



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta přírodovědně-humanitní
a pedagogická



FAKULTA PRÍRODNÝCH VIED
UNIVERZITA MATEJA BELA, BANSKÁ BYSTRICA



Sborník konference



EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY

citační bibliografické údaje

DRÁBKOVÁ Jindra a Jan BERKL, eds. *Sborník konference Didinfo 2020*
[online]. Liberec, 2020. ISBN 978-80-7494-532-8, ISSN 2454-051X.

Dostupné na

http://www.didinfo.net/images/DidInfo/files/Didinfo_2020.pdf

Didinfo 2020

Mezinárodní konference o vyučování informatiky

12. až 14. února 2020 | Liberec | Česká republika

Programový výbor konference:

doc. Ing. Jarmila Škrinářová, PhD., Univerzita Mateja Bela v Banskej Bystrici (SK) – předsedkyně

Mgr. Jan Berki, Ph.D., Technická univerzita v Liberci (CZ) – šéf organizace

doc. RNDr. Gabriela Andrejková, CSc., Univerzita Pavla Jozefa Šafárika v Košiciach (SK)

Sonya Armstrong, Ph.D., West Virginia State University (US)

Mgr. Martin Cápay, PhD., Univerzita Konštantína Filozofa v Nitre (SK)

doc. RNDr. Miroslava Černochová, CSc., Univerzita Karlova v Praze (CZ)

prof. Dr. Valentina Dagiene, Vilniaus universitetas (LT)

Ing. Dana Horváthová, PhD., Univerzita Mateja Bela v Banskej Bystrici (SK)

prof. Dr. Mirjana Ivanovic, Univerzitet u Novom Sadu (SRB)

Ing. Jana Jacková, PhD., Univerzita Mateja Bela Banská Bystrica (SK)

prof. RNDr. Ivan Kalaš, PhD., Univerzita Komenského v Bratislave (SK)

doc. RNDr. Zuzana Kubincová, PhD., Univerzita Komenského v Bratislave (SK)

doc. RNDr. Gabriela Lovászová, PhD., Univerzita Konštantína Filozofa v Nitre (SK)

Ing. Božena Mannová, Ph.D., České vysoké učení technické v Praze (CZ)

RNDr. Alžbeta Michalíková, PhD., Univerzita Mateja Bela v Banskej Bystrici (SK)

RNDr. Pavel Pešat, Ph.D., Národní institut pro další vzdělávání (CZ)

Prof. Dr. Kate Sanders, Rhode Island College (USA)

prof. Ing. Veronika Stoffová, CSc., Trnavská univerzita (SK)

doc. RNDr. Petr Šaloun, PhD., Technická univerzita Ostrava (CZ)

doc. RNDr. Ľubomír Šnajder, PhD., Univerzita Pavla Jozefa Šafárika v Košiciach (SK)

doc. RNDr. Pavel Töpfer, CSc., Univerzita Karlova v Praze (CZ)

doc. Ing. Ľudovít Trajtel', PhD., Univerzita Mateja Bela v Banskej Bystrici (SK)

Dr. Livia Tudor, Universitatea Petrol – Gaze din Ploiești (RO)

doc. PaedDr. Jiří Vaníček, Ph.D., Jihočeská univerzita v Českých Budějovicích (CZ)

PaedDr. Patrik Voštinár, PhD., Univerzita Mateja Bela v Banskej Bystrici (SK)

Recenzenti:

Mgr. Jan Berki, Ph.D., Technická univerzita v Liberci (CZ)
doc. RNDr. Miroslava Černochová, CSc., Univerzita Karlova v Praze (CZ)
Ing. Jindra Drábková, Ph.D., Technická univerzita v Liberci (CZ)
PhDr. Zbyněk Filipi, Ph.D., Západočeská univerzita v Plzni (CZ)
doc. Ing. Dalibor Frydrych, Ph.D., Technická univerzita v Liberci (CZ)
doc. RNDr. PaedDr. Hashim Habiballa, PhD., Ph.D., Ostravská univerzita (CZ)
doc. PhDr. Miroslav Chráska, Ph.D., Univerzita Palackého v Olomouci (CZ)
PhDr. Tomáš Jakeš, Ph.D., Západočeská univerzita v Plzni (CZ)
Mgr. Bc. Libor Klubal, Ph.D., Ostravská univerzita (CZ)
Ing. Jana Kolaja Ehlerová, Ph.D., Technická univerzita v Liberci (CZ)
doc. Ing. Kateřina Kostolanyová, Ph.D., Ostravská univerzita (CZ)
doc. PaedDr. Martina Maněnová, Ph.D., Univerzita Hradec Králové (CZ)
Ing. Božena Mannová, Ph.D., České vysoké učení technické v Praze (CZ)
Mgr. Miroslav Meier, Ph.D., Technická univerzita v Liberci (CZ)
PhDr. Ondřej Neumajer, Ph.D., Univerzita Karlova v Praze (CZ)
PhDr. Magda Nišpanská, Ph.D., Technická univerzita v Liberci (CZ)
PhDr. Jitka Novotová, Ph.D., Technická univerzita v Liberci (CZ)
PhDr. Lucie Rohlíková, Ph.D., Západočeská univerzita v Plzni (CZ)
doc. RNDr. Petr Šaloun, Ph.D., Technická univerzita Ostrava (CZ)
Mgr. Martin Slavík, Ph.D., Technická univerzita v Liberci (CZ)
doc. Mgr. František Tůma, Ph.D., Masarykova univerzita v Brně (CZ)
doc. Dr. PaedDr. Petr Urbánek, Technická univerzita v Liberci (CZ)
doc. PaedDr. Jiří Vaníček, Ph.D., Jihočeská univerzita v Českých Budějovicích (CZ)

Copyright © 2020 autoři

ISBN: 978-80-7494-532-8

ISSN: 2454-051X

Recenzovaný sborník

Obsah

Kurikulární souvislosti výuky informatiky

Informatika a revize rámcových vzdělávacích programů	12
<i>Daniela Růžičková</i>	
Nový standard studia ICT koordinátora/metodika	23
<i>Ondřej Neumajer</i>	
Reálné využívání digitálních technologií v mateřských školách	30
<i>Václav Dobiáš</i>	
Prínos informatiky a informačných technológií k výučbe prírodovedných predmetov	37
<i>Gabriela Andrejková, Zuzana Ješková, Marián Kíreš, Mária Ganajová a Mária Kožurková</i>	
Rozvoj kritického myslenia študentov učiteľstva informatiky prostredníctvom prípadových štúdií	45
<i>Gabriela Lovászová a Nika Klimová</i>	
Kooperativní učení ve výuce informatiky	53
<i>Rostislav Fojtík</i>	
Rovnaké alebo rôzne? Čo spôsobila zmena motivácie v dvoch úlohách v súťaži iBobor	60
<i>Lucia Budinská a Karolína Miková</i>	
Kategorizácia teoretických úloh algoritmickej súťaže PRASK	68
<i>Michal Anderle</i>	
Úroveň informatického myslenia žiakov zapojených do Národného projektu IT Akadémia	76
<i>Ján Guniš, Ľubomír Šnajder, Veronika Jurková a Zuzana Tkáčová</i>	
Závislosť na internetu, počítačových hrách a on-line komunikaci u študentů učiteľských študijních programů	87
<i>Miroslav Chráska</i>	
Možnosť implementácie ochrany kultúrneho dedičstva do vyučovania informatiky	97
<i>Ondrej Takáč</i>	

Algoritmizace a programování ve škole

Hravá forma rozvíjania algoritmického myslenia na základnej škole	104
<i>Krisztina Czakoová a Veronika Stoffová</i>	
Ako na grafy vo vyučovaní informatiky na druhom stupni základných škôl	112
<i>Lucia Budinská a Karolína Miková</i>	
Prezentovanie binárneho vyhľadávania žiakom základných škôl prostredníctvom interaktívnej aplikácie	121
<i>Iveta Demková a Michal Anderle</i>	
Simulačné modely vo vyučovaní programovania.....	129
<i>Veronika Stoffova a Ladislav Végh</i>	
The Role of Sprego Programming in Teaching Informatics	141
<i>Maria Csernoch</i>	
Code review v škole a na programátorskom krúžku	149
<i>Iveta Demková a Zuzana Kubincová</i>	
Mikrosvet na výučbu programovania priateľský k nevidiacim študentom sekundárneho vzdelávania	156
<i>Lucia Budinská, Ľudmila Jašková, Michal Kováč, Karolína Mayerová a Mária Karasová</i>	
Jazyk Python ako hlavný nástroj vo výučbe informatických predmetov na stredných školách	164
<i>Miroslav Biňas, Matej Madeja a Danka Bruothová</i>	
Bádateľsky orientované metodiky pre použitie App Inventora na strednej škole	173
<i>Monika Tomcsányiová a Peter Tomcsányi</i>	
Ako spestriť výučbu informatiky na školách?	180
<i>Patrik Voštinár, Dana Horváthová a Jarmila Škrinárová</i>	
Prezentace vzdělávacích materiálů pro Scratch 3.0	188
<i>Ingrid Nagyová</i>	

Robotika ve vzdělávání

Online cloudové prostředí pro výuku informatiky a robotiky na ZŠ 197

Petr Mališ, Petr Šaloun, Lucie Bryndová, Petr Hrůza, Tomáš Dragon a Milan Klement

Ověřování učebnice LEGO robotiky pro podporu rozvoje informatického myšlení 203

Tomáš Jakeš, Jan Bařko a Filip Frank

Využití robotické stavebnice VEX IQ ve výuce na základních školách 212

Hana Hyksová

Simulácia procesov inteligentnej domácnosti s micro:bitom..... 220

Martin Cápay a Magdaléna Bellayová

Rozšířené abstrakty a názory

Zhmotňovanie nápadov pomocou digitálnych technológií výroby 230

Eva Kalužáková and Jozef Vaško

Východiská stratégie vzdelávania pre potreby IT odvetvia v regióne Košického samosprávneho kraja
..... 232

Martin Šechný, Juraj Ľapák, Martin Džbor and Viliam Fedák

Diskusní příspěvek k porovnání požadavků na informatiky a učitele informatiky 234

Libor Dostálek

Content

The Curriculum Context of Computer Science Teaching

Informatics and Innovation of Framework Education Programmes	12
<i>Daniela Růžičková</i>	
A New Study Standard for ICT Coordinator/Methodology Specialist	23
<i>Ondřej Neumajer</i>	
Real Use of Digital Technologies in Preschool Education	30
<i>Václav Dobiáš</i>	
The Contribution of Informatics and Information Technologies to Science Teaching	37
<i>Gabriela Andrejková, Zuzana Ješková, Marián Kíreš, Mária Ganajová a Mária Kožurková</i>	
Critical Thinking Development of Informatics Pre-Service Teachers with the Support of Case Studies	45
<i>Gabriela Lovászová a Nika Klimová</i>	
Cooperative Learning in Teaching Informatics	53
<i>Rostislav Fojtík</i>	
Are They the Same or Are They Different? What Effect Had the Change of Motivation on Two Bebras Task	60
<i>Lucia Budinská a Karolína Miková</i>	
The Algorithmic Tasks Categorization in the Middle School Competition PRASK	68
<i>Michal Anderle</i>	
The Level of Computer Thinking of Pupils Involved in the National IT Academy Project	76
<i>Ján Guniš, Ľubomír Šnajder, Veronika Jurková a Zuzana Tkáčová</i>	
Addiction to Internet, Computer Games and Online Communication of Students of Teacher Studies Programs	87
<i>Miroslav Chráska</i>	
The Possibility of Implementing the Protection of Cultural Heritage in the Teaching of Informatics	97
<i>Ondrej Takáč</i>	

Algorithm Development and Programming at School

A Playful Form of Developing Algorithmic Thinking in Primary School	104
<i>Krisztina Czakoóová a Veronika Stoffová</i>	
How to Include Graphs in Informatics Education in Lower Secondary Schools.....	112
<i>Lucia Budinská a Karolína Miková</i>	
The Presentation of the Binary Search Algorithm to Middle Schoolers Using the Interactive Application	121
<i>Iveta Demková a Michal Anderle</i>	
Simulation Models in Programming Teaching.....	129
<i>Veronika Stoffova a Ladislav Végh</i>	
The Role of Sprego Programming in Teaching Informatics	141
<i>Maria Csernoch</i>	
Code Review at School and at Programming Club	149
<i>Iveta Demková a Zuzana Kubincová</i>	
Microworld for Learning Programming Friendly to Secondary Blind Students.....	156
<i>Lucia Budinská, Ľudmila Jašková, Michal Kováč, Karolína Mayerová a Mária Karasová</i>	
Python as the Main Programming Language in the Teaching of Computer Courses in High Schools	164
<i>Miroslav Biňas, Matej Madeja a Danka Bruothová</i>	
Exploration-Oriented Teacher Materials for Using App Inventor in Upper Secondary Schools	173
<i>Monika Tomcsányiová a Peter Tomcsányi</i>	
How to Brighten up the Teaching of Computer Science at Schools?	180
<i>Patrik Voštinár, Dana Horváthová a Jarmila Škrinárová</i>	
Presentation of Educational Materials for Scratch 3.0	188
<i>Ingrid Nagyová</i>	

Robotics in Education

The Online Cloud Environments for Teaching Computer Science and Robotics at Elementary School	197
---	-----

Petr Mališ, Petr Šaloun, Lucie Bryndová, Petr Hrůza, Tomáš Dragon a Milan Klement

Verification of the LEGO Robotics Textbook to Support the Development of Computational Thinking	203
---	-----

Tomáš Jakeš, Jan Batko a Filip Frank

Use of VEX IQ Robotic Kit in Primary Schools	212
--	-----

Hana Hyksová

The Simulation of Smart Home Processes with a Micro:bit	220
---	-----

Martin Cápay a Magdaléna Bellayová

Extended Abstracts and Opinions

Tangling Ideas Using Digital Manufacturing Technologies	230
---	-----

Eva Kalužáková and Jozef Vaško

Baseline of Training Strategy for the IT Sector in the Košice Region	232
--	-----

Martin Šechný, Juraj Ťapák, Martin Džbor and Viliam Fedák

Discussion Paper to Compare Requests for Computer Science Experts and Computer Science Teachers	234
---	-----

Libor Dostálek

Kurikulární souvislosti výuky informatiky

The Curriculum Context of Computer Science Teaching

Informatika a revize rámcových vzdělávacích programů

Informatics and Innovation of Framework Education Programmes

Daniela Růžicková
Národní pedagogický institut
Weilova 1271
102 00 Praha 10
Česká republika
daniela.ruzickova@npicr.cz

ABSTRAKT

V dubnu roku 2016 zahájil Národní ústav pro vzdělávání práce na inovacích ICT kurikula v rámcových vzdělávacích programech. Byla připravena koncepce rozvoje digitální gramotnosti a informatického myšlení dětí a žáků a návrh vzdělávacího obsahu vzdělávací oblasti Informatika. Příspěvek podává zprávu o průběhu a aktuálních výsledcích tohoto úkolu.

ABSTRACT

In April 2016, the National Institute for Education started work on innovations of ICT curriculum in Framework Educational Programmes. The concept of development of digital literacy and computational thinking of children and pupils was prepared and the educational content of the educational area of Informatics was designed. The paper reports on the progress and actual results of this task.

Klíčová slova

informatické myšlení, digitální gramotnost, informatika, národní kurikulum

Keywords

computational thinking, digital literacy, informatics, computer science, computing, national curriculum

1 ÚVOD

Informatika a informační a komunikační technologie byly zařazeny do závazného kurikula ve všeobecném vzdělávání zavedením rámcových vzdělávacích programů (RVP): v základním vzdělávání v roce 2005, na gymnáziích a ve středním odborném vzdělávání v roce 2007¹. V průběhu let, kdy tato vzdělávací oblast jako jedna z mála zůstávala v RVP beze změny, se nashromáždilo mnoho podnětů k jejich úpravám, některé časem zastaraly a přestaly být v konkrétní formulaci aktuální, některé přetrvaly dodnes. Už v roce 2008 odborná skupina tehdejšího Výzkumného ústavu pedagogického navrhovala zaměřit se na revizi rámcových vzdělávacích programů v oblasti ICT a doporučila, kromě jiného, inovovat obsah oblasti ICT společně s obsahem jiných oborů a rozdělit téma ICT v RVP na dva celky, digitální technologie a informatiku [1].

Tvorba kurikula a kurikulárních dokumentů, respektive **rozvoj závazného vzdělávacího obsahu** je disciplína, která vychází z odborných a pedagogických disciplín, řídí se společenskou poptávkou a je závislá na rozhodnutích státní správy [2].

Vznik vládní Strategie digitálního vzdělávání do roku 2020 (SDV) [3] umožnil se vývoji ICT kurikula věnovat s konkrétním zadáním revidovat RVP v oblasti ICT. Dva ze tří prioritních cílů této strategie jsou (1) rozvíjet digitální gramotnost a (2) informatické myšlení žáků. Zahájení výuky dle

¹ Rámcové vzdělávací programy pro střední odborné vzdělávání byly zaváděny ve třech vlnách, první vlna vešla v platnost v roce 2007.

DIDINFO 2020, Technická univerzita v Liberci, Fakulta přírodovědně-humanitní a pedagogická
aktualizovaných znění vzdělávací oblasti ICT v rámcových vzdělávacích programech je jeden z jejích sedmi indikátorů úspěchu.

Článek podává zprávu o průběhu, problémech a aktuálních výsledcích revizí ICT kurikula v rámcových vzdělávacích programech.

2 SOUČASNÝ STAV ICT KURIKULA

Vzdělávací oblast Informační a komunikační technologie v základním vzdělávání a převážná část vzdělávacích oblastí Informatika a informační technologie (gymnázia) a Vzdělávání v informačních a komunikačních technologiích (střední odborné vzdělávání) jsou zaměřeny na rozvoj schopnosti pracovat s ICT² a informacemi na uživatelské úrovni jako základ pro aplikaci v ostatních vzdělávacích oblastech [4], [5], [6]. Přehled tematických okruhů a časové dotace dává následující tabulka:

Tabulka 1: Přehled tematických okruhů a časové dotace informatiky a ICT v RVP

RVP ZV – ICT (po 1 hodině na každém stupni)	RVP G – Informatika a ICT (4 hodiny)	RVP SOV – Vzdělávání v ICT (nejčastěji 4 hodiny)
Základy práce s počítačem	Digitální technologie	Práce s počítačem, operační systém, soubory, adresářová struktura, souhrnné cíle
Vyhledávání informací a komunikace	Zdroje a vyhledávání informací, komunikace	Informační zdroje, celosvětová počítačová síť Internet Práce v lokální síti, elektronická komunikace, komunikační a přenosové možnosti Internetu
Zpracování a využití informací	Zpracování a prezentace informací	Práce se standardním aplikačním programovým vybavením

Tematický okruh Zpracování a prezentace informací (RVP G) obsahuje mj. i očekávaný výstup *aplikuje algoritmický přístup k řešení problémů*. Tematický okruh Práce s počítačem, operační systém, soubory, adresářová struktura, souhrnné cíle (RVP SOV) obsahuje mj. i očekávaný výstup *ovládá principy algoritmicizace úloh a sestavuje algoritmy řešení konkrétních úloh (dekompozice úlohy na jednotlivé elementárnější činnosti za použití přiměřené míry abstrakce)*. RVP SOV (na rozdíl od RVP ZV a RVP G) má mezi klíčovými kompetencemi i Kompetenci využívat prostředky informačních a komunikačních technologií a pracovat s informacemi a mezi průřezovými tématy i téma Informační a komunikační technologie.

Aplikační rovina se měla stát součástí ostatních oborů, popis ale nebyl na úrovni RVP dostatečně rozpracován. V praxi to znamená, že většina škol má v učebním plánu svého školního vzdělávacího programu (ŠVP) vyučovací předmět informatika či ICT, ve kterém je rozpracován vzdělávací obsah příslušné vzdělávací oblasti dle RVP. Přitom pojetí a orientaci těchto předmětů výrazně ovlivňuje úroveň (ICT, informatických) kompetencí, postoje a osobní preference učitelů, kteří tyto předměty

² Poznámka k terminologii: v textu se vyskytují termíny „informační a komunikační technologie“ či „ICT“ a „digitální technologie“ a na ně navazující termíny (vzdělávací oblast, gramotnost, kompetence apod.). Zpravidla je termín „ICT“ použit tam, kde se hovoří o současném kurikulu, kde je ve stávajících RVP používán, a „digitální“ tam, kde se hovoří o obsahu jeho inovací, v rámci kterých se pracovní skupina k revizím dohodla pro změnu terminologie. Jako každé pravidlo, má i toto výjimky (např. současný Rámcový vzdělávací program pro gymnázia), prosíme laskavého čtenáře o shovívavost.

vyučují [7]. Výzkum byl zaměřen na základní školy, nicméně zkušenosti ukazují, že situace na středních školách je obdobná. Současně většina učitelů ostatních předmětů necítí odpovědnost za rozvoj schopnosti aplikovat a dále rozvíjet vědomosti, dovednosti a postoje žáků získané při výuce předmětu ICT [8]. S rostoucími nároky na digitální kompetence je stále problematičtější většinu z nich rozvíjet odděleně od kontextu ostatní práce žáků ve škole, tj. směřovat rozvoj schopnosti zacházet s digitálními technologiemi do jediného vzdělávacího oboru, respektive předmětu [8]. Obor informatika je vědní obor, má vlastní obsah, její význam roste a základ, který školská informatika může nabídnout ostatním vzdělávacím oborům, směřuje k základním principům zpracování informací a ke schopnosti uplatnit informatické postupy při řešení problémů [9].

3 REVIZE RÁMCOVÝCH VZDĚLÁVACÍCH PROGRAMŮ A POSTAVENÍ INFORMATIKY V NICH

Návrh nového ICT kurikula je výsledkem práce skupiny odborných pracovníků kurikulárního ústavu (přímořizované organizace ministerstva školství), akademiků i učitelů z praxe, která zvažovala celý komplex otázek, jejich rámec lze shrnout do tří bodů: (1) čeho chceme dosáhnout, tj. cíle, které ideálně vidíme, (2) čemu jsme schopni poskytnout podporu, tj. máme představy o tom, jak by mohly vypadat pedagogické postupy a vzdělávací zdroje na podporu výuky, (3) co terén unese, tj. co jsou učitelé schopni, po zaučení, realizovat.

Revize rámcových vzdělávacích programů v oblasti informatiky a ICT probíhají od května 2016 a byly rozděleny do několika dílčích úkolů:

- Začlenit informatická témata do závazného všeobecného kurikula. Cílem není výchova IT profesionálů, je jim rozvoj informatického myšlení a poskytnutí základního vhledu do problematiky dat, informací a jejich zpracování s očekáváním, že určitou část tak to vzdělávaných žáků obor informatika zaujme a budou v jeho studiu dále pokračovat.
- Revidovat všechny vzdělávací obory podle toho, jak vývoj digitálních technologií ovlivnil vývoj v jejich mateřských oborech i v nejrůznějších oblastech lidských činností obecně.
- Navrhnout novou koncepci rozvoje digitální gramotnosti dětí a žáků. Podmínkou udržitelného rozvoje digitální gramotnosti je využívání digitálních technologií žáky při běžné, každodenní školní práci.

Revize byly zahájeny vymezením pojmu digitální gramotnost a pojmu informatické myšlení.

Digitální gramotnost

Digitální gramotnost pojímáme jako soubor digitálních kompetencí, které jedinec potřebuje k bezpečnému, sebejistému, kritickému a tvořivému využívání digitálních technologií při práci, při učení, ve volném čase i při svém zapojení do společenského života.

Digitální kompetence chápeme jako průřezové klíčové kompetence, tj. kompetence, bez kterých není možné rozvíjet u dětí a žáků plnohodnotně další klíčové kompetence [10]. Jejich základní charakteristikou je aplikace – využití digitálních technologií při nejrůznějších činnostech, při řešení nejrůznějších problémů. Z toho plyne i jejich proměnlivost v čase v závislosti na tom, jak se mění způsob a šíře využívání digitálních technologií ve společnosti a v životě člověka.

Informatické myšlení

Informatické myšlení pojímáme jako způsob uvažování, které jedinci umožňuje rozpoznávat informatické aspekty světa a využívat informatických prostředků k porozumění a uvažování o přirozených i umělých systémech a procesech. Informaticky myslící jedinec při řešení nejrůznějších životních situací cílevědomě a systematicky volí a uplatňuje optimální postupy [11], [12].

Dítě, žák se postupně učí rozpoznávat a formulovat problémy s ohledem na jejich řešitelnost, získávat, zaznamenávat, uspořádávat, strukturovat, předávat data a informace, rozkládat systémy a procesy na části, odhalovat jejich vztahy a strukturu, modelovat situace, vytvářet a formulovat postupy a řešení, která lze přenechat k vykonání jinému člověku nebo stroji, vytvářet formální popisy skutečných

DIDINFO 2020, Technická univerzita v Liberci, Fakulta přírodovědně-humanitní a pedagogická
situací a pracovních postupů, testovat, analyzovat, vyhodnocovat, porovnávat a vylepšovat uvažovaná řešení.

Dalším krokem byla konkretizace digitálních a informatických vzdělávacích cílů a kompetencí a rozhodnutí o způsobu, jak je začlenit do rámcových vzdělávacích programů. Byly zpracovány podklady a připraven první návrh revizí RVP v oblasti informatiky a ICT, součástí návrhu je i koncepce rozvoje digitální gramotnosti a informatického myšlení žáků [13].

V roce 2018 byl návrh revizí rámcových vzdělávacích programů v oblasti informatiky a ICT na základě veřejných konzultací upraven a spolu s rámcem očekávaných výstupů pro informatiku a digitální gramotnost pro předškolní, základní a střední vzdělávání slouží jako podklad, ze kterého vychází pokusné ověřování rozvoje informatického myšlení dětí a žáků (vyhlášeno MŠMT v září 2018) a rozvoje digitální gramotnosti dětí a žáků (vyhlášeno MŠMT v únoru 2019). Návrh revizí ICT kurikula byl publikován na stránkách NÚV [14].

3.1 Pojetí rozvoje digitální gramotnosti a informatického myšlení dětí a žáků v RVP a ve škole

Koncepce rozvoje digitální gramotnosti a informatického myšlení zahrnuje všechny etapy vzdělávání, včetně předškolního vzdělávání. S technologiemi se děti setkávají prakticky ve chvíli, kdy se narodí, a jsou s nimi (v domácím prostředí) denně v kontaktu. Neznamená to vždy, že je nadměrně používají nebo zanedbávají „nedigitální“ aktivity, ale návyky a postoje související s používáním technologií si začínají vytvářet už v raném věku [15]. To je důvod, proč začlenit podněty k smysluplné, tvořivé a bezpečné práci s digitálními technologiemi již ve vzdělávání v mateřské škole.

Východiskem pro revize RVP dalších etap vzdělávání byla představa rozvoje digitálních, informatických i ostatních (oborových) kompetencí související s používáním digitálních technologií v systému, který obsáhne celou školní výuku od počátku školní docházky, zahrnuje aktivity žáků ve škole i jejich zkušenosti z aktivit mimo školu.

Úvahy o tom, kdy a jak začleňovat práci s daty, informacemi a digitálními technologiemi do rámcových vzdělávacích programů a do výuky, vyústily do návrhu strukturace vzdělávacího obsahu v RVP a doporučení, kterými směry zaměřit metodickou podporu při zavádění inovací ve výuce informatiky a ICT.

Do rámcových vzdělávacích programů je v návrhu nově začleněn vzdělávací obor informatika s těžištěm v rozvoji informatického myšlení a v informatických tématech, se svým příspěvkem k rozvoji digitálních kompetencí žáků, obdobným jako mají ostatní vzdělávací obory. Vzdělávací obsah je rozčleněn do čtyř tematických okruhů: Data, informace a modelování, Algoritmizace a programování, Informační systémy, Počítač a jeho ovládání (digitální technologie).

Vzdělávací obsah informatiky byl připraven pro časovou dotaci 3 hodiny na 1. stupni, 4 hodiny na 2. stupni ZŠ a 4 hodiny na SŠ (maturitní obory).

3.1.1 Ukázka z návrhu revizí ICT kurikula, Rámcové očekávané výstupy pro informatiku, SŠ (K, M, L0)

Data, informace a modelování

žák

- posuzuje množství informace podle úbytku možností; interpretuje získané výsledky a závěry, vyslovuje předpovědi na základě dat, uvažuje při tom omezení použitých modelů; odhaluje chyby a manipulace v cizích interpretacích a závěrech; odhalí a sám se vyvaruje kognitivních zkreslení*
- rozlišuje a používá různé datové typy; porovná různé způsoby kódování z různých hledisek a vysvětlí proces a úskalí digitalizace, včetně principů fungování bezeztrátové a ztrátové komprese dat*

- *formuluje problém a požadavky na jeho řešení; získává potřebné informace, posuzuje jejich využitelnost a dostatek (úplnost) vzhledem k řešenému problému; používá systémový přístup k řešení problémů; pro řešení problému sestaví model*
- *převede data z jednoho modelu do jiného; najde chyby daného modelu a odstraní je; porovná různé modely s ohledem na užitečnost pro řešení daného problému*

Algoritmizace a programování

žák

- *vysvětlí daný algoritmus, program; určí, zda je daný postup algoritmem*
- *rozdělí problém na menší části, rozhodne, které je vhodné řešit algoritmicky, své rozhodnutí zdůvodní; sestaví a zapíše algoritmy pro řešení problému*
- *zobecní řešení pro širší třídu problémů; ověří správnost, najde a opraví případnou chybu v algoritmu*
- *ve vztahu k charakteru a velikosti vstupu hodnotí nároky algoritmů; algoritmy podle různých hledisek porovná a vybere pro řešený problém ten nejvhodnější; vylepší algoritmus podle zvoleného hlediska*
- *v textovém programovacím jazyce sestaví přehledný program, ten otestuje a optimalizuje*
- *používá opakování, větvení programu se složenými podmínkami, proměnné, seznamy a objekty, podprogramy s parametry a návratovými hodnotami, externí knihovny; ve snaze o vyšší efektivitu navrhuje, řídí a hodnotí souběh procesů*

Informační systémy

- *vysvětlí, co je informační systém a k čemu slouží; analyzuje a hodnotí veřejné informační systémy z hlediska struktury a vzájemné provázanosti; rozpozná informační toky v přirozených systémech*
- *vyhledává data úpravou databázového dotazu*
- *formuluje problém a požadavky na jeho řešení, specifikuje a stanoví požadavky na IS*
- *navrhne procesy zpracování dat a roli/e jednotlivých uživatelů*
- *navrhne a vytvoří strukturu vzájemného propojení tabulek*
- *otestuje svoje řešení IS se skupinou vybraných uživatelů (spolužáků), vyhodnotí výsledek testování, případně navrhne vylepšení, naplánuje kroky k jeho plnému nasazení do provozu, rozpozná chybový stav, zjistí jeho příčinu a navrhne způsob jeho odstranění*

Počítač a jeho ovládání

žák

- *vysvětlí pojem počítač, porovná jednotlivé typy, popíše jejich strukturu a jednotlivé části; vysvětlí, jakým způsobem pracuje počítač s daty*
- *rozumí fungování hardwaru natolik, aby jej mohl efektivně používat a snadno se naučil používat nový*
- *vyjmenuje jednotlivé typy operačních systémů a vysvětlí rozdíly mezi nimi, jak z uživatelského, tak z hlediska vnitřního fungování; popíše, jakým způsobem operační systém zajišťuje své hlavní úkoly*
- *rozumí fungování softwaru natolik, aby jej mohl efektivně používat a snadno se naučil používat novou verzi i nové aplikace*
- *porovná jednotlivé způsoby propojení počítačů, charakterizuje počítačové sítě a internet; vysvětlí, pomocí čeho a jak je komunikace mezi jednotlivými zařízeními v síti zajištěna*
- *rozumí fungování sítí natolik, aby je mohl efektivně používat*
- *identifikuje a řeší technické problémy vznikající při práci s digitálními zařízeními; poradí druhým při řešení typických závad*
- *chrání digitální zařízení, digitální obsah i osobní údaje v digitálním prostředí před poškozením či zneužitím; reaguje na změny v technologiích ovlivňujících bezpečnost*

- *s vědomím souvislosti fyzického a digitálního světa vytváří a spravuje jednu či více digitálních identit; kontroluje svou digitální stopu, ať už ji vytváří sám nebo někdo jiný, v případě potřeby dokáže používat služby internetu anonymně*
- *identifikuje v historii vývoje hardwaru i softwaru zlomové okamžiky; ukáže, které koncepty se nemění a které ano a jak*

Digitální gramotnost je v návrhu na revize RVP popsána jako celek – souhrn digitálních kompetencí, kde prakticky každý vzdělávací obor závazným způsobem přispívá k budování jejich základu a k jejich dalšímu rozvoji dochází aplikací v různých kontextech školní práce.

Digitální kompetence jsou rozděleny do následujících oblastí: Člověk, společnost a digitální technologie, Tvorba digitálního obsahu, Informace, sdílení a komunikace v digitálním světě.

Toto pojetí odpovídá kategoriím RVP klíčová kompetence a průřezové téma, časová dotace specificky pro rozvoj digitálních kompetencí nebyla tedy v návrhu stanovena.

3.1.2 Ukázka z návrhu revizí ICT kurikula, Rámcové očekávané výstupy pro digitální gramotnost, SŠ (K, M, L0)

Člověk, společnost a digitální technologie

žák

- *vyhledává příležitosti k zapojení se do občanského života prostřednictvím vhodných digitálních technologií a služeb; chápe význam digitálních technologií pro sociální začleňování, pro osoby s hendikepem, pro kvalitu života*
- *kriticky posuzuje, jak vývoj technologií včetně umělé inteligence ovlivňuje různé aspekty života jedince a společnosti a životní prostředí; zvažuje příležitosti a rizika, snaží se rizika minimalizovat*
- *běžně a samozřejmě využívá vhodné technologie a jejich kombinace k naplnění svých potřeb; vybavení a způsob jeho použití nastavuje a mění podle toho, jak se vyvíjí dostupné možnosti a jak se mění jeho vlastní potřeby*
- *využívá digitální technologie k vlastnímu vzdělávání a osobnímu rozvoji; buduje si osobní vzdělávací prostředí; rozpozná, kdy je třeba vlastní digitální kompetence zdokonalit nebo aktualizovat; je schopen podpořit ostatní v rozvoji jejich digitálních kompetencí*
- *s vědomím souvislosti fyzického a digitálního světa vytváří a spravuje jednu či více digitálních identit; kontroluje svou digitální stopu, ať už ji vytváří sám nebo někdo jiný*
- *chrání sebe a ostatní před možným nebezpečím v digitálním prostředí; chrání digitální zařízení, digitální obsah i osobní údaje v digitálním prostředí před poškozením či zneužitím; při využívání digitálních služeb posuzuje jejich spolehlivost a postupuje vždy s vědomím existence zásad ochrany osobních údajů a soukromí dané služby*
- *při používání digitálních technologií předchází situacím ohrožujícím tělesné i duševní zdraví, přizpůsobuje své digitální i fyzické pracovní prostředí tak, aby bylo v souladu s ergonomií a bezpečnostními zásadami*
- *zná a uplatňuje právní normy v digitálním prostředí včetně norem týkajících se ochrany citlivých a osobních údajů a duševního vlastnictví*
- *při interakcích v digitálním prostředí respektuje pravidla chování a jedná eticky, respektuje kulturní rozmanitost; s daty získanými prostřednictvím různých nástrojů a služeb, v různém digitálním prostředí pracuje s ohledem na dobrou pověst svou i ostatních*
- *navrhne taková řešení prostřednictvím digitálních technologií, které mu pomohou vylepšit postupy či technologie*
- *dokáže poradit s vyřešením technických problémů*

Tvorba digitálního obsahu

- *vytváří a upravuje digitální obsah v různých formátech, vyjadřuje se za pomoci digitálních prostředků*
- *pozměňuje, vylepšuje a zdokonaluje obsah nebo ho zpracovává do stávajících děl s cílem vytvořit nový, originální a relevantní obsah*

Informace, sdílení a komunikace v digitálním světě

- *získává data, informace a obsah z různých zdrojů v digitálním prostředí; při vyhledávání používá různé strategie; získaná data a informace kriticky hodnotí, posuzuje jejich spolehlivost a úplnost*
- *přizpůsobuje organizaci a uchování dat, informací a obsahu prostředí a účelu*
- *komunikuje prostřednictvím různých digitálních technologií a přizpůsobuje prostředky komunikace danému kontextu*
- *sdílí prostřednictvím digitálních technologií data, informace a obsah s ostatními; používá digitální technologie pro spolupráci a společné vytváření zdrojů a znalostí*

Rozdělení ICT kurikula na informatiku a digitální kompetence naznačuje i jejich začlenění do výuky ve školách. Slovo „naznačuje“ je použito záměrně a je třeba jej chápat s touto výhradou: rámcové vzdělávací programy neurčují, v jakých předmětech bude jimi vymezený vzdělávací obsah realizován. Digitální kompetence je však vhodné rozpracovat ve školním vzdělávacím programu v co nejužší vazbě na činnosti a témata v jednotlivých vyučovacích předmětech (všeobecně vzdělávací i odborné složky vzdělávání). Možnou variantou je i kombinace samostatného vyučovacího předmětu věnovaného rozvoji digitálních kompetencí a integrace do ostatních (vybraných) vyučovacích předmětů podle organizačních, materiálních a personálních podmínek školy. Základem je však vždy poskytnout žákům dostatek příležitostí, situací a kontextů, ve kterých se budou učit orientovat v digitálním prostředí a bezpečně a efektivně využívat různé digitální technologie.

3.2 Začlenění informatiky a rozvoje digitální gramotnosti do RVP

Pokusné ověřování nového (revidovaného) ICT kurikula realizují pedagogické fakulty ve dvou projektech – Podpora rozvoje digitální gramotnosti (DG) a Podpora rozvíjení informatického myšlení (PRIM). Do každého z projektů jsou zapojeny všechny pedagogické fakulty a kromě zpětné vazby k novému kurikulu mají za úkol připravit učební materiály pro žáky, metodiky a vzdělávání pro učitele, návrhy sylabů pro výuku v přípravě učitelů a popularizaci témat digitální gramotnosti a informatického myšlení mezi širokou veřejností.

Pokusné ověřování skončí v červnu 2020. Na základě průběžných výsledků pokusného ověřování se připravuje finální verze revidovaného ICT kurikula.

Průběh revizí ICT kurikula byl (a je) silně ovlivněn úkolem zahájit tzv. komplexní revize RVP, tj. revidovat jak pojetí, tak obsah rámcových vzdělávacích programů pro všeobecné vzdělávání. V roce 2016 bylo stanoveno, že revize vzdělávacích oborů budou probíhat po etapách (informatika a ICT byly zařazeny do první etapy), později (2018) byl úkol změněn na revizi nejprve RVP ZV jako celku, v roce 2019 byly komplexní revize zastaveny zcela.

Paralelně se zpracovávaly revize odborné složky (tzv. aktualizace) RVP SOV. V roce 2018 bylo rozhodnuto k aktualizaci odborné části RVP SOV přidat i revize ICT kurikula ve všeobecné části těchto programů. Revize vycházela ze zpracovaného návrhu [14]. Byla aktualizována klíčová kompetence „Kompetence využívat prostředky informačních a komunikačních technologií a pracovat s informacemi“, jednotně pro všechny skupiny oborů s novým názvem „Digitální kompetence“. Byla aktualizována vzdělávací oblast „Vzdělávání v informačních a komunikačních technologiích“, s odlišnou náročností pro skupiny oborů M, H a E. Obsah aktualizované vzdělávací oblasti je oproti stávajícímu stavu výrazně informatický. Návrh revizí ICT kurikula [14] byl upraven tak, aby střední školy mohly ve stávající časové dotaci vyučovat žáky bez průpravy ze ZŠ, náročnější části byly vypuštěny. Nový název vzdělávací oblasti je „Informatické vzdělávání“. Bylo aktualizováno průřezové téma „Informační a komunikační technologie“, s odlišnou náročností pro skupiny oborů

M, H a E. Obsah průřezového tématu byl rozpracován tak, aby usnadnil začleňování rozvoje digitálních kompetencí do osnov jednotlivých vyučovacích předmětů. Nový název průřezového tématu je „Člověk a digitální svět“.

Aktuálně (leden 2020) se připravuje revize ICT kurikula v základním vzdělávání. Termín zahájení výuky podle takto upraveného RVP ZV je stanoven na září 2021.

V návrhu [14] se předpokládá, že každý vzdělávací obor v RVP bude revidován tak, aby zahrnul explicitně i rozvoj schopnosti pracovat s informacemi a digitálními technologiemi a případně i nová témata, podle toho, jak rozvoj digitálních technologií zasáhl do obsahu, činností a postupů jeho mateřského oboru. Nová koncepce se tedy opírá o revize obsahu všech vzdělávacích oborů, kde očekávanou úroveň digitálních kompetencí určují oborové problémy, které by měl být žák schopen (za pomoci technologií či v digitálním prostředí) řešit.

Práce na komplexních revizích RVP ZV byly obnoveny – s termínem zahájení výuky podle nich v září 2023. Součástí zadání tohoto úkolu je redukce vzdělávacího obsahu v RVP.

4 DISKUZE

Vzdělávací obory projevují vcelku pochopitelnou nechuť zpracovávat dílčí zásahy do současného (ne vždy vyhovujícího) kurikula, navíc souběžně s jeho komplexními revizemi a pod tlakem požadavku na redukci vzdělávacího obsahu. Informatika a ICT naopak v obavách z dalších odkladů, hledá řešení pro rok 2021.

Současně se dokončují úpravy ICT kurikula v RVP SOV, kde od roku 2018 probíhají náročná vyjednávání s asociacemi odborného vzdělávání o finální podobě ICT kurikula v RVP, zejména ve vzdělávací oblasti Informatické vzdělávání. Výhrady asociací a škol odborného vzdělávání lze shrnout do tří okruhů:

1. absence explicitního rozpracování výsledků učení pro práci s kancelářským balíkem (zejména s textovým editorem a tabulkovým kalkulátorem) v Informatickém vzdělávání,
„...V aktualizacích RVP nejsou přímo v rámci vzdělávací oblasti Informatické vzdělávání uvedeny programy kancelářského balíku.“
„...Proč chybí internet, práce s textem a tabulkami?“
„...Největší problém vidíme v absenci výuky textového editoru a tabulkového kalkulátoru v části Informatické vzdělávání. Oba tyto nástroje jsou pro praxi absolventů odborných škol klíčové a žáci ze základních škol s těmito nástroji pracovat příliš neumí. Kolegové v dalších předmětech (především v češtině a matematice) zase nemají dost prostoru a znalostí, aby danou problematiku do potřebné hloubky vyložili.“
„...RVP je v oblasti informatiky posunutá od praktické aplikace použití operačního systému a aplikačního software do teoretické roviny. Úplně vypadl kancelářský software.“
„...V obecných cílech zcela chybí:
 - *naučit žáky pracovat s prostředky ICT,*
 - *pracovat s informacemi,*
 - *naučit žáky na uživatelské úrovni používat operační systém, kancelářský software a pracovat s dalším běžným aplikačním programovým vybavením (včetně specifického programového vybavení používaného v příslušné profesní oblasti).“*
2. příliš vysoká náročnost informatického obsahu pro nematuritní obory a jeho zbytečnost pro netechnické obory,
„39-41-L/01 Autotronik
Informatické vzdělávání:
 - *Návrh algoritmu, návrh programu, optimalizace programu – těžko zvládnutelné pro žáky*
 - *Kódování informací a dat – zdá se mi zbytečné...“**„Pro všechny H obory*
Informatické vzdělávání:

- *Návrh algoritmu, návrh programu, optimalizace programu, příkazy, syntaxe – těžko zvládnutelné pro žáky učebního oboru*
- *Ikonické modely, grafy?*
- *Kódování informací a dat – zdá se mi zbytečné*
- *(...)*
- *Žák by se měl učit ovládat PC jako běžný uživatel, programování je vhodné pro žáky na IT školách...*

„Velmi těžko si dokážeme představit, že žák oboru Truhlář bude schopen algoritmicky myslet při řešení problému, či hodnotit, zda jsou v modelu všechna data potřebná k řešení problému.“

„Informatické vzdělávání připomínají zejména ředitelé, kteří mají na své škole i další neelektrotechnické a neinformatické obory.

– algoritmizace, databáze, síť – nesouvisí se samotným oborem vzdělání. V současnosti musí věnovat čas dovednostem, které by měli mít žáci ze ZŠ (kancelářský SW, práce s grafikou) a také musíme zohlednit jejich odbornost a tudíž vyučovat oblasti, kde je ICT využito jejich oboru vzdělání využito. Pro ekonomy, strojaře, stavaře, ale i řezbáře, stavebníky varhan a další je to úplně mimo...“

„...V obecných cílech nepovažujeme pro uvedený obor vzdělání (Zdravotnické lyceum, Ošetřovatel, pozn, aut.) za důležité:

výuka informatiky přispívá k hlubšímu a komplexnímu porozumění počítači a principům, na kterých počítač funguje.“

„Uvedené dovednosti (navrhovat taková (bezpečná) řešení prostřednictvím digitálních technologií, která jim pomohou vylepšit postupy či technologie; dokázat druhým poradit s vyřešením technických problémů, týká se průřezového tématu, pozn. aut.) odpovídají oborům vzdělání zaměřeným na informační technologie a technicky zaměřeným oborům vzdělání, nikoli oboru vzdělání Sociální činnost.“

3. nedostatečné kompetence učitelů neICT předmětů pro výuku zahrnující práci s digitálními technologiemi.

„Kompetence pedagogů, jejichž aprobace není informatika, nejsou dostatečné na to, aby zajistili plnohodnotnou výuku aplikací a zároveň řešili technické problémy související s provozem vybavení.“

Výhrady k (ne)návaznosti na RVP ZV do výčtu nebyly zahrnuty, protože se bezprostředně netýkají obsahu informatiky či koncepce rozvoje digitálních kompetencí.

V průběhu jednání se projevilo malé porozumění novému (informatickému) obsahu i nechuť měnit zavedené pořádky a postupy. Zatímco první, jak se při debatách ukázalo, je většinou poměrně dobře řešitelné vysvětlováním a ukázkami, které jsou k dispozici díky připravovaným učebním materiálům a učebnicím, druhé nejspíš povede ke kompromisu a návratu o krok zpět, blíž k původnímu obsahu.

Předmětem příspěvku není rozebírat a hodnotit kroky vzdělávací politiky, MŠMT či kurikulárního ústavu, ani situace, které k nim vedly. Zmíněny jsou proto, že poskytují kontext probíhající akademické debaty nad obsahem školské informatiky, digitálních kompetencí a jejich začleněním do závazného kurikula.

Přetrvávajícím problémem revizí ICT kurikula je stanovení obsahu školské informatiky a jejího vztahu k rozvoji digitálních kompetencí.

V prvním návrhu revizí i v jeho pozdějších úpravách bylo nutné se vyrovnat s pojetím digitálních kompetencí, které se v dílčích konceptech a kompetencích překrývá s vymezením informatického myšlení (práce s daty, programování) [16], tendencí oddělovat pojmy základní digitální dovednosti (basic digital skills) a digitální kompetence [17] i vymezováním digitální (ICT) gramotnosti jako schopnosti používat digitální technologie (ICT) na „uživatelské úrovni“ a informatického myšlení mj. i jako cesty k „autorskému přístupu“ k počítači a digitálním technologiím [9].

Vymezování vzdělávacího obsahu informatiky a jeho formulace v RVP prochází vývojem. Počáteční rozhodování na úrovni „uživatelské dovednosti“ (nepatří do informatiky), „autorský přístup“ (patří do informatiky) postupně překrývají složitější úvahy na úrovni ontodidaktické transformace [18] (vědního) oboru informatika do vzdělávacího oboru informatika a zvažování toho, co je příspěvkem (školské) informatiky k rozvoji digitálních kompetencí.

Úvahy, diskuze i veřejné konzultace na toto téma jsou také zatíženy převažující praxí ve školách a vnímáním učitelů toho, co je (školská) informatika. Ve vyučovacím předmětu s názvem informatika je nyní realizován stávající obsah RVP (zaměřený v základním vzdělávání výhradně, ve středním převážně na rozvoj schopnosti pracovat s ICT a informacemi na uživatelské úrovni jako základ pro aplikaci v ostatních vzdělávacích oblastech) [7], [8].

Postupně a v obecné rovině stále méně lidí zpochybňuje význam (školské) informatiky a smysl jejího začlenění do RVP. Tváří v tvář konkrétním řešením, jak bylo naznačeno, není snadné změny obhájit a prosadit.

5 ZÁVĚR

Byl zpracován návrh revizí rámcových vzdělávacích programů v oblasti informatiky a ICT [14]. Digitální gramotnost je v návrhu na revize RVP popsána jako celek – souhrn digitálních kompetencí, kde prakticky každý vzdělávací obor závazným způsobem přispívá k budování jejich základu a k jejich dalšímu rozvoji dochází aplikací v různých kontextech školní práce. Do rámcových vzdělávacích programů je v návrhu nově začleněn vzdělávací obor informatika s těžištěm v rozvoji informatického myšlení a v informatických tématech, se svým příspěvkem k rozvoji digitálních kompetencí žáků, obdobným jako mají ostatní vzdělávací obory.

Článek popisuje průběh a výsledky probíhající revize ICT kurikula a formuluje vybrané problémy, na které revize narážejí.

Po stránce odborné se při vývoji nového závazného ICT kurikula jako klíčové ukázalo vymezení vztahu vznikající školské informatiky ke konceptu digitálních kompetencí, tj. schopnosti (způsobilosti) zacházet s digitálními technologiemi odpovídající aktuálnímu vývoji ve společnosti a na trhu práce [16]. Vývoj nového závazného kurikula má ale i složku politickou, protože změny je nutné prosadit a realizovat. Zde se rovněž ukazuje jako klíčové porozumění obsahu (nové) školské informatiky, tomu, co nového do kurikula přináší, proč je nový obsah pro všechny žáky ve všeobecném vzdělávání důležitý. Tato potřeba vysvětlit a obhájit nový obsah zasahuje i do odborného diskurzu, používané terminologie, vymezování základních konceptů i jejich vztahu. Pojetí digitálních kompetencí zahrnující jak „uživatelský“, tak i „autorský“ přístup k technologiím může sloužit jako jeden z argumentů pro začlenění informatických témat do závazného kurikula všeobecného vzdělávání.

Vývoj závazného kurikula je práce kurikulárního ústavu a jeho nejbližším úkolem v oblasti informatiky a ICT je zpracovat v konkrétní podobě rozvoj digitálních kompetencí do RVP ZV a obhájit jeho finální verzi ve veřejné diskuzi. Navazující úkol je zpracovat nový obsah v komplexních revizích a vytvořit základ pro rozvoj digitálních a informatických kompetencí kontinuálně od předškolního po střední vzdělávání.

Vývoj závazného kurikula ale nelze realizovat bez zapojení dalších aktérů, je potřeba pokračovat ve zkoumání školské informatiky a rozvoji její oborové didaktiky, zkoumat digitální kompetence dětí, žáků a učitelů, prosadit jejich rozvoj v oborových didaktikách, zahájit výuku v duchu uspokojivě revidovaných RVP a monitorovat a analyzovat průběh realizace nového vzdělávacího obsahu.

6 BIBLIOGRAFICKÉ ODKAZY

- [1] NEUMAJER, Ondřej. Proč a jak inovovat pojetí ICT v rámcových vzdělávacích programech. *Metodický portál: Články* [online]. 14. 02. 2009, [vid. 12. 1. 2020]. Dostupný na: <https://clanky.rvp.cz/clanek/c/Z/2989/PROC-A-JAK-INOVOVAT-POJETI-ICT-V-RAMCOVYCH-VZDELAVACICH-PROGRAMECH.html>. ISSN 1802-4785.

- [2] TUPÝ, Jan. *Tvorba kurikulárních dokumentů v České republice: historicko-analytický pohled na přípravu kurikulárních dokumentů pro základní vzdělávání v letech 1989–2017*. 2., doplněné a upravené vydání. Brno: Masarykova univerzita, 2018, 230 s. Pedagogický výzkum v teorii a praxi, svazek 46. ISBN 978-80-210-8997-6.
- [3] *Strategie digitálního vzdělávání do roku 2020*. (2014). Praha: MŠMT.
- [4] *Rámcový vzdělávací program pro základní vzdělávání*. (2017). Praha: MŠMT.
- [5] *Rámcový vzdělávací program pro gymnázia*. (2016). Praha: MŠMT.
- [6] *Rámcový vzdělávací program pro obor vzdělání Ekologie a životní prostředí*. (2008). MŠMT.
- [7] RAMBOUSEK, Vladimír., et al. *Rozvoj informačně technologických kompetencí na základních školách*. Praha: České vysoké učení technické v Praze, 2013. ISBN 978-80-01-05407-9.
- [8] ČŠI. *Rozvoj informační gramotnosti na základních a středních školách v roce 2016/17*, tematická zpráva [online]. 2017. [vid. 12. 1. 2020]. Dostupné na: <https://www.csicr.cz/cz/Dokumenty/Tematicke-zpravy/Tematicka-zprava-Rozvoj-informacni-gramotnosti-na>.
- [9] VANÍČEK, Jiří, ČERNOCHOVÁ, Miroslava. Didaktika informatiky na startu. In Stuchlíková, I., Janík, T., et al. *Oborové didaktiky: vývoj – stav – perspektivy*. Brno: Munipress, 2015. ISBN 978-80-210-7769-0. Str. 161–164.
- [10] FERRARI, Anusca. *DIGCOMP: A Framework for Developing and Understanding Digital Competence in Europe* [online]. EUR 26035 EN, 2013, str. 2, doi:10.2788/52966. [vid. 12. 1. 2020]. Dostupné na: <https://publications.jrc.ec.europa.eu/repository/handle/JRC83167>.
- [11] LESSNER, Daniel. Analýza významu pojmu „computational thinking“. In: *Journal of Technology and Information Education* [online], 6 (1), Olomouc 2014, pp. 71–88. [vid. 12. 1. 2020]. Dostupné na: <http://www.jtie.upol.cz/pdfs/jti/2014/01/06.pdf>.
- [12] FURBER, Steve. *Shut down or restart? The way forward for computing in UK schools* [online]. London: The Royal Society, 2012. [vid. 12. 1. 2020]. Dostupné na: <https://royalsociety.org/~media/education/computing-in-schools/2012-01-12-computing-in-schools.pdf>.
- [13] NÚV. *Přehled podkladů k revizím ICT kurikula* [online]. 2019. [vid. 12. 1. 2020]. Dostupné na: <http://www.nuv.cz/file/4208/>.
- [14] NÚV. *Návrh revizí ICT kurikula* [online]. 2018. [vid. 12. 1. 2020]. Dostupné na: <http://www.nuv.cz/t/revize-rvp-ict>.
- [15] CHAUDRON, Stephane. *Young children (0–8) and digital technology: A qualitative exploratory study across seven countries* [online]. 2015. JRC 93239/EUR 27052. ISBN 978-92-79-45023-5 ISSN 1831-9424 [vid. 12. 1. 2020]. Dostupné na: <https://publications.jrc.ec.europa.eu/repository/handle/JRC93239>.
- [16] CARRETERO, Stephanie; VUORIKARI, Riina and PUNIE, Yves. *DigComp 2.1: The Digital Competence Framework for Citizens with eight proficiency levels and examples of use* [online]. 2017. EUR 28558 EN, 978-92-79-68006-9 (pdf) 978-92-79-68005-2 (print) 978-92-79-74173-9 (ePub) [vid. 12. 1. 2020]. Dostupné na: <https://publications.jrc.ec.europa.eu/repository/handle/JRC106281>.
- [17] EU. *Doporučení Rady ze dne 22. května 2018 o klíčových kompetencích pro celoživotní učení (2018/C 189/01)* [online]. Úřední věstník Evropské unie 2018 [vid. 12. 1. 2020]. Dostupné na: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.C_.2018.189.01.0001.01.ENG&toc=OJ:C:2018:189:TOC.
- [18] JANÍK, Tomáš, SLAVÍK, Jan. Vztah obor a vyučovací předmět jako metodologický problém. *Orbis Scholae*. 2007, roč. 2, č. 1, s. 54–66.

Nový standard studia ICT koordinátora/metodika

A New Study Standard for ICT Coordinator/Methodology Specialist

Ondřej Neumajer
Učitel naživo, z. ú.
Boleslavova 250/1
140 00 Praha 4-Nusle
Česká republika

a
Katedra informačních technologií a technické výchovy, Pedagogická fakulta, Univerzita Karlova
Magdalény Rettigové 4
116 39 Praha 1
Česká republika
ondrej@neumajer.cz

ABSTRAKT

Již patnáctým rokem může na každé české škole působit ICT metodik, někdy také nazývaný ICT koordinátor a pomáhat vedení školy a ostatním učitelům s integrací digitálních technologií do života školy. Článek shrnuje historii této specializované činnosti, vysvětluje jisté pojmové nejasnosti kolem názvu této funkce, překryv s obdobnou profesní kvalifikací v Národní soustavě kvalifikací, mapuje několik historických pokusů o aktualizaci standardu tohoto studia a především představuje nový návrh standardu studia, který v roce 2019 vznikl na základě požadavku Ministerstva školství, mládeže a tělovýchovy (MŠMT). V článku jsou popsány důvody pro inovace standardu a hlavní rozdíly nového konceptu oproti stávajícímu znění.

ABSTRACT

For the 15th year, every Czech school can establish the ICT methodologists, sometimes also called the ICT coordinator, and help school management and other teachers integrate digital technologies into school life. The paper summarises the history of this specialized activity, explains some conceptual ambiguities surrounding the title of this role, overlaps with similar professional qualifications in the National Qualifications Framework, maps several historical attempts to update the standard of this study, and, above all, presents a new draft of the standard of study, which was established in 2019 under the requirement of The Ministry of Education, Youth and Sports. The paper describes the reasons for the innovation of the standard and the main differences between the new concept and the current wording.

Klíčová slova

ICT koordinátor, ICT metodik, standard, studium, digitální technologie

Keywords

ICT coordinator, ICT methodologist, standard, study, digital technology

1 ÚVOD

Standard studia specializované činnosti koordinace v oblasti informačních a komunikačních technologií (ICT) je důležitý nejen pro vzdělávací organizace, které toto studium pedagogickým pracovníkům nabízejí, ale také proto, že obsah pracovní náplně ICT koordinátora/metodika není v žádném legislativním dokumentu blíže popsán. V praxi školy pracovní náplň učitele pověřeného touto činností často odvíjejí právě od tohoto standardu. V mnoha školách je ale tato pozice využívána

k jiným účelům, nejčastěji pro správu školní počítačové sítě. To je samozřejmě v rozporu s metodicko-pedagogickým charakterem této specializované činnosti, kterou má vykonávat kvalifikovaný pedagogický pracovník. Těžko za to ale ředitele škol kritizovat, když například poslední tematická zpráva České školní inspekce (ČŠI) věnovaná digitálním technologiím z roku 2017 konstatuje, že situace ve správě digitální infrastruktury škol je kritická: „*Mnoho škol řeší pravidelnou a běžnou správu digitálních technologií svépomocí (často ředitel školy, koordinátor/metodik ICT nebo učitel ICT)...*“ [1]. Vlastního správce ICT zaměstnává pouze 5,2 % malých základních škol (ZŠ), 15,8 % velkých ZŠ (tj. nad 150 žáků) a 31,6 % středních škol (SŠ) a vyšších odborných škol (VOŠ).

Jisté pojmové zmatení školního *ICT metodika* a *ICT koordinátora*, jak je tato funkce pojmenována v různých legislativních dokumentech, měl eliminovat článek zveřejněný na Metodickém portálu RVP.CZ v roce 2010 [2]. Ten konstatuje, že **ICT koordinátor i ICT metodik je různé pojmenování jedné role** a dále popisuje možnou sníženou míru přímé vyučovací povinnosti a nárok na příplatek pro učitele, který tuto funkci zastává a studium absolvoval. Uvedený článek se svými 36 tisíci zhlédnutí patří k nejčtenějším za celou historii Metodického portálu RVP.CZ.

Sjednocení terminologie v této oblasti si bere za cíl Strategie digitálního vzdělávání ČR do roku 2020 ve svém opatření 6.3.1. Pojem *koordinátor ICT* by měl být zaveden změnou terminologie původně uvedené v Nařízení vlády č. 75/2005 Sb., tato změna by měla být realizována v průběhu roku 2020.

Citovaná zpráva ČŠI zjistila, že ICT koordinátoři/metodici působí v 86,6 % velkých základních školách, v menších školách je to logicky méně, jen 46,6 %. Kvalifikovaných absolventů tohoto studia bylo v době šetření na malých ZŠ pouze 29,3 % oproti 56,9 % ve velkých ZŠ. Uvedená data doplňuje zjištění Nejvyššího kontrolního úřadu z šetření podpory rozvoje digitalizace vzdělávání v roce 2019, kdy mezi nejčastější důvody, proč v dané škole není pozice ICT koordinátora/metodika zřízena, uvedlo 39 % škol nedostatek finančních prostředků a 32,6 % přesvědčení, že tato funkce není na škole zapotřebí [3].

2 DŮVODY REVIZE STANDARDU

Standard studia k výkonu specializované činnosti koordinace v oblasti informačních a komunikačních technologií, jak zní dle vyhlášky č. 317/2005 Sb. oficiální název, vytvořilo MŠMT v době realizace Státní informační politiky ve vzdělávání známé spíše pod označením *Internet do škol* a je poplatný době svého vzniku. Digitální technologie se za poslední půl druhé dekády změnily a dnes mnohem intenzivněji zasahují do našich životů. Ne jinak je tomu i ve škole.

Standard studia z roku 2005 byl vymezen prostřednictvím klíčových cílových kompetencí a doporučených tematických okruhů [4]. Téměř všechny vzdělávací instituce, které se rozhodly si studium u MŠMT v rámci systému Dalšího vzdělávání pedagogických pracovníků (DVPP) akreditovat a učitelům následně nabízet, přistoupily k praxi tím způsobem, že svůj program strukturovaly přesně podle tematických okruhů uvedených jako doporučené. Těch bylo ve standardu celkem 16, např. *Hygiena, ergonomie a pravidla bezpečnosti práce s ICT*, *Tvorba ICT plánu školy*, *Interakce ICT a RVP* nebo *Principy a možnosti počítačových sítí*.

3 PŘEDCHOZÍ POKUSY O REVIZI STANDARDU

Volání a i pokusy o změnu standardu studia zaznělo v minulosti hned několik, žádná ale nebyla tak úspěšná, aby sama o sobě vedla k jeho revizi. Například na jaře roku 2013, kdy vzniká na MŠMT nová *Strategie vzdělávací politiky do roku 2020* [5] a současně probíhá celonárodní kampaň *Česko mluví o vzdělávání* se na 10. ročníku učitelské konference *Počítač ve škole* pořádané Gymnáziem Vincence Makovského v Novém Městě na Moravě sešli učitelé nad doporučením, co by bylo vhodné v oblasti digitálních technologií ve vzdělávání měnit. Jedna část tohoto doporučení se věnuje právě studiu ICT metodiků/koordinátorů. Výsledný návrh přichází například s **požadavkem zařadit do standardu studia k výkonu této činnosti téma tzv. „měkkých“ dovedností, konkrétně dovedností**

vést úspěšná jednání s pedagogy (kolegy), vedením školy (nadřízenými) a firmami (partnery) nebo téma věnované aktuálním novinkám z oblasti moderních vzdělávacích technologií. Tato doporučení byla předána vedení MŠMT, aby je ministerstvo mohlo zohlednit při vytváření v dlouhodobější koncepci. To se pak skutečně stalo, když se v roce 2014 do nově vytvořené a vládou schválené Strategie digitálního vzdělávání do roku 2020 dostal i požadavek na revizi standardu studia specializované činnosti *Koordinace v oblasti ICT* (opatření 6.3.1).

V **Národní soustavě kvalifikací vznikal** od roku 2013 kvalifikační a hodnotící standard pro profesní kvalifikaci *Koordinátor v oblasti ICT* (kód: 75-011), který MŠMT jakožto autorizující orgán schválilo v listopadu 2016 [6]. Ačkoli se tato profesní kvalifikace NSK obsahově téměř překrývá s platným standardem studia dle vyhlášky č. 317/2005 Sb., nejednalo se o specializovanou činnost dle této vyhlášky a Zákona o pedagogických pracovnících 563/2004 Sb. Tím pádem úspěšný absolvent zkoušky v systému NSK nezíská osvědčení, které by bylo uznáváno jako plnohodnotná varianta studia dle vyhlášky.

Ze stejného opatření Strategie digitálního vzdělávání vycházel i Národní institut pro další vzdělávání (NIDV), který na konci roku 2016 dokončil vlastní návrh standardu studia a počítal v něm, že jej promítne do tehdy vznikajícího Kariérního systému. Obsahově vycházel tento návrh do značné míry z původní (a stále platné) verze z roku 2005, okruhy studia v něm byly ale rozpracovány do větších podrobností. Nejen z důvodů nepřijetí Kariérního systému ale nebyl ve finále využit a ani se nedostal ke schvalování na MŠMT.

4 VZNIK NÁVRHU NOVÉHO STANDARDU

Koncem roku 2018 vytvořil NIDV pracovní skupinu, která dostala za úkol navrhnout v souladu s citovaným opatřením Strategie digitálního vzdělávání nový standard. Součástí této odborné skupiny byl zástupce základní školy, střední školy, vysoké školy vzdělávající učitele, zástupce NIDV i MŠMT. Pracovní skupina vycházela z popsaných snah o změnu standardu, dále z různých závěrečných akademických prací, které se tématu věnovaly, např. [7] a [8], zahraničních zkušeností (viz dále). Postupné návrhy hlavních oblastí, kam standard směřovat, byly předmětem jednání **ICT panelu** Národního ústavu pro vzdělávání v prosinci 2018 a dále konzultovány se zástupci odborné veřejnosti na **workshopu konference Počítač ve škole** v dubnu 2019 [9] vedeném participativní metodou World Café. K návrhu se také vyjadřovali nově zřízení **krajští ICT metodici** (tzv. KIM) projektu Systém podpory profesního rozvoje učitelů a ředitelů (SYPO) realizovaném NIDV. Všechny formulované a zaznamenané připomínky byly pracovní skupinou postupně diskutovány a případně po zvážení zapracovány.

Z výše uvedeného textu je patrné, že pracovní skupina chtěla založit nové směřování standardu na zkušenostech široké skupiny expertů i praktikujících aktérů ve školách při respektování moderních pedagogických trendů a zohledňování obdobných aktivit v zahraničí.

5 OBSAH NÁVRHU NOVÉHO STANDARDU

Jednou z důležitých charakteristik standardů studia specializovaných činností je skutečnost, že **standard není přímo určen učitelům, ale vzdělávacím institucím**, které v souladu s ním mohou žádat o akreditaci vlastního vzdělávacího programu v systému Dalšího vzdělávání pedagogických pracovníků (DVPP). Učitelům tedy budou k dispozici různé vzdělávací programy směřující k naplňování stejných vzdělávacích cílů, které se vzájemně budou jistě lišit, a je na každém zájemci, pro který z nich se rozhodne.

Obsah pracovní náplně ICT koordinátora/metodika není v žádném legislativním dokumentu blíže popsán, takže **standard má značný formativní charakter** na pracovní činnosti ICT koordinátorů/metodiků ve školách.

Délka studia je minimálně **250 vyučovacích hodin** po dobu nejméně jednoho roku. Nejméně 110 vyučovacích hodin probíhá jako prezenční výuka (včetně dvou čtyřhodinových exkurzí ve škole

DIDINFO 2020, Technická univerzita v Liberci, Fakulta přírodovědně-humanitní a pedagogická
či školském zařízení a osmihodinové stáže formou stínování apod.) a nejméně 30 a nejvýše 140 hodin
distanční formou (e-learning, webináře, online vzdělávání). Studium je **ukončeno závěrečnou ústní
zkouškou před komisí a obhajobou závěrečné práce.**

Do studia by měli být přijímáni kvalifikovaní pedagogičtí pracovníci, kteří vládou digitálními
kompetencemi minimálně na úrovni B1 dle Rámce digitálních kompetencí pedagogů (DigCompEdu)
[10]. To je úroveň, která je v českém jazyce označena jako *Praktik*, kromě potřebných znalostí
vyžaduje i určité zkušenosti z práce ve škole.

Celé studium je vymezeno profilem absolventa, který je popsán čtyřmi kompetencemi:

- K1 – kompetence pro vedení lidí k zavádění digitálních technologií do života celé školy
a k jejich efektivnímu používání pro výuku a učení žáků,
- K2 – kompetence mentorské, koučovací a lektorské – andragogické techniky,
- K3 – kompetence koordinační a manažerské – řízení procesů a projektů,
- K4 – kompetence k digitálním technologiím (technické a technologické).

U každé kompetence jsou dále podrobněji rozvedeny tyto tři složky:

- předpokládané výsledky učení,
- učivo, resp. obsah výuky,
- indikátory dosažených výsledků učení.

Předpokládané kompetence a výsledky učení jsou rozpracovány následovně:

K1 – Kompetence pro vedení lidí k zavádění digitálních technologií do života celé školy a k jejich efektivnímu používání pro výuku a učení žáků

Předpokládané výsledky učení:

- a) facilituje proces tvorby digitální strategie školy;
- b) zajišťuje pravidelnou reflexi a vyhodnocení naplňování digitální strategie školy;
- c) koordinuje a aktivně podporuje další vzdělávání pedagogických pracovníků s využitím
- d) digitálních technologií;
- e) podporuje kulturu inovací a spolupráce ve škole;
- f) působí jako pedagogický lídr.

K2 – Kompetence mentorské, koučovací a lektorské – andragogické techniky

Předpokládané výsledky učení:

- a) uplatňuje prvky mentorských a koučovacích dovedností a technik v prostředí školy a ve své
práci;
- b) využívá pro reflexi praxe kompetenční rámec kvalitní pedagogické praxe se zaměřením
na využití digitálních technologií;
- c) vzdělává se v lektorských dovednostech a andragogice tak, aby mohl pomáhat dospělým učit
se a volit jejich cestu celoživotního vzdělávání;
- d) orientuje se v různých metodách vzdělávání s ohledem na nové vědecké poznatky a aktuální
technologický vývoj, dokáže je dalším osobám erudovaně doporučit a podpořit je v jejich užití
pro jejich další osobní i profesní růst;
- e) podílí se na tvorbě a sestavení plánu seberochovy i autoevaluace pedagogů, na profesním
portfoliu a sebemotivaci;
- f) vzdělává se v oblasti vedení lidí a pedagogického leadershipu.

K3 – Kompetence koordinační a manažerské – řízení procesů a projektů

Předpokládané výsledky učení:

- a) organizuje a koordinuje využívání digitálních zařízení ve škole;
- b) spoluvytváří směrnice a pravidla pro bezpečné používání digitálních technologií ve škole
včetně kyberbezpečnosti;
- c) vyhledává a navrhuje možné finanční zdroje pro pořízení a údržbu digitální infrastruktury
a softwaru;

- d) spoluvytváří plán rozvoje školy v oblasti digitální strategie školy;
- e) integruje digitální technologie do ŠVP;
- f) navrhuje a koordinuje provoz informačního systému školy;
- g) orientuje se v základních pravidlech projektového řízení.

K4 – Kompetence k digitálním technologiím (technické a technologické)

Předpokládané výsledky učení:

- a) kvalifikovaně doporučuje dle individuálních potřeb školy školní informační systémy;
- a) orientuje se v oblasti moderních digitálních technologií používaných ve školách a ve vzdělávání a aktivně sleduje trendy ve vzdělávání s využitím digitálních technologií;
- b) orientuje se v problematice budování, údržby, správy a zabezpečení počítačové sítě;
- c) navrhuje postupy a pravidla zabezpečení digitální infrastruktury včetně dat;
- d) orientuje se v problematice autorských práv a licencí
 - a. včetně veřejných licencí (Creative Commons);
 - b. včetně otevřeného softwaru (Open Source);
- e) podílí se na nastavování licenční politiky školy.

Výše uvedená struktura standardu (z důvodů rozsahu zde neuvádíme učivo, resp. obsah výuky a indikátory dosažených výsledků učení) je společná všem novým standardům, které MŠMT aktuálně plánuje zveřejnit. Kompletní schválená verze standardu by měla být dostupná na webu MŠMT v sekci Vzdělávání/Další vzdělávání/Vzdělávání pedagogů/Akreditace v systému DVPP.

6 KLÍČOVÉ CHARAKTERISTIKY

Za klíčovou charakteristiku návrhu standardu lze označit akcent na **zaměření se na práci s lidmi, jejich spolupráci a pedagogické vedení**. Pedagogické vedení je v moderní literatuře označováno pojmem **pedagogický leadership** (*school leadership*) a znamená péči o dlouhodobé směřování – úsilí o to, aby škola dělala správné věci [11] resp. způsob zvyšování kvality vyučování [12]. V případě ICT koordinátora/metodika by bylo například žádoucí, aby sám používal postupy využívání digitálních technologií, které jsou ostatním učitelům příkladem, a přispíval ke zkvalitňování personálních vztahů ve škole. Takto vytvářené klima je předpokladem k rychlejšímu šíření pedagogických inovací.

Snahou je, aby ICT koordinátor/metodik dovedl také vhodně využívat prvky **lektorských, mentorských a koučovacích technik**. Neočekává se, že tyto andragogické techniky zcela ovládne a že z něj bude profesionální lektor, mentor a kouč dohromady. Zkušenosti posledních let ale ukazují, že integrace digitálních technologií do života školy je podmíněna **především postoji a hodnotami, které jednotliví aktéři vzdělávání zastávají a kterým věří** [13] a [14]. Proto je vhodné kandidáty na pozici ICT koordinátora/metodika v oblasti práce s lidmi, v jejich motivování a podporování dále rozvíjet. Uvedené techniky podporují také schopnost přesvědčovat a vyjednávat, kterou ICT koordinátor/metodik využije i při jednání s vedením školy a školními dodavateli digitálních technologií.

Důležitou součástí navrhovaného pojetí studia je **síťování účastníků, vznik a posilování profesních komunit odborníků**, kteří si vzájemně (za podpory digitálních technologií) pomáhají a podporují se. To vše za účelem poskytování kvalitní metodické podpory v integraci digitálních technologií do života školy ostatním pedagogickým pracovníkům. S tímto záměrem jsou nově do studia zařazeny i exkurze a stáže. Výstižně popisuje potřebu posilování dovedností ČŠI [15], když shrnuje zjištění, že „v práci ICT koordinátorů byly zaznamenány významné příležitosti k lepší podpoře učitelů v oblasti využití digitálních technologií pro didaktické účely“.

Uvedený akcent na (pedagogický) **leadership, práci s vizemi a spolupráci lidí** je pozorovatelný také v zahraničních vzdělávacích systémech [8]. Například etablovaná organizace International Society for Technology in Education (ISTE) v poslední verzi svých standardů [16] považuje za hlavní

charakteristiky následující: agent změny, propojený student, spolupracovník, průvodce učením, podporovatel učení, rozhodování se na základě dat, obhájce digitálního občanství.

Samotné technické a technologické kompetence musí být i nadále součástí výbavy dobrého školního ICT koordinátora/metodika. Jejich aktuálnost je ale často vázána k době, ve které studium probíhá, časem zastarávají, resp. jsou překonány novými digitálními trendy a vývojem v této oblasti. Očekává se, že potřebné novinky z oboru si bude ICT koordinátor/metodik permanentně **průběžně prohlubovat samostudiem i v rámci DVPP** (semináře, konference, webináře, e-learning, MOOC, online komunity...), jednorázové absolvování specializovaného studia nemůže průběžné vzdělávání v žádném případě nahradit. Může jej však navést na vzdělávací zdroje a příležitosti (např. digitální stopu čelních učitelů), které bude v dalším vzdělávání využívat.

Oproti standardu z roku 2005 klade nový standard menší důraz na technické a technologické aspekty samotných digitálních technologií. Důvodem jsou i **změny ve způsobech provozování školní digitální infrastruktury** a stále se zvyšující **kvalita uživatelských rozhraní**. Učitelé dnes mají oproti situaci před 15 lety k dispozici více výkonných nástrojů, které se ovládají jednodušeji a uživatelsky přívětivěji. Příkladem mohou být hojně využívané cloudové služby jako je G Suite či Office 365, jejichž používání minimalizuje nároky na technické náležitosti jejich provozování na školním hardwaru (jsou provozovány na hardwaru poskytovatele) i uživatelské dovednosti pro jejich správu a užívání (mnohá nastavení mohou jednoduše měnit sami uživatelé v prostředí, které se snaží být co nejvíce intuitivní). Analýza dat získaných ČŠI [3] ukazuje, že cloudové služby využívá v ČR již 26 % ZŠ, SŠ a VOŠ. V mnoha případech jsou také učitelé schopni vykonávat jednodušší pedagogicko-administrativní práce sami (např. správa žáků v systémech řízení výuky či nastavování práv sdílení dokumentů) a tak tuto agendu nemusí řešit specialista, který se může více soustředit na metodické otázky.

7 SHRNUŤÍ

Popisovaný návrh standardu studia specializované činnosti koordinace v oblasti ICT je v době psaní tohoto článku posuzován MŠMT. Pokud bude přijat, bude se jednat o první změnu obsahu tohoto studia od jeho vzniku v roce 2005. Za hlavní obsahový záměr lze považovat větší akcent v činnosti ICT koordinátora/metodika směrem k metodicko-pedagogickým činnostem integrace digitálních technologií do života školy, přímou práci s kolegy učiteli a k podpoře jejich spolupráce.

8 BIBLIOGRAFICKÉ ODKAZY

- [1] Tematická zpráva ČŠI. *Využívání digitálních technologií v mateřských, základních, středních a vyšších odborných školách*. Praha: Česká školní inspekce, 2017 [vid. 25. 5. 2020]. Dostupné na <http://www.csicr.cz/cz/Dokumenty/Tematicke-zpravy/Tematicka-zprava-Vyuzivani-digitalnich-technologii>.
- [2] ÚLOVEC, Roman. *ICT metodik, ICT koordinátor*. Metodický portál RVP.CZ [online]. 10. 3. 2010 [vid. 25. 5. 2020]. Dostupné na <https://clanky.rvp.cz/clanek/c/Z/8013/ICT-METODIK-ICT-KOORDINATOR.html>. ISSN 1802-4785.
- [3] Kontrolní závěr z kontrolní akce. *Datová příloha ke kontrolnímu závěru 18/18 – Podpora rozvoje digitalizace vzdělávání v ČR* [online]. Praha: Nejvyšší kontrolní úřad, 12. 8. 2019. [vid. 25. 5. 2020]. Dostupné na <https://www.nku.cz/scripts/detail.php?id=10616>.
- [4] *Standardy pro udělování akreditací DVPP. Standardy Studia k výkonu specializovaných činností* [online]. Praha: MŠMT, 2005 [vid. 25. 5. 2020]. Dostupné na <http://www.msmt.cz/vzdelavani/dalsi-vzdelavani/standardy-a-metodicka-doporuceni>.
- [5] *Strategie digitálního vzdělávání do roku 2020* [online]. Praha: MŠMT, 2014. Dostupné na <http://sdv.msmt.cz/>.

- [6] Národní soustava kvalifikací, profesní kvalifikace *Koordinátor v oblasti informačních a komunikačních technologií, kód: 75-011-T* [online]. Národní soustava kvalifikací. NÚV a TREXIMA. 29. 11. 2016 [vid. 25. 5. 2020]. Dostupné na https://www.narodnikvalifikace.cz/kvalifikace-696-Koordinator_v_oblasti_informacnich_a_komunikacnich_technologii.
- [7] ČEMUSOVÁ, Lucie. *Funkce ICT koordinátora na základní škole*. Praha: PedF UK, 2014. Bakalářská práce.
- [8] BLAŽEK, Jiří. *Školní metodik ICT v mezinárodním kontextu*. Praha: PedF UK, 2016. Diplomová práce. Dostupné na <https://is.cuni.cz/webapps/zzp/detail/150279/>.
- [9] NEUMAJER, Ondřej. *Workshop: Návrh nového standardu studia ICT metodiků/koordinátorů* [online]. Konference Počítač ve škole, Nové Město na Moravě: 17. 4. 2019 [vid. 25. 5. 2020]. Dostupné na <https://www.pocitacveskole.cz/rocnik/2019/prispevek/navrh-noveho-standardu-studia-ict-metodikukoordinatoru>.
- [10] NEUMAJER, Ondřej a Daniela RŮŽIČKOVÁ, Bořivoj BRDIČKA. *Evropský rámec digitálních kompetencí pedagogů DigCompEdu* [online]. Metodický portál RVP.CZ. 24. 9. 2018 [vid. 25. 5. 2020]. Dostupný z <https://spomocnik.rvp.cz/clanek/21855/EVROPSKY-RAMEC-DIGITALNICH-KOMPETENCI-PEDAGOGU-DIGCOMPEDU.html>. ISSN 1802-4785.
- [11] POL, Milan. *Škola v proměnách*. Brno: Masarykova univerzita, 2007.
- [12] VESELÝ, Arnošt. *Vzdělávací reforma v Ontariu: kritické shrnutí a inspirace pro vzdělávací politiku v ČR*. Orbis scholae, vol. 7, str. 11–28, 2013. ISSN 1802-4637.
- [13] GREENFIELDOVÁ, Susan. *Změna myšlení. Jak se mění naše mozky pod vlivem digitálních technologií*. Brno: Bizbooks, 2016. ISBN 978-80-265-0450-4.
- [14] ZOUNEK, Jiří a Libor JUHAŇÁK, Hana STAUDKOVÁ a Jiří POLÁČEK. *E-learning: učení (se) s digitálními technologiemi*. Praha: Wolters Kluwer, 2016. ISBN 978-80-7552-217-7.
- [15] Informační bulletin ČŠI [online]. Praha: ČŠI, 18. 12. 2019 [vid. 25. 5. 2020]. Dostupné na <https://www.csicr.cz/cz/Aktuality/Informacni-bulletin-Ceske-skolni-inspekce-12-2019>
- [16] *Standards for Coaches* [online]. ISTE, 2019. Dostupné na <https://www.iste.org/standards/for-coaches>.

Reálné využívání digitálních technologií v mateřských školách

Real Use of Digital Technologies in Preschool Education

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

ABSTRAKT

Cílem této práce je popsat současnou vybavenost a způsob práce s digitálními technologiemi v mateřských školách. Naše práce se dále zaměřuje na názory učitelů na využívání digitálních technologií v mateřských školách. V celém textu se snažíme uvádět příklady dobré praxe práce s digitálními technologiemi. Práce byla zpracována pomocí kvalitativní analýzy seminárních prací studentů učitelství pro mateřské školy na více jak čtyřiceti mateřských škol převážně z Jihočeského kraje.

Z analýzy vyplývá nechuť nemalé části učitelů mateřských škol integrovat digitální technologie do edukačního procesu v mateřských školách. Tento problém může být překážkou pro úspěšné zavedení připravované revize rámcových vzdělávacích programů v oblasti informatiky a informačních a komunikačních technologií v mateřských školách.

ABSTRACT

The aim of this work is to describe current equipment and way of working with digital technologies in kindergartens. Our work also focuses on teachers' opinions on the use of digital technologies in kindergartens. Throughout the text, we try to give examples of good practice in working with digital technologies. We qualitative analyzed seminar works of teachers for kindergartens from more than forty kindergartens mainly from South Bohemia.

The analysis shows considerable reluctance part of the nursery teachers to integrate digital technology into the educational process in the kindergartens. This problem can be an obstacle to the successful implementation of the upcoming revision of the informatics and information and communication technologies curriculum in kindergartens.

Klíčová slova

Mateřské školy, digitální technologie, reforma výuky informatiky, informatické myšlení

Keywords

Kindergartens, digital technology, reform of computer science education, computational thinking

1 ÚVOD

Již několik let je v NÚV (nyní NPI) připravována revize rámcových vzdělávacích programů v oblasti informatiky a informačních a komunikačních technologií [1]. Připravovaná revize předpokládá zavedení informatických témat i do předškolního vzdělávání. Domníváme se, že pro úspěšné zavedení reformy je důležité přesvědčení učitelů o důležitosti a vhodnosti zavedení daného tématu

do výuky. Pokud učitelé nebudou přesvědčeni, pravděpodobně provedou reformu pouze po formální stránce.

Při výuce předmětu digitálních technologií jsme se nejednou setkali s názorem učitelů mateřských škol z praxe: „U nás ve školce nemáme ani na čtvrtky, natožpak na digitální technologie.“ Nedostatečná vybavenost mateřských škol digitálními technologiemi může být dalším faktorem potenciálně ovlivňujícím průběh zavádění revize rámcových vzdělávacích programů v oblasti informatiky a informačních a komunikačních technologií. Cílem této práce je popsat současnou vybavenost a způsob práce s digitálními technologiemi v mateřských školách. Naše práce se dále zaměřuje na názory učitelů na využívání digitálních technologií v mateřských školách a na chystanou reformu informatiky a informačních a komunikačních technologií. Domníváme se, že díky popsání reality a identifikaci potenciálních problémů pomůže tato práce k lepší integraci připravované revize informatiky a informačních a komunikačních technologií.

2 METODOLOGIE

Výzkum byl realizován analýzou seminárních prací distančních i prezenčních studentů 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 dále zjišťovali názory učitelů v konkrétní mateřské škole na využívání digitálních technologií v mateřské škole. V této oblasti se ukázaly jako velmi přínosné především práce distančních studentů, kteří většinou popisovali mateřskou školu, ve které zároveň pracují. Velmi dobře tak znali popisované prostředí.

Celkem zpracovalo seminární práce 31 distančních studentů a 21 prezenčních studentů. Dohromady tedy bylo zpracováno 52 seminárních prací. V některých případech více seminárních prací popisovalo stejnou mateřskou školu. Celkem bylo do výzkumu zahrnuto 42 různých mateřských škol. Většina mateřských škol zúčastněných na výzkumu byla z Jihočeského kraje. Rozdělení mateřských škol podle kraje je uvedeno v tabulce 1. Seminární práce byly zpracovávány od listopadu 2017 do listopadu 2019.

Tabulka 1: Rozdělení mateřských škol podle krajů

Kraj	Počet
Jihočeský kraj	32
Kraj Vysočina	5
Středočeský kraj	1
Praha	1
Jihomoravský kraj	2
Západočeský kraj	2
Celkem mateřských škol	42

Podle rejstříku škol MŠMT [2] je v Jihočeském kraji celkem 328 zařízení věnujících se předškolnímu vzdělávání, naše práce tedy obsáhla 10 % všech mateřských škol Jihočeského kraje. Celkem bylo do výzkumu zahrnuto 14 mateřských škol z vesnic do 1 000 obyvatel, 14 mateřských škol s malých měst velikosti od 1 000 do 10 000 obyvatel a 14 mateřských škol z velkých měst o velikosti větší než 10 000 obyvatel. Samotné seminární práce byly zpracovávány kvalitativně-quantitativní formou. Kvalitativní část byla realizována v programu Atlas.ti [3]. Pro získání přehledu nad daty byla data kódována otevřeným kódováním podle Strauss a Corbinové [4]. Data kvantitativního charakteru byla zaznamenávána do tabulkového procesoru. Následně byly z těchto dat tvořeny kontingenční tabulky. Mateřské školy byly grupovány podle velikosti obce, ve které fungují a podle způsobu fungování na klasické a alternativní.

3 VÝSLEDKY

V mateřských školách se využívají rozdílné digitální technologie. Nejdříve popíšeme jednotlivé typy technologií v mateřských školách a způsob práce s nimi. Následně zdokumentujeme názory učitelů mateřských škol na využívání digitální techniky a povědomí učitelů o plánované reformě výuky informatiky. Dále se zaměříme na způsob přijímání digitálních technologií v „alternativních“ mateřských školách. Výsledná kontingenční tabulka: *Rozložení digitální techniky v mateřských školách podle velikosti obcí* je uvedena v závěru.

3.1 Interaktivní tabule

Interaktivní tabule je nejčastější digitální zařízení zastoupené v mateřských školách. Interaktivní tabuli vlastnila asi polovina mateřských škol. Nejčastěji je interaktivní tabule využívána pro různé předpřipravené výukové programy. Tyto programy jsou většinou placené jako například Barevné kamínky. Málokdy si paní učitelky samy tvoří aktivity v programech k tomu určených, jako je například smartnotebook apod. Předškoláci jedné mateřské školy pak pravidelně docházeli ke své budoucí paní učitelce v první třídě základní školy. Zde měla jejich budoucí paní učitelka připravenou výuku pomocí interaktivní tabule.

Interaktivní tabule jsou poměrně často využívány k pouhému promítání obsahu. Děti podle tabulí cvičí či koukají na pohádky. Bohužel v některých mateřských školách jsou interaktivní tabule využívány pouze k tomuto účelu. V jiných mateřských školách se pak interaktivní tabule nevyužívají vůbec anebo minimálně. Celkem třetina interaktivních tabulí instalovaných v mateřských školách byla využívána pouze jako projektor anebo nebyla vůbec využívána. Někteří, převážně starší, učitelé pak odmítli instalaci interaktivních tabulí do svých tříd. Jako hlavní příčiny tohoto stavu jsou zmiňovány nedostatečná schopnost ovládání interaktivních tabulí a s ní související nedostatek školení věnujících se digitálním technologiím v mateřských školách.

Alternativou k interaktivní tabuli je interaktivní koberec, či interaktivní stůl smart table. Tyto technologie měly ale pouze minoritní zastoupení.

3.2 Počítače

V současné době je většina mateřských škol vybavena alespoň jedním počítačem běžně připojeným k internetu. Počítače jsou hlavně využívány pro nepřímou pedagogickou činnost, především pro přípravu výuky, jako je stahování a tvorba rozdílných výukových materiálů a jejich tisk, či dohledávání informací. Učitelé dále v několika případech využívali počítač pro emailovou komunikaci s rodiči, či jiné administrativní úkony. Zde již ale v některých případech narážejí učitelé na limity své digitální gramotnosti jako například: „Jediná možnost, jak lze omluvit dítě, je přes email, což pro mnohé učitelky bylo zpočátku nepřijatelné, ale v průběhu tří let každá z nich umí přijmout a odpovědět na email.“

Pokud byly počítače využívány v průběhu přímé pedagogické činnosti, šlo především o pouštění hudby, případně byly počítače propojeny s dataprojektory, či televizí a byly na nich pouštěny pohádky, případně cvičení. V jednom případě předškoláci pracovali s výukovými aplikacemi přímo na počítači.

Celkem pět mateřských škol vůbec nebylo vybaveno žádným počítačem. Všechny tyto mateřské školy jsou situovány ve městě. Ve třech alternativních mateřských školách využívají paní učitelky mobilní telefony připojené na wifi. Zbýlé dvě běžné mateřské školy nevyužívají digitální technologie, nejsou připojeny na internet a pravděpodobně to ani neplánují.

