ako prezenčné školenia po celom Slovensku, avšak vzhľadom na pandemickú situáciu sú momentálne všetky realizované ako online školenia. Počas online školení sú učitelia vedení lektormi k riešeniu konkrétnych názorných úloh, ktoré môže využiť vo svojom vyučovaní. Vzhľadom k množstvu úloh boli školenia kategorizované na [5]:
 - Základy na BBC micro:bit,
 - Bezdrôtová komunikácia, hudba a elektrické obvody s BBC micro:bit,
 - LED diódy a LED pásiky s BBC micro:bit,
 - Komplexnejšie projekty s BBC micro:bit,
 - Smart Home Kit,
 - Elecrow Basic Kit,
 - Ring:bit Car Kit V2,
 - MOVE: mini robot.
- Do digitálnej knižnice enter.study bolo vytvorených viacero či už textových návodov a metodík pre učiteľov a žiakov, alebo aj videonávodov na pokročilé projekty s micro:bitmi (obr. 2).
- Veľmi dôležitým aspektom projektu je popularizácia témy digitálneho vzdelávania priamo medzi žiakmi. Za tým účelom boli do projektu prizvaní influenceri, a to najmä youtuberi GoGo a Patra Bene. V ich videách sa diváci môžu zoznámiť s využitím micro:bitov. V niektorých videách vystupujú aj známi hostia, ako napríklad cyklista Peter Sagan, víťazka Česko Slovensko má talent Nikoleta Šurínová či komik Bekim.
- Okrem žiakov sa projekt zamerával aj na popularizáciu témy digitálneho vzdelávania medzi širokou verejnosťou – najmä reportážami o využití micro:bitov a dôležitosti digitálnych zručností v Televíznych novinách, Teleráne či v online médiách.
- Súťaž pre žiakov, ktorá mala podporiť tímovú prácu a projektové vyučovanie, bola vzhľadom na pandemickú

situáciu presunutá na školský rok 2021/2022. Taktiež bola presunutá popularizačná roadshow po školách, v rámci ktorej mali známe osobnosti žiakom ukazovať využitie micro:bitov.

- Názorné online hodiny pre žiakov s micro:bitmi, v rámci ktorých sa žiaci môžu zoznámiť s BBC micro:bit a prostredím Microsoft MakeCode si naprogramovať niekoľko jednoduchých aktivít s micro:bitom.



Obrázok 2: Digitálna knižnica

3 ZÁVER

Projektom ENTER sa Slovensko zaradí na mapu štátov, ktoré sa aktívne zapájajú do inovácie obsahu vzdelávania cez programovanie hardvéru. Školenia, ktoré aktuálne prebiehajú majú učiteľom pomôcť pri získavaní praktických skúseností nielen pri práci s minipočítačom, ale aj v práci vo virtuálnom vzdelávacom prostredí. Našou snahou je to, aby učitelia dostali okrem podpory pri zabezpečení hardvérového vybavenia školy, aj dostatočnú metodickú podporu a aby boli pripravení na rôzne formy vzdelávania. Z priebežnej spätnej väzby je zatiaľ zjavné, že sa ciele darí postupne naplňovať. Veríme tiež, že výstupy projektu budú tiež dostatočné na to, aby sa vytvorila silná komunita učiteľov, ktorí môžu aj so svojimi žiakmi posúvať myšlienku projektu vpred aj v budúcnosti.

POĎAKOVANIE

Tento článok bol finančne podporený Univerzitnou grantovou agentúrou a projektom Implementácia nových trendov v informatike do výučby algoritmického myslenia a programovania v predmete informatika v sekundárnom vzdelávaní (018UMB-4/2020).

BIBLIOGRAFICKÉ ODKAZY

- [1] BALL, T. Microsoft Touch Develop and the BBC micro:bit. In *ICSE '16*. 2016, p. 637-640.
- [2] Micro:bit Educational Foundation [online] [cit. 2021-02-06] <https://microbit.org/>
- [3] PAPERT, S., HAREL, I. Situating constructionism. In *Constructionism*, vol. 36, New York: Ablex Publishing, 1991.
- [4] STILLER, E. Teaching Programming Using Bricolage. In *Journal of Computing Sciences in Colleges*, vol. 24. 2009.
- [5] Učíme s Hardvérom [online] [cit. 2021-02-06] <https://www.ucimeshardverom.sk/>
- [6] Python blog [online] [cit. 2021-02-06] <https://python.sk/blog/microbit-v2/>
- [7] ENTER [online] [cit. 2021-02-06] <https://enter.study/>