3.3 Tablety a mobilní telefony

Celkem v pěti mateřských školách mají k dispozici více tabletů, které využívají k přímé činnosti s dětmi. V tabletech jsou nainstalovány didaktické programy, které děti mohou využívat většinou při řízených činnostech. Čas strávený na tabletech je regulován na dobu od 10 do 30 minut. Velmi zajímavé řešení zvolili v Prachaticích. Bylo zde zakoupeno celkem 8 tabletů. Tablety si mezi sebou

půjčují všechny mateřské školy v Prachaticích. Pravděpodobně vzhledem k nutnosti přepravy tabletů k nim mají přístup pouze předškoláci. Využívají je jednou za měsíc na 30 minut při řízené činnosti.

V několika dalších mateřských školách pak paní učitelky využívají mobilní telefon k jiným účelům než k telefonování. Takovéto využívání mobilních telefonů je ale sporadické. Používají mobilní telefon jako zdroj hudby, hledají obrázky, či odpovědi na zvědavé otázky dětí. Jedna paní učitelka pak: „při výletech a procházkách v přírodě využívá fotoaparát a kameru nebo např. nahrává zvuky zvířat, a pak ve školce s dětmi poznávají, jaké zvíře to bylo”.

3.4 Robotické hračky určené k rozvoji informatického myšlení

Celkem čtyři mateřské školy (10 %) byly vybaveny robotickými včelkami bee bot. Všechny mateřské školy vlastní robotické včelky jsou vesnické anebo maloměstské mateřské školy. Velmi pravděpodobně jsou včelky ve všech mateřských školách využívány pouze k volné hře. Nedochází zde k žádnému záměrnému rozvoji informatického myšlení anebo algoritmizace pomocí robotických včelek, tak, jak to popisují například Maněnová a Pekárková [5]. Reakce učitelek na včelky jsou různé. Od pozitivních: „Děti tuto interaktivní podložku naprosto zbožňují a u této činnosti většinou vydrží sedět opravdu dlouho” a „Tyto včelky mají také děti velice rády, hlavně, když včelky do sebe nabourají.” Až po negativní „Hrají si s ní zřídka, učitelka upřednostňuje jiné činnosti.” a „Zná robotické hračky, nemyslí si, že je důležité je v MŠ mít.” Ze všech výše uvedených citací je patrné, že citování učitelé nedokáží náležitě využít didaktického potenciálu robotických včelek. Kromě robotických včelek jsme nezaznamenali žádné další technologie, či aktivity vedoucí k rozvoji informatického myšlení dětí.

3.5 Audio technika

Běžně je každá třída vybavena audio technikou primárně určenou k přehrávání CD. Nejčastěji se využívají ke cvičení, dramatizaci, relaxaci, či poslechu mluveného slova. Ve dvou mateřských školách využívali mikrofon, na který nahrávali různé zvuky, které pak poznávali a nebo nahrávali zpěv dětí. Zajímavým příkladem využití audio techniky byla besídka v jedné mateřské škole:

„Z každé třídy měly děti nacvičenou nějakou pohádku či příběh a pomocí digitálních technologií paní učitelky předcházely velké části komplikací. Délka textů, komplikovanost děje i předpoklad určitého stresu byly vyřešeny pomocí nahrávacího zařízení, kdy celé pohádky byly předem s některými dětmi nahrány, namixovány s hudbou či zvuky a časově rozvrženy a nazkoušeny se všemi dětmi. Výsledkem bylo, že při samotné besídce děti mluvily současně s nahrávkou. Některé děti svůj text odříkaly bez chyby a nahrávku by k tomu snad ani nepotřebovaly, jiné kousky textu zapomínaly, ale s pomocí nahrávky si vzpomněly, a ti nejmenší mnohdy jen předstírali mluvení. Díky nahrávkám se ale mohli divadel zúčastnit plnohodnotně všichni.”

3.6 Fotografická technika

Přibližně třetina mateřských škol uvedla, že využívají digitální fotoaparát. Většinou pomocí fotoaparátů paní učitelky zaznamenávají různé aktivity dětí. Výsledné fotografie pak často umísťují na web tištěných zpravodajů, či se vytištěné dávají na nástěnky. Zajímavým využitím fotoaparátu je: „Fotografujeme např. 1x týdně starou seschlou slunečnici – čekáme, co s ní čas a příroda udělá.”

Pouze čtyři mateřské školy uvedly, že nechávají děti samostatně fotit. Děti většinou fotí přírodniny, jako květiny, kameny, či jiné zajímavé přírodní úkazy. V jedné vesnické mateřské škole paní učitelka tablet využívá jako fotoaparát. „Děti si fotku vyfotí a následně s fotografií pracují. Například si fotku vytisknou a následně zachycený okamžik ztvárňují při výtvarné činnosti nebo udělají výřez z fotografie, zkoumají detaily či fotku upravují. Děti se seznamují s tím, že fotku si mohou v tabletu uložit a například za delší časový úsek si mohou udělat rekapitulaci zážitků, které nafotily.”

3.7 Ostatní digitální technika

Zaznamenali jsme několik dalších digitálních technologií využívaných v mateřských školách. Všechny níže uvedené technologie byly využívány maximálně ve dvou mateřských školách. Převážně

při volných hrách byla využívána Albi tužka. S dětskou kamerou si děti hrály na kameramany a natáčely pěvecké vystoupení. Dále byly využívány Recordable pegs „Tyto skřípečky slouží k nahrávání úkolů, pokynů. Paní učitelka má děti rozdělené po stolečcích do skupin a na každý nahraje jeden úkol. Ty poté plní. Dá se to zpětně přehrávat, nebo vymazat a zapsat jiný pokyn.“

3.8 Názory učitelů na využívání digitálních technologií v mateřské škole

Celkem vyslovilo svůj názor na využívání digitálních technologií asi 60 učitelů mateřských škol. Z nich přibližně polovina nesouhlasila s využíváním digitálních technologií při přímé pedagogické činnosti. Šlo spíše o starší učitele. Jednou z příčin jejich odmítnutí bylo jejich nízké mínění o své digitální gramotnosti a „často ostych a obava z neúspěchu“. Příkladem může být citace: „Negativní reakce zazněly od učitelek, které mají za sebou dlouholetou praxi – důvody jejich nesouhlasu byly převážně kvůli neznalosti běžné manipulace s PC a také, že mají doma těchto „vymožeností“ nad hlavu.“

Dalším důvodem odmítnutí digitálních technologií je názor, že děti mají na digitální technologie ještě dostatek času, nyní je třeba se věnovat rozvoji motoriky, komunikačních schopností a praktickým činnostem, navíc děti v domácím prostředí velmi často až příliš využívají digitálních technologií.

Mezi další důvody odmítnutí digitálních technologií dále patří finanční náročnost, strach z poškození cenného vybavení a poruchovost technologií. Jako například: „ne vždy fungují, často se něco porouchá, interaktivní tabule nepracuje, jak má, nebo signál připojení na internet je pomalý nebo nestabilní a často dochází k výpadkům. To je potom ke vzteku.“

V některých případech ale byly investovány nemalé peníze do vybavení digitální technikou, která následně byla vinou nízké digitální gramotnosti učitelů minimálně využívána: „Za svůj krátký pobyt na postu ředitelky vybavila každou třídu notebookem a projektorem. Jeden výhradně pro práci ředitelky a tři byly učitelkám k ruce, ty je ale využívaly jen svátečně, protože je neměly ve svých třídách a ani je nenapadlo, že by se s takovou technikou dalo pracovat ve výuce.“

Druhá polovina učitelů se již digitálním technologiím nebrání, často ale zdůrazňují, že jejich využívání nesmí být na úkor ostatních aktivit. Z těchto ale většina využívá předpřipravené aktivity. Pouze minimum učitelů vytváří vlastní aktivity, anebo kreativně využívá digitální technologie.

3.9 Názory učitelů mateřských škol na reformu výuky informatiky

Názory na chystanou reformu výuky informatiky se nám podařilo sehnat asi u poloviny mateřských škol. Žádná z učitelek o chystané reformě informatiky nevěděla.

Při dotazu na jejich názor na chystanou reformu informatiky, která se bude týkat i mateřských škol, si díky své neznalosti reformy učitelky představovaly povinné zavádění digitálních technologií do mateřských škol a jejich následné analogické využívání k současné výuce předmětu informatika na základních školách. Vůči takové představě se téměř všechny učitelky vymezily, čímž odmítly chystanou reformu výuky informatiky. „Zeptala jsem se, jak by se zachovaly v případě, že by opravdu došlo k zavedení výuky informatiky v MŠ. Názor zněl takto: „Vždy přijde někdo s nějakou reformou, která se nakonec neuskuteční nebo neosvědčí. Doufáme, že tomu tak bude i v tomto případě nebo, že informatika do MŠ bude zavedena v době, kdy my už tady nebudeme a budeme v důchodu.“ Děti se zkrátka nejvíc naučí, když se budou učit mezi sebou, v přímém kontaktu a nebudou „rušeny“ digitálními technologiemi, s kterými se setkávají doma a na které narazí ve vyšším věku.“

Mnohé z učitelek v souvislosti s reformou dále vyjádřily obavu nad svými schopnostmi práce s digitálními technologiemi.

3.10 Vztah alternativních mateřských škol k digitálním technologiím

Do výzkumu bylo zahrnuto celkem pět alternativních mateřských škol. Tři z nich se hlásily k Montessori pedagogice, po jedné k Waldorfské pedagogice a k pedagogice Franze Ketta. Všechny tyto mateřské školy se shodovaly v odmítnutí digitálních technologií v přímé pedagogické činnosti. Upozorňují na různé problémy spojené s využíváním, jako je nedostatečný rozvoj motoriky,

jazykových schopností dětí, časté využívání technologií v domácnostech a tak podobně „Děti v tomto věku potřebují zažít situace co nejvíce smysly, tedy osahat si předměty, o kterých mluvíme.“ Velmi často tak preferují především praktické reálné činnosti.

Specifická byla Waldorfská mateřská škola. „Děti zde za celou dobu pobytu v mateřské škole nepřijdou do kontaktu s počítačem ani ničím jiným souvisejícím s digitálními technologiemi, protože to je v rozporu s waldorfskou pedagogikou jako takovou.“ „Nenajdeme zde jedinou plastovou hračku.“

4 ZÁVĚR

Před započítáním jsme očekávali, že nejlépe budou vybaveny digitálními technologiemi městské mateřské školy. Naše očekávání se ale nepotvrdilo. Mezi městskými mateřskými školami, je mnoho škol odmítajících využívání digitálních technologií při přímé pedagogické činnosti. Naproti tomu robotickými včelkami byly vybaveny dvě vesnické mateřské školy z vesnic do 1 000 obyvatel a dvě maloměstské mateřské školy z měst do 10 000 obyvatel. Konkrétní výsledky jsou uvedeny v kontingenční tabulce 2.

Tabulka 2 Rozložení digitální techniky v mateřských školách podle velikosti obcí

	Počet	Interaktivní tabule	Beebot	PC	Mobil	Fotoaparát	Tablet	TV
Vesnice	14	8	2	14	4	5	2	5
Malé město	14	8	2	13	2	6	4	6
Město	14	7	0	10	3	1	3	4
Celkový součet	42	23	4	37	9	12	9	15

Domníváme se proto, že v současné době o vybavenosti digitálními technologiemi rozhoduje především přesvědčení učitelů a ředitelů mateřských škol o smysluplnosti využití digitálních technologií v mateřských školách.

„Myslím si, že kdyby se sem zavedla nějaká digitální technologie např.: interaktivní tabule nebo digitální hračky, tak by to bylo neproduktivní. Byla by to zbytečná investice, která děti nebude rozvíjet, protože ji pedagogové nebudou používat ve výuce.“ Učitelé musí být přesvědčeni o smysluplnosti využívání digitálních technologií v mateřských školách, stejně tak i o smysluplnosti zavedení informatických témat do výuky v mateřských školách. Pokud tomu tak nebude, veškeré změny budou provedeny pouze formálně, reálné změny ale provedeny nebudou.

Příkladem takového jednání je vesnická škola s prvky Montessori pedagogiky: „Paní ředitelka psala písemnou zprávu školní inspekci o fungování této mateřské školy. Školce bylo vytknuto, že se dítě v tomto prostředí nesetkává s digitálními technologiemi. Doporučili, aby se dítě seznamovalo s těmito technologiemi aspoň hodinu týdně. S tímto názorem pedagogický sbor nesouhlasí. Paní ředitelka odpověděla, že bere na sebe zodpovědnost za to, že se v této mateřské školce nebudou děti seznamovat s digitálními technologiemi.“

Další důležitou překážkou k využívání digitálních technologií v mateřských školách je digitální gramotnost mnohých učitelů, ze které následně pramení obava z využívání digitální techniky a především neznalost, či neschopnost využití digitální techniky produktivním, či kreativním způsobem. Díky tomuto omezení pak chápou digitální techniku pouze jako nástroj určený k zábavě, což podporuje jejich přesvědčení o nevhodnosti, či nedůležitosti digitálních technologií v mateřských školách. Domníváme se, že k dalšímu rozšíření digitálních technologií v mateřských školách je především důležitá digitální gramotnost učitelů, z tohoto důvodu je proto vhodné hledat informační kanály, které budou učitelům představovat možné způsoby využívání digitálních technologií.

K úspěšnému zavedení plánované revize výuky informatiky je velmi důležité informovat učitele o jejím skutečném obsahu, který zatím neznají a přesvědčit je o jeho smysluplnosti.

5 BIBLIOGRAFICKÉ ODKAZY

- [1] NUV. *Návrh revizí rámcových vzdělávacích programů v oblasti informatiky a informačních a komunikačních technologií* [online]. [vid. 23. 11. 2019]. Dostupné na: <http://www.nuv.cz/file/3362/>.
- [2] *Rejstřík škol a školských zařízení. MŠMT* [online]. [vid. 27. 11. 2019]. Dostupné na: <https://rejstriky.msmt.cz/rejskol/>.
- [3] FRIESE, Susanne. *Qualitative data analysis with ATLAS.ti*. 1st ed. London: SAGE, 2012, ISBN 978-085-7021-304.
- [4] STRAUSS, A. a J. CORBIN. *Základy kvalitativního výzkumu: Postupy a techniky metody zakotvené teorie*. Boskovice: Albert, 1999, ISBN 80-858-3460-X.
- [5] MANĚNOVÁ, M. a S. PEKÁRKOVÁ. *Algoritmizace s využitím robotických hraček pro děti do věku 8 let* [online]. [vid. 21. 11. 2019]. České Budějovice: PF JČU, 2019. Dostupné na: <https://www.imysleni.cz/ucebnice/rozvoj-informatickeho-mysleni-s-vyuzitim-robotickych-hracek-v-materske-skole-a-na-1-stupni-zs>.

Prínos informatiky a informačných technológií k výučbe prírodovedných predmetov

The Contribution of Informatics and Information Technologies to Science Teaching

Gabriela Andrejková
Ústav informatiky,
Prírodovedecká fakulta UPJŠ
Jesenná 5
04001 Košice
Slovensko

gabriela.andrejkova@upjs.sk

Zuzana Ješková
Marián Kíreš
Ústav fyzikálnych vied,
Prírodovedecká fakulta UPJŠ
Park Angelinum 9
04001 Košice
Slovensko

zuzana.jeskova@upjs.sk
marian.kires@upjs.sk

Mária Ganajová
Mária Kožurková
Ústav chemických vied,
Prírodovedecká fakulta UPJŠ
Moyzesova 11
04001 Košice
Slovensko

maria.ganajova@upjs.sk
maria.kozurkova@upjs.sk

ABSTRAKT

Projekt „IT Akadémia – vzdelávanie pre 21. storočie“ má medzi novovytvorenými predmetmi zaradený predmet „Informatika v prírodných vedách a matematike“, ktorý má motivovať žiakov stredných škôl ku štúdiu prírodných vied, matematiky a informatiky. Autorský kolektív pripravil 6 zošitov (Zošit Biológia, Fyzika, Geografia, Chémia, Informatika a Matematika) s vybranými témami vhodnými a odporúčanými pre výučbu predmetu. V témach je zdôraznené prepojenie prírodovedných predmetov s informatikou a informačnými technológiami, čo môže byť významným prínosom pre ich výučbu. V článku detailnejšie opisujeme pripravené metodické a pracovné listy ponúkaného predmetu v témach z fyziky a chémie.

Vo fyzike sme sa zamerali na informatický prínos v oblasti modelovania, zobrazovacích metód a databázových systémov. V oblasti modelovania predstavujeme sériu metodických materiálov zameraných na aktívne žiacke bádanie zamerané na riešenie problémov pomocou počítačového modelovania. Žiaci vytvárajú dynamické modely fyzikálnych javov, ktoré programujú využívajúc ikonografické modelovanie a následne modely porovnávajú s výsledkom reálneho experimentu. V časti zobrazovacie metódy ozrejmujeme proces ultrazvukového snímání a následného zobrazovania, približujeme problematiku termografie a spracovania záznamu pohybujúcich sa objektov. Pri práci s databázami hľadáme argumentmi podložené odpovede na aktuálne problémy informačnej spoločnosti.

V chémii sme sa v časti modelovanie zamerali na modelovanie štruktúry organických zlúčenín, reakčných schém a kreslenie chemických aparátov v programe ChemSketch. Žiaci sa zoznámia s rôznymi spôsobmi znázorňovania štruktúr molekúl, získajú predstavy o väzbových dĺžkach a uhloch, naučia sa pracovať s hotovými databázami vzorcov aromatických zlúčenín, aminokyselín, vitamínov, sacharidov, DNA/RNA bázu atď. Cieľom metodiky „Modelovanie biosyntézy nukleových kyselín a bielkovín, genetický kód“ je naučiť žiakov využiť programovací jazyk Python na riešenie zápisov sekvencií aminokyselín, translácie a transkripcie. V ďalších metodikách sa naučia žiaci vyhľadávať periodické vlastnosti prvkov v databázach, využívať počítačom podporované merania na stanovenie vybraných ukazovateľov kvality vody (teplota, pH, vodivosť, obsah chloridových a dusičnanových aniónov) spracovať dáta do tabuliek a grafov, interpretovať a analyzovať dáta získané meraním.