Workshopy

Workshops

Základy práce s micro:bitom

Richard Balogh

Ústav automobilovej mechatroniky
FEI STU

balogh@elf.stuba.sk

ABSTRAKT

Na workshope si predstavíme malé úžasné zariadenie - micro:bit. Povieme si, čo to micro:bit je, na čo ho vieme využiť, zoznámime sa s programovacím prostredím makecode.org, v blokovom prostredí si naprogramujeme pár jednoduchých úloh, ktoré môžete študentom ukázať aj pri dištančnom vzdelávaní v online priestore. A že nemáte micro:bit? To vôbec nevádi, počas workshopu budeme pracovať so simulátorom micro:bitu. Tiež vám ukážeme zopár vychytávok, ako napríklad "Classroom", ktorý pri programovaní micro:bitu s triedou môžete perfektne využiť.

Programovanie mikrokontroléra Arduino

Patrik Voštinár, Vladimír Siládi, Michal Vagač

Katedra informatiky FPV UMB

Tajovského 40

974 09 Banská Bystrica

Slovensko

patrik.vostinar@umb.sk

ABSTRAKT

Na workshope si ukážeme prácu s mikrokontrolérom Aduino – čo je to mikrokontrolér, ako sa zapájajú LED svetlá, tlačidlá, potenciometer, buzdiak. Počas workshopu budeme zapájať tieto komponenty na stránke Tinkercad, nie je preto potrebné vlastniť sady s Arduinom.

Programovanie 2D hier v MakeCode Arcade

Patrik Voštinár, Jarmila Škrinárová

Katedra informatiky FPV UMB

Tajovského 40

974 09 Banská Bystrica

Slovensko

patrik.vostinar@umb.sk

ABSTRAKT

Na workshope si vyskúšame vytváranie 2D hier v prostredí MakeCode Arcade. Ukážeme si online prostredie, simulátor, zoznámime sa s programovacím prostredím makecode.org. Pomocou blokov kódu si vytvoríme jednoduchú 2D hru, ktorú môžete študentom ukázať.

Organizátori

Organisers



UNIVERZITA MATEJA BELA

Fakulta prírodných vied

Katedra informatiky

Katedra informatiky je pevnou súčasťou Fakulty prírodných vied Univerzity Mateja Bela. Má priznané právo udeľovať titul „bakalár“ a akademický titul „magister“ absolventom dennej formy štúdia a externej formy štúdia v študijnom programe Aplikovaná informatika a aj absolventom dennej formy štúdia v študijnom programe Učiteľstvo informatiky v kombináciách, ktoré sú zriadené podľa zákonných predpisov a uskutočňované na Fakulte prírodných vied Univerzity Mateja Bela. Katedra informatiky má priznané právo udeľovať akademický titul „doktor pedagogiky“ (v skratke „PaedDr.“) a akademický titul „doktor prírodných vied“ (v skratke „RNDr.“) absolventom študijných programov, ktorí získali titul „magister“, po vykonaní rigorózneho skúšky v študijnom odbore, v ktorom získali vysokoškolské vzdelanie alebo v príbuznom študijnom odbore uskutočňovanom na Fakulte prírodných vied Univerzity Mateja Bela. Katedra informatiky vykonáva aj výučbu mnohých celofakultných predmetov a celouniverzitných predmetov, ktoré sú buď povinnými, povinne voliteľnými alebo výberovými predmetmi na katedrách FPV UMB v Banskej Bystrici, resp. na iných katedrách; aj v rámci spoločného základu.

Zameranie edukačnej, vedeckej a výskumnej činnosti Katedry informatiky FPV UMB sa odvíja od hlavnej výskumnej línie fakulty, profilácie katedry a od záujmu jednotlivých členov katedry v symbióze s potrebou obohacovať predmety v študijných programoch o najnovšie poznatky vedy a výskumu v odbore. Hlavnými smermi činnosti katedry sú:

- Teória vyučovania informatiky.
- Softvérové inžinierstvo.
- Umelá inteligencia.
- Aplikovaná informatika.

Katedru informatiky charakterizuje:

- vysoký dopyt po absolventoch našich odborov,
- široké spektrum uplatnenia absolventov v učiteľskej i odbornej praxi,
- kontakt so školami i s firmami z oblasti IT už počas štúdia,
- viacročná tradícia v príprave učiteľov a odborníkov pre oblasť informatiky,
- ústretové výučbové prostredie,
- tím skúsených akademických a vedeckých pracovníkov katedry,
- previazanosť teoretických a praktických poznatkov,
- primerane vybavené laboratóriá,
- možnosť absolvovať časť odborného štúdia na zahraničnej partnerskej univerzite.