ABSTRACT

The project “IT Academy - Education for the 21st Century” includes the subject “Informatics in Science and Mathematics”, which should motivate secondary school students to study science,

mathematics and informatics. The author's team prepared 6 workbooks (Biology, Physics, Geography, Chemistry, Informatics and Mathematics) with selected topics suitable and recommended for teaching the subject. The themes emphasize the interconnection with informatics and information technology, which can be an important contribution in the teaching of them. In the paper, we describe in detail prepared methodological and work sheets of the offered subject in topics from physics and chemistry.

In physics, we focused on the IT contribution in modeling, imaging methods and database systems. In the field of modeling, we present a series of methodological materials focused on active student research focused on solving problems using computer modeling. Students create dynamic models of physical phenomena, which they program using iconographic modeling and then compare the models with the results of a real experiment. In the part of imaging methods we clarify the process of ultrasonic scanning and subsequent imaging, we approach the problems of thermography and processing of recording of moving objects. When working with databases, we are looking for arguments based on current problems of the information society.

In chemistry, we focused on modeling the structure of organic compounds, reaction schemes and chemical apparatus drawing in ChemSketch. Students will learn about different ways of displaying the structures of molecules, get ideas about binding lengths and angles, learn to work with ready-made databases of aromatic compounds, amino acids, vitamins, carbohydrates, DNA / RNA base, etc. The aim of the methodology "Modeling of Biosynthesis of Nucleic Acids and Proteins, Genetic Code" is to teach pupils to use the Python programming language to deal with amino acid sequences, translation and transcription. In other methodologies, pupils will learn to search for periodic properties of elements in databases, use computer-assisted measurements to determine selected water quality indicators (temperature, pH, conductivity, chloride and nitrate anion content), process data into tables and graphs, interpret and analyze data obtained by measurement.

Klíčové slová

Informatika, prírodné vedy, matematika, modelovanie, práca s obrázkami, databázové systémy

Keywords

Informatics, natural science, mathematics, modeling, image processing, database systems

1 ÚVOD

Predmet Informatika v prírodných vedách a matematike (IvPVaM), ktorý bol vytvorený ako motivačný predmet v projekte IT Akadémia, bol prvýkrát koncepčne predstavený na DIDINFO 2018 [1]. Na DIDINFO 2019 [2] sme predstavili kontinuálne vzdelávanie zamerané na prípravu učiteľov vyučovať tento predmet. Odvtedy predmet prešiel recenziami a nasledujúcimi odporúčanými úpravami. Zjednodušili sme pracovné listy z informatiky (napríklad, žiaci dopĺňujú vopred pripravené kostry programov, používajú vopred vytvorené funkcie).

Vo fyzike sme pripravili metodické listy s možnosťou využiť na modelovanie prostredie COACH alebo programovať v Pythone (pre šikovnejších žiakov). Tak majú žiaci možnosť vidieť modely v prostredí COACH a potom sami programovať a experimentovať s vlastným programom.

V chémii využívame softvér ChemSketch, ktorý umožňuje písanie vzorcov, chemických reakcií a kreslenie chemických aparátúr, žiaci sa naučia zobrazovať štruktúru molekúl rôznymi spôsobmi.

Práca s existujúcimi databázami ukáže žiakovi možnosti ich využitia pri analýze dostupných dát.

2 FYZIKA

Časť fyzika je zameraná na informatický prínos v oblasti modelovania, zobrazovacích metód a databázových systémov a zahŕňa spolu 10 aktivít.

2.1 Modelovanie

Modelovanie, resp. tvorba teórií je popri experimentovaní neoddeliteľnou súčasťou práce vedca. Správanie sa skúmaného objektu alebo javu vedci modelujú pomocou matematických rovníc, ktorých riešenie často vyúsťuje do náročných výpočtov realizovaných prostredníctvom počítačového programu. Výsledky počítačového modelu sa potom interpretujú a porovnávajú s výsledkom experimentu. Naším cieľom je žiakom predstaviť takýto vedecký prístup k štúdiu javov okolo nás a význam informatiky a programovania pri riešení problémov ale na jednoduchých žiakom blízkych situáciách.

Pri tvorbe modelov využívame metódu dynamického modelovania, ktorá umožňuje aj na úrovni strednej školy riešiť komplexnejšie problémy. Vzhľadom na matematickú náročnosť žiaci model programujú využívajúc ikonografické modelovanie, kedy sú premenné modelu a vzťahy medzi nimi reprezentované ikonami. Toto chápeme ak prvý krok k programovaniu v profesionálnom programovacom prostredí [3]. Na tvorbu modelov ako aj experimentovanie využívame komplexný počítačový systém COACH (<https://cma-science.nl/>) [4]. Žiaci postupujú od jednoduchších ku komplexnejším modelom, ktoré sa postupne rozširujú tak, aby porozumeli významu premenných, ktoré v modeli vystupujú. V každom prípade žiaci porovnávajú výsledky získané z modelovania s reálnymi dátami alebo s výsledkami experimentu. Časť fyzika obsahuje spolu päť aktivít (tabuľka 1). Pre tých, ktorí nemajú systém COACH, sme pripravili kostry programov v Pythone, pomocou ktorých žiaci môžu experimentovať s modelmi. V nasledujúcich častiach uvidíme ukážky dvoch aktivít.

Tabuľka 1: Zoznam žiackych aktivít zameraných na modelovanie

Téma aktivity	Stručný obsah
Úvod do modelovania	Od jednoduchého modelu napúšťania nádrže k modelu pohybu rovnomerného a rovnomerne zrýchleného pohybu
Ako sila ovplyvní pohyb	Od modelovania pádu lopty vo vákuu a vo vzduchu k modelu pádu parašutistu
Ako Felix dosiahol nadzvukovú rýchlosť	Od modelovania pádu parašutistu k modelu pádu zo stratosférického balóna
Ako štartuje raketa	Od modelu pohybu telesa, na ktoré pôsobí sila k modelu pohybu telesa s premennou hmotnosťou
Ako udržať optimálnu teplotu v dome	Od modelu pohybu k modelu teploty v dome ovplyvnenej rozličnými podmienkami

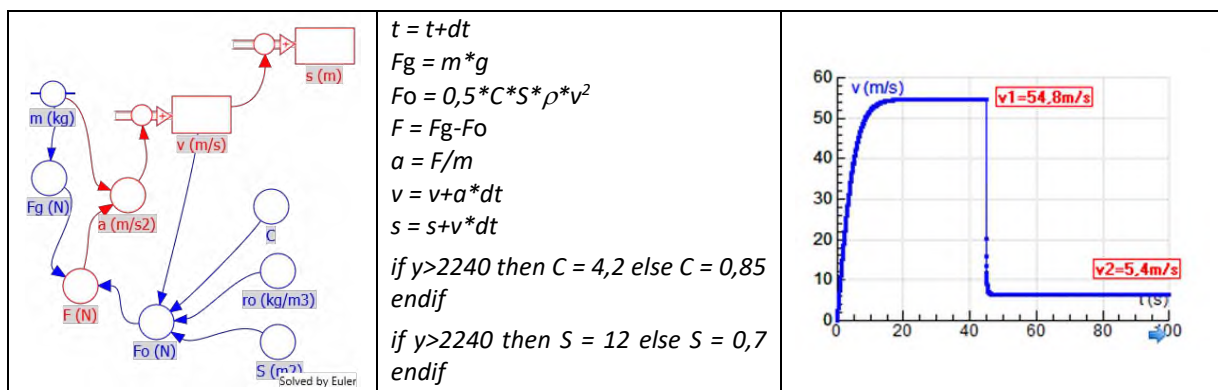
2.1.1 Ako sila ovplyvní pohyb

Aktivita smeruje k tvorbe modelu pohybu parašutistu a k riešeniu problému aké fyzikálne princípy využíva parašutista k tomu, aby bezpečne dopadol na zem. Na základe analógie s predchádzajúcimi modelmi (tabuľka 1) žiaci modelujú pád ťažkej a následne ľahkej lopty vo vzduchu, ktorý porovnávajú s výsledkom video merania. Žiaci model dopĺňajú a rozširujú až sa dopracujú k modelu skoku parašutistu, ktorého výsledky následne porovnajú s reálnym skokom parašutistu nasnímaného na videozázname. Na obrázku 1 je výsledný model (hmotnosť parašutistu $m = 90$ kg, hustota vzduchu $\rho = 1,2$ kg/m³), pričom parašutista vyskočí z výšky 3000 m, padá najskôr bruchom obráteným k Zemi (súčiniteľ odporu $C = 0,85$ a obsah priečneho rezu $S = 0,7$ m²) a vo výške 760 m otvorí padák (súčiniteľ odporu $C = 4,2$ a obsah priečneho rezu $S = 12$ m²) (obrázok 1, ikonografický a textový

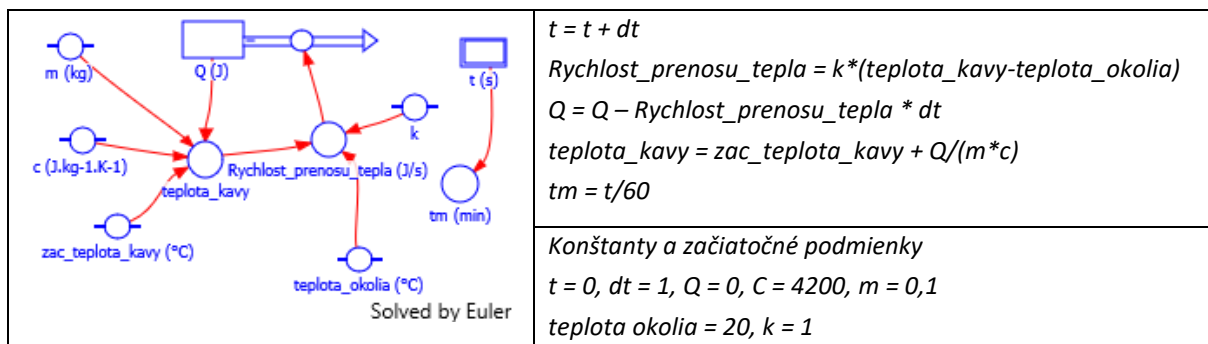
DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitní a pedagogická model a časový priebeh rýchlosti). Model počíta výslednú silu F pôsobiacu počas pohybu, zrýchlenie a , rýchlosť v a dráhu s prejdenu parašutistom s časovým krokom $dt = 0,005$ s.

2.1.2 Ako udržať optimálnu teplotu v dome

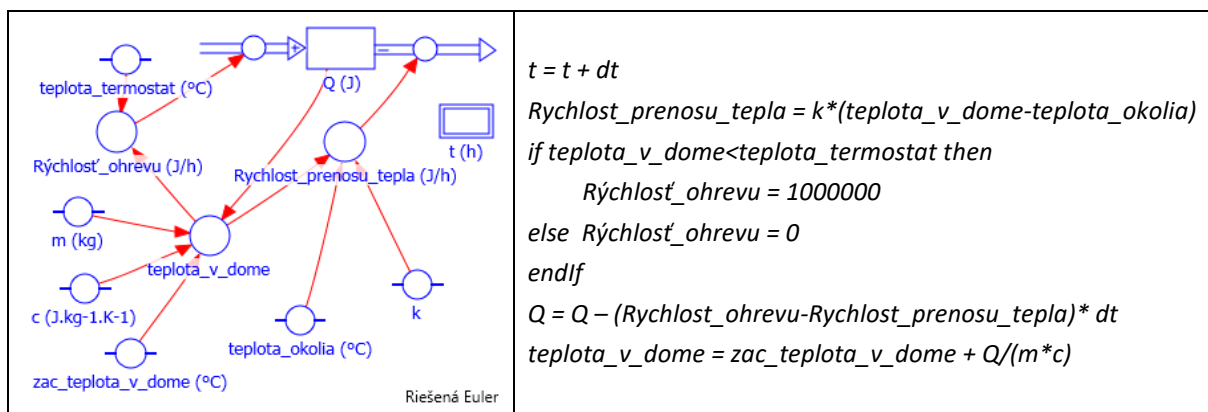
Na základe znalostí a zručností získaných pri predchádzajúcich aktivitách žiaci analyzujú parametre, ktoré ovplyvňujú teplotu v budove. Najskôr však vytvoria jednoduchý model ochladzovania (obrázok 2, ikonografický a textový model), ktorý porovnávajú s výsledkom reálneho experimentu chladnutia šálky s kávou. Následne model upravujú pridávaním ďalších relevantných parametrov do modelu (vonkajšia a vnútorná teplota, tepelná izolácia, kúrenie, klimatizácia), čím model spresňujú a približujú realite (obrázok 3).



Obrázok 1: Model pádu parašutistu, ktorý v istej výške otvorí padák



Obrázok 2: Model ochladzovania kávy



Obrázok 3: Model teploty v dome s kúrením a termostatom

2.2 Zobrazovacie metódy

Pri vnímaní vonkajšieho prostredia využívame naše zmysly ale aj špecifické receptory rozmiestnené po celom tele, z ktorých informácie sa dostávajú do mozgu na vyhodnotenie. Analogicky chceme žiakom priblížiť spracovanie a zobrazenie informácií v digitálnom svete. Ľudský receptor je nahradený senzorom, prenos informácie namiesto nervovej sústavy rieši hardvér a spracovanie má namiesto mozgu na starosti softvér. Aby sme dokázali správne zbierať, vyhodnocovať a interpretovať informácie, je potrebné porozumieť princípom, dôkladne poznať rozsah, možnosti ale aj obmedzenia jednotlivých metód.

Tabuľka 2: Zoznam žiackych aktivít zameraných na zobrazovacie metódy

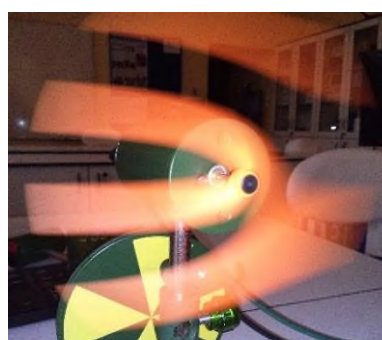
Téma aktivity	Stručný obsah
Ako zvuk premeníme na obraz	Meraním času a intenzity vyslaného a po odraze od prekážky získaného signálu žiaci objavujú základy echolokácie.
Termografia – bezkontaktné meranie teploty	Meraním povrchovej teploty materiálov s rôznou emisivitou si žiaci ozrejmuju princípy termografie.
Ako fotiť pohybujúce sa objekty	Fotografovaním v manuálnom režime žiaci odhaľujú význam parametrov pre získanie ostrého obrazu.

2.2.1 Termografia – bezkontaktné meranie teploty

Aktivita vychádza z vedomostí o žiarení absolútne čierneho telesa a rozširuje diskusiu o praktické aplikácie a zavedenie pojmu šedé teleso. Žiaci hľadajú odpoveď na otázku, ako funguje bezkontaktný teplomer a ako dokáže rozlíšiť a zobraziť teplotu ľudského tela, izbovú teplotu a teplotu nápoja, po stlačení príslušného tlačidla. Od bezkontaktného teplomera prechádzajú k termografickému zobrazovaniu rozloženie povrchových teplôt objektov.

2.2.2 Ako fotiť pohybujúce sa objekty

Takmer každodenne vytvárame fotografie pomocou mobilu alebo digitálneho fotoaparátu, spravidla v automatickom režime. Pri záberoch pohybujúcich objektov nás môže prevapíť výsledok, napr. ako je zobrazený na obrázku 4. Žiaci postupne fotografujú a sledujú parametre doby expozície a ISO pri rôznych režimoch fotografovania. Porozumením ich vplyvu a významu nastavenia sa dopracujú k vytvoreniu fotografie rýchlo sa pohybujúceho objektu. Ich záznam je obdobný ako na obrázku 4b, preto hľadajú informácie o CMOS čipe, na ktorý je obraz zaznamenávaný. V závere ozrejmuju vlastné zistenia a príčiny nekorektne získaných záberov.



Obrázok 4: Fotografie rotujúcich listov vrtule lietadla počas letu a rotujúcej vrtuľky.

2.3 Databázové systémy

Žiaci hľadajú odpovede na výskumné otázky, oboznamujú sa s predpoveďami a hypotézami, ktoré majú charakter kvalifikovaného odhadu. Chceme ukázať, že globálne dáta majú obrovský význam pri tvorbe rozhodnutí v prospech životného prostredia, plánovaní náročných investícií, pri strategických rozhodnutiach národného hospodárstva. Na riešenie výskumných otázok a potvrdenie hypotéz je možné použiť údaje dostupné z dlhodobých sledovaní a záznamov v databázach.

Tabuľka 3: Zoznam žiackych aktivít zameraných na databázové systémy.

Téma aktivity	Stručný obsah
Riešenie Fermiho úloh s využitím údajov z databáz	Oboznámenie sa so stratégiou riešenia Fermiho úloh, ich tvorba a využitie dostupných údajov pri ich riešení.
Dokážeme formulovať predpoveď, potvrdiť/vyvrátiť hypotézu?	Formulácia predpovedí a hypotéz, ktoré majú charakter kvalifikovaného odhadu.

2.3.1 *Riešenie Fermiho úloh s využitím údajov z databáz*

Formulácia výskumných otázok so zaujímavým obsahom (Fermiho úloh) môže byť pre žiaka inšpiratívna a podnecujúca ho k samostatnej poznávacej činnosti. Žiaci prechádzajú vzorové riešenia, aby pochopili postup riešenia takýchto úloh. Ich úlohou je formulovať a zodpovedať vlastnú výskumnú otázku, ktorej riešenie si vyžaduje získanie údajov z databáz vedeckých informácií.

3 CHÉMIA

Súbor metodík vytvorených pre predmet **IvPVaM, Zošit CHÉMIA**, pozostáva z 2 častí: „Modely a modelovanie vo výučbe chémie“ a „Databázy vo výučbe chémie“.

V prvej časti sme navrhli metodiky zamerané na modelovanie v programe ChemSketch. Program ChemSketch umožňuje písanie vzorcov, chemických reakcií a kreslenie chemických aparátúr, žiaci sa naučia zobrazovať štruktúru molekúl rôznymi spôsobmi: pomocou guľôčkového, kalotového, tyčinkového modelu, získajú predstavy o väzbových dĺžkach a uhloch, naučia sa pracovať s hotovými databázami vzorcov aromatických zlúčenín, aminokyselín, sacharidov, DNA/RNA bázy atď. Metodiky riešia problém spojený s nedostatočným záujmom žiakov o učivo organickej chémie čo súvisí s jeho porozumením. Pre žiakov je náročné zdôvodniť chemické vlastnosti organických látok v súvislosti s ich štruktúrou, čo si vyžaduje priestorovú predstavivosť – vedieť pretransformovať dvojrozmerný obraz do trojrozsmernej podoby.

Výhodou programu ACD/ChemSketch je jeho dostupnosť na viacerých internetových adresách ako freeware verzia, ktorý je možné i bez registrácie využívať na osobné i vzdelávacie účely [5]. Keďže k tomuto programu existuje niekoľko verejne dostupných manuálov v slovenčine i češtine, vzniká otázka v čom je prínos navrhnutých metodík? Odpoveďou je, že učivo sa nesprístupňuje formou „kuchárskej knihy“ ale metódou aktívneho učenia. V metodikách sa uplatňuje riadené bádanie – žiaci riešia problém sformulovaný učiteľom na základe pripraveného postupu, pričom výsledok nepoznajú a nasmerované bádanie – žiaci riešia problém sformulovaný učiteľom na základe postupu, ktorý sami pripravujú (navrhujú).

Navyše týmto spôsobom sa rozvíjajú kompetencie potrebné pre uplatnenie sa na trhu práce v 21. storočí, hlavne digitálna, vedecké kompetencie – rozvíja aj záujem žiakov o chémiu ako vedu [6], kompetencie spojené s myslením a učením apod.

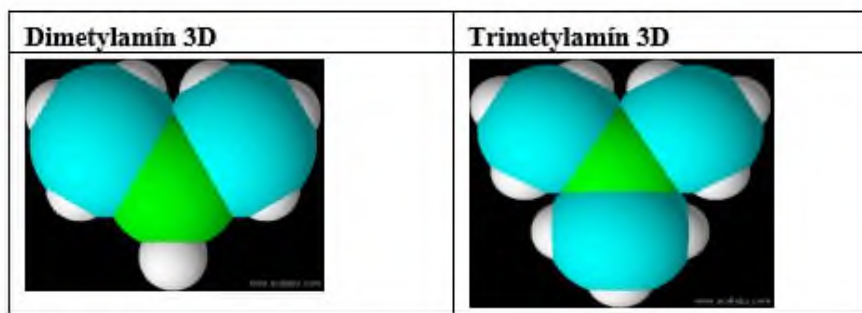
4 MODELOVANIE A VYUŽÍVANIE DATABÁZ VO VÝUČBE CHÉMIE

4.1 Metodiky vytvorené v programe ChemSketch

V tejto kapitole uvádzame tieto metodiky:

- Kreslíme molekuly uhlíkovodíkov.
- Tvoríme molekuly derivátov uhlíkovodíkov a reakčné schémy.
- Izoméria v organickej chémii.
- Kreslíme molekuly látok v živých organizmoch.
- Kreslíme chemické aparatúry a vypracúvame chemické protokoly.

Nasledovná ukážka úlohy z metodiky Tvoríme molekuly derivátov uhlíkovodíkov a reakčné schémy poukazuje na kompetencie, ktoré sa týmito metodikami rozvíjajú [7].



Obrázok 5: 3D modely dimetylamínu $\text{NH}(\text{CH}_3)_2$ a trimetylamínu $\text{N}(\text{CH}_3)_3$

Úloha: Nakreslite molekuly a vytvorte 3D modely dimetylamínu $\text{NH}(\text{CH}_3)_2$ a trimetylamínu, $\text{N}(\text{CH}_3)_3$. Vložte ich do nasledujúcej tabuľky a skúste na základe modelov vysvetliť vyššiu zásaditosť dimetylamínu ($\text{pK}_\text{B} = 3,32$) oproti trimetylamínu ($\text{pK}_\text{B} = 4,19$). Výsledkom sú vykreslené molekuly na obrázku 5.

Vysvetlenie: Kvôli kladnému indukčnému efektu troch metylových skupín by mal byť trimetylamín zásaditejší ako dimetylamín. Nie je tomu tak, pretože metylová skupina je objemnejšia ako atóm vodíka a atóm vodíka sa na voľný elektrónový pár dusíka v trimetylamíne viaže ťažšie z priestorových dôvodov.

Podobne sú spracované ďalšie metodiky z tejto databázy ako napr. Izoméria, kde si žiaci musia uvedomiť, čo spôsobuje rozdiely vo fyzikálnych i chemických vlastnostiach organických zlúčenín, i keď majú tieto zlúčeniny rovnaké usporiadanie atómov v molekule.

Metodiky boli hodnotené učiteľmi gymnázia prostredníctvom dotazníkovej metódy. Z analýzy odpovedí vyplynulo, že učelia v týchto metodikách oceňujú: názornosť, rozvíjanie priestorovej predstavivosti žiakov pri tvorbe modelov (rotácia), vizualizáciu. Metodiky navrhujú využívať pri fixácii vedomostí, pri písaní protokolov z laboratórnych cvičení, pri príprave na maturitu apod.

Cieľom metodiky „Modelovanie biosyntézy nukleových kyselín a bielkovín, genetický kód“ je naučiť žiakov využiť programovací jazyk Python na riešenie zápisov sekvencií aminokyselín, translácie a transkripcie.

4.2 Metodiky na tvorbu a využívanie databáz vo výučbe chémie

Metodika „Periodickosť vlastností prvkov – práca s databázami“ je zameraná na prácu žiakov s internetovou stránkou <https://www.schoolmykids.com/>, na základe ktorej môžu žiaci vysvetliť rozdielne hodnoty atómového polomeru, elektronegativity, ionizačnej energie a elektrónovej afinity prvkov v periódach a skupinách PSP.

V metodike „Kyslé dažde“ sa žiaci naučia využívať počítačom podporované merania na stanovenie vybraných ukazovateľov kvality vody v kyslých zrážkach (teplota, pH, vodivosť, obsah chloridových a dusičnanových aniónov) spracovať dáta do tabuliek a grafov, interpretovať a analyzovať dáta získané meraním.

Úloha: Porovnajte namerané hodnoty s hodnotami a koncentračnými limitmi pitnej vody na základe informácií na nasledujúcich stránkach: https://www.ecoli.sk/files/documents/nv_496_2010.pdf, http://www.bvsas.sk/files/o-vode/ukazovatele-kvalityvody/zakonypreludisk3542006_zz_20160101-1.pdf.

5 ZÁVER

Pripravený systém zošitov, z ktorých každý obsahuje 8–10 didakticky pripravených tém na výučbu na stredných školách, nemusí byť viazaný len na predmet Informatika v prírodných vedách a matematike, ale každý zošit je použiteľný aj samostatne v jednotlivých prírodovedných predmetoch a matematike.

Naše skúsenosti z prípravy učiteľov na túto výučbu sú aj pozitívne aj negatívne. Medzi pozitívne by sme mohli zaradiť: (1) Učitelia považujú zvolené témy za zaujímavé a prínosné pre výučbu. (2) Pripravené metodické materiály sú spracované systematicky. Recenzie pomohli materiálom vylepšiť. (3) Učitelia oceňujú možnosť absolvovať inováčné vzdelávanie pred výučbou predmetu. Medzi negatívne skúsenosti patrí predovšetkým: (1) Učitelia sa sťažujú na to, že „žiaci nemajú dostatočné vedomosti, aby toto zvládli“, a pod. (2) V niektorých témach sú sťažnosti na to, že „táto téma sa nedá odučiť za navrhované dve hodiny“, a pod. Vo väčšine tém je však možnosť rozsah zredukovať. Výučbu realizovalo ešte málo učiteľov (prebieha ďalšie inováčné vzdelávanie), preto štatistické vyhodnotenie obsahu, kvality spracovania a ďalších parametrov predpokladáme po skončení projektu. Napriek tomu, predmet je na viacerých školách zavedený do výučby. Jeho obsahová zložitosť si vyžaduje pevné rozhodnutie školy, ale tiež rozhodnutie učiteľov, ktorí ho chcú učiť. Budúcnosť pripravených učebných materiálov – zošitov vidíme v ich využívaní aj v jednotlivých prírodovedných predmetoch, informatike a matematike.

6 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.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] ANDREJKOVÁ, G. et al: Koncepcia predmetu Informatika v prírodných vedách a matematike. In: DRÁBKOVÁ, J. a BERKI, J., eds. Sborník konference Didinfo 2018, Liberec, 2018. ISBN 978-80-7494-424-6, ISSN 2454-051X. Elektronický zborník. Dostupné na: didinfo.net/images/DidInfo/files/didinfo_2018.pdf.
- [2] ANDREJKOVÁ, G., KIMÁKOVÁ, K., MIŠIANIKOVÁ, A., TKÁČOVÁ, Z.: Príprava učiteľov na výučbu predmetu Informatika v prírodných vedách a matematike. In: HORVÁTHOVÁ, D., MICHALÍKOVÁ, A., ŠKRINÁROVÁ, J., VOŠTINÁR, P., eds. Zborník konferencie DIDINFO 2019, Banská Bystrica, 2019. ISBN 978-80-557-1533-9, ISSN 2454-051X. Elektronický zborník. Dostupné na: <https://www.fpv.umb.sk/app/cmsSiteAttachment.php?ID=4558>.
- [3] TIMKOVÁ, V., JEŠKOVÁ, Z.: Computer modeling in physics teaching. In: *The role of laboratory work in Improving Physics Teaching and Learning 2018*, Springer, Krakow, Poland, 2018, s. 121–130.
- [4] HECK, A., KEDZIERSKA, E., ELLERMEIER, A. L.: Design and implementation of an integrated computer working environment for doing mathematics and science. In: *Journal of Computers and Science Teaching 2009*, roč. 28, č.2., s. 147–161.
- [5] VERMIŘOVSKÝ, J.: *Výuka chemie s programem ChemSketch 12.0*. Výukový materiál pro projekt Perspektiva 2010. 43 s.
- [6] ŠULCOVÁ, R., ZÁKOSTELNÁ, B.: Možnosti využití vybraných elektronických prostředků v přírodovědném vzdělávání. In: *Digitální média a metody poznávání ve výuce chemie – XXII. Mezinárodní seminář o výuce chemie, sborník příspěvků*. Praha, 2012, 136. s. ISSN 1214-9187.
- [7] KARÁSKOVÁ, N., KOLÁŘ, K.: Molekulární modely a výuka chemie na gymnáziu a ZŠ. In: *Didaktika chemie a její kontexty – sborník příspěvků z XXIV. Mezinárodní konference o výuce chemie*, 20.–21. 5. 2015, Brno, Masarykova univerzita, 2001, 270 s. ISBN 978-80-210-7996-0.

Rozvoj kritického myslenia študentov učiteľstva informatiky prostredníctvom prípadových štúdií

Critical Thinking Development of Informatics Pre-Service Teachers with the Support of Case Studies

Gabriela Lovászová, Nika Klimová

Katedra informatiky, Univerzita Konštantína Filozofa v Nitre

Tr. A. Hlinku 1

949 74 Nitra

Slovensko

glovaszova@ukf.sk, nika.klimova@ukf.sk

ABSTRAKT

Technologické a spoločenské zmeny súvisiace s informatizáciou života a spoločnosti si vyžadujú aj zmeny v oblasti vzdelávania. Jednou z hlavných zručností, ktoré žiak potrebuje pre svoj život v informáciami presýtenom svete, je kritické myslenie. Práve tejto zručnosti sa venujeme v našom článku. Na rozvíjanie kritického myslenia v príprave budúcich učiteľov informatiky využívame metódu prípadových štúdií. Vytvorili sme zbierku didaktických situácií, ktoré sme spracovali do prípadových štúdií so štruktúrou AAA (anotácia, analýza, alternácia) a doplnili o cvičenia zamerané na kritické uvažovanie o danom prípade a ďalšie tvorivé rozpracovanie témy. Ako ukážku uvádzame prípadovú štúdiu na tému E-mail.

ABSTRACT

Technological and social changes related to informatisation of life and society require changes in education. One of the main skills, a student needs for his/her life in an overpressure of informational world, is critical thinking. It is the skill that we present in our article. We use the case study method to develop critical thinking in preparing pre-service Informatics teachers. We created a collection of didactic situations, which we processed into case studies with the structure of AAA (annotation, analysis, alternation) and supplemented by exercises focused not only on critical but also on creative thinking elaboration of the topic. As an example, we present a case study on E-mail.

Kľúčová slova

Vyučovanie informatiky, kritické myslenie, prípadová štúdia, reflexia.

Keywords

Informatics education, critical thinking, case teaching method, reflection.

1 ÚVOD

Kritické myslenie je aktuálna globálna téma súčasnosti. Rozvoj informačných a komunikačných technológií umožňuje ľuďom jednoduchší prístup k množstvu informácií. Problémom môže byť, ako zaobchádzať s toľkými informáciami, ako ich analyzovať, posudzovať, hodnotiť. Na to je potrebné kritické myslenie.

V demokratickej spoločnosti zohráva kritické myslenie dôležitú úlohu. Byť aktívnym a informovaným občanom, orientovať sa v komplexných problémoch súčasnosti a mať schopnosť zodpovedne sa rozhodovať ako volič znamená kriticky myslieť.

Kritické myslenie sa považuje za jeden z hlavných, dosiaľ nevyriešených, pedagogických problémov. Viacero autorov opisuje tento pojem rôzne. Sternberg v [1] sa prikláňa k názoru, že kritické myslenie je súbor mentálnych procesov, stratégií a znázornení, ktoré ľudia používajú na riešenie problémov;

schopnosť rozhodovať a učiť sa nové koncepty. Autori v [2] tvrdia, že rozvoj kritického myslenia má byť primárnym cieľom prírodovedného vzdelávania so schopnosťou vyvodiť platné závery, identifikovať vzťahy, analyzovať pravdepodobnosť, predpovedať v závislosti od logických rozhodnutí a riešiť komplexné problémy. Za podstatné znaky kritického myslenia považujú analýzu, hodnotenie, dedukciu, indukciu a argumentáciu. Ruisel v [3] opisuje kriticky mysliaceho človeka ako človeka s otvorenou myslou, prirodzene zvedavého, flexibilného, chápujúceho rozmanité hľadiská, hľadajúceho argumenty pre formuláciu konečných rozhodnutí, odmietajúceho povrchnosť, pred prijatím záverov vždy zvažujúceho dôkazy bez ohľadu na to, kto je ich nositeľom. Podobné vlastnosti zaraďuje Facione [4] medzi osobnostné dispozície, ktoré spolu s kognitívnymi schopnosťami tvoria jadro kritického myslenia. Ako kognitívne schopnosti uvádza interpretáciu, analýzu, hodnotenie, usudzovanie, vysvetľovanie, sebareguláciu. Teda kriticky mysliaci človek má také osobnostné predpoklady a mentálne schopnosti, ktoré mu umožňujú interpretovať, analyzovať a hodnotiť kvalitu informácií a myšlienkových postupov (vlastných alebo cudzích), na základe toho formulovať úsudky, prezentovať a obhajovať závery.

Vo vzdelávaní s cieľom formovania kriticky mysliacej osobnosti, ktorá sa vie úspešne začleniť do spoločnosti a podieľať sa na jej ekonomickom a kultúrnom rozvoji, je potrebné zamerať sa na rozvoj kritického myslenia a s ním súvisiacich vyššie uvedených osobnostných vlastností a kognitívnych schopností. Výsledky štúdie [5] z viac ako 150 krajín ukazujú, že viac ako 70 % vzdelávacích systémov uprednostňuje 6 kognitívnych zručností, t. j. gramotnosť, počítanie, komunikácia, tvorivosť, kritické myslenie a riešenie problémov. Vo výskume [6] zaoberajúcom sa predpovedaním potrebných zručností v blízkej budúcnosti sa dospelo k záveru, že z celkového počtu študovaných článkov 75 bol 30krát spomenutý pojem „kritické myslenie“, čo je viac než ostatných 10 pojmov. Zároveň autori uviedli, že pojem „kritické myslenie“ bol druhým najpoužívanejším pojmom, čo svedčí o aktuálnosti problému začleňovania kritického myslenia do vzdelávacích systémov.

V súvislosti s rýchlym vývojom informačno-komunikačných technológií v informačnej ére a slobodným prístupom k množstvu informácií rôznej kvality by mali byť kriticky mysliaci študenti schopní tvoriť úsudky a rozhodnutia, ktoré sú podložené informáciami a argumentmi a zahŕňajú v sebe úvahy. Takíto študenti vedia pomocou informačno-komunikačných technológií klásť a odpovedať na otázky týkajúce sa problému, posúdiť vhodnosť zdroja pre daný problém, uplatniť argumenty pre tvrdenia na základe ich súladu s inými tvrdeniami, prepojiť fakty, nápady a predstavy, navrhnúť nové témy na diskusiu [7]. Je zrejmé, že kritické myslenie je v dnešnej spoločnosti dôležité, oblasť vzdelávania nevynímajúc.

Viacero reforiem vo svete zameraných na edukáciu zdôrazňuje rozvoj kritického myslenia. Halpern v [8] tvrdí, že kriticky myslieť sa dá naučiť. Vytvorila štvordielny model výučby, prostredníctvom ktorého vyučuje kritické myslenie spočívajúce v motivácii študentov, precvičovaní kritického myslenia, riadení vzdelávacích aktivít, monitorovaní a hodnotení.

2 KRITICKÉ MYLENIE VO VZDELÁVANÍ BUDÚCICH UČITEĽOV

Rozvíjaním kritického myslenia budúcich učiteľov sa zaoberáme v projekte APVV-15-0368 s názvom Prax v centre odborovej didaktiky, odborová didaktika v centre praktickej prípravy. Ide o spoluprácu 3 fakúlt Univerzity Konštantína Filozofa v Nitre. Na projekte sa podieľa 50členný riešiteľský kolektív. Projekt sa zameriava na problematiku skvalitnenia a rozšírenia praktickej prípravy učiteľov, ktorá sa do pozornosti dostala v súvislosti s výsledkami medzinárodných meraní PISA. Ciele projektu sú orientované na identifikáciu adaptívnych vyučovacích stratégií uplatňujúcich kognitívne orientovaný prístup pre rozvoj kritického a tvorivého myslenia žiakov a ďalších kľúčových psychodidaktických tém a stratégií v konkrétnych odborových didaktikách a ich implementáciu v pregraduálnej praktickej príprave učiteľov sekundárneho vzdelávania prostredníctvom excelentného centra praktickej prípravy