Katedra je medzinárodne uznávaným a akceptovaným pracoviskom. Má kontakty s rôznymi domácimi aj zahraničnými pracoviskami a organizáciami, s ktorými dochádza k pravidelnej výmene učiteľov, študentov i ďalšej spolupráci v oblasti vedy a vzdelávania: TU Liberec, TU Ostrava, UP Olomouc, MU a ČVUT Praha, MU Brno (Česká republika); Petroleum-Gas University, Ploiești (Rumunsko); "Prof. Dr. Assen Zlatarov" University, Burgas (Bulharsko); Oulu University of Applied Sciences, Oulu (Fínsko); Akdeniz University, Antalya a Alanya Alaaddin Keykubat University, (Turecko); Utrecht University, Utrecht (Holandsko); Liepaja University, Liepaja (Lotyšsko); Corvinus University of Budapest (Maďarsko); Technische Universität Chemnitz (Nemecko) a ďalšie.



Na katedře aplikované matematiky najdete lidi se dvojitým odborným zaměřením. Jednak jsou zde, jak název napovídá, matematici. Ti se zaměřují především na výuku matematiky aplikované v ostatních disciplínách. Jedná se především o diskrétní matematiku, lineární algebru a statistiku. Druhou skupinou jsou informatici. Ti se podílí na dalším rozvoji ICT gramotnosti studentů bez ohledu na jejich profilaci. U budoucích učitelů se zaměřujeme na zvládnutí vybraných technologií, aby pak dále měli dobrý základ pro rozvoj počítačem podporované výuky ve svých specializacích. Pro budoucí učitele informatiky pak katedra zajišťuje část odborných předmětů, jako jsou kódování a šifrování či matematika pro informatiky. Informatická část katedry se pak soustředí v rámci přípravy těchto učitelů na didaktiku informatiky, programování a počítačem podporovanou výuku. Je to pracoviště, pod nímž jsou akreditovány všechny studijní obory připravující budoucí učitele informatiky na univerzitě – jak v bakalářském, tak navazujícím magisterském stupni. Část kolegů se specializuje na GIS a jejich využití.

Členové katedry se také zapojují do komunitního života v Jednotě školských informatiků a v Jednotě českých matematiků a fyziků. Spolupodílíme se na organizaci Bobříka informatiky a Matematické olympiády. Katedra se významně podílí i na řešení různých projektů na fakultě. Mezi nejvýznamnější projekty z poslední doby patřily:

- Podpora rozvíjení informatického myšlení,
- Podpora rozvoje digitální gramotnosti,
- Učitelem moderně a odborně,
- Zvýšení kvality vzdělávání žáků, rozvoje klíčových kompetencí, oblastí vzdělávání a gramotností,
- Prameny spojují krajiny a lidi.

Informatická část dlouhodobě spolupracuje s Národním pedagogickým institutem a má i několik dlouhodobých partnerů v zahraničí. Patří mezi ně Univerzita Mateje Bela v Banské Bystrici, se kterou spolupracujeme právě na pořádání konference Didinfo. V kontaktu jsme také s kolegy z Univerzity P. J. Šafárika v Košiciach, Univerzity Komenského v Bratislavě, Vilniaus universitetas nebo Carl von Ossietzky Universität Oldenburg. Těšíme se, že se nám jednou podaří po obměně a doplnění informatické sekce navázat i užší výzkumnou spolupráci.

Sponzori a mediálni partneri
Sponsors and media partners

NEXTTECH

Quark
Magazín o vede a technike

Continental 
The Future in Motion



lāskarduino
bastli s láskou

 **it** **AKADÉMIA**



**BANSKOBYSŤRICKÝ
SAMOSPRÁVNÝ KRAJ**

Názov: **DidInfo 2021**
Editori: Ing. Dana Horváthová, PhD., RNDr. Alžbeta Michalíková, PhD.,
doc. Ing. Jarmila Škrinárová, PhD., PaedDr. Patrik Voštinár, PhD.
Vydanie: 1. vydanie
Vydavateľ: Univerzita Mateja Bela, Fakulta prírodných vied v Banskej Bystrici, Slovensko
Rok: 2021
Rozsah: 193 strán
Formát: elektronický zborník
ISBN: 978-80-557-1823-1
ISSN: 2454-051X

Title: **Proceedings of conference DidInfo 2021**
Editors: Ing. Dana Horváthová, PhD., RNDr. Alžbeta Michalíková, PhD.,
doc. Ing. Jarmila Škrinárová, PhD., PaedDr. Patrik Voštinár, PhD.
Edition: First edition
Publisher: Matej Bel University, Faculty of Natural Sciences in Banská Bystrica, Slovakia
Year: 2021
Pages: 193
Format: electronic proceedings
ISBN: 978-80-557-1823-1
ISSN: 2454-051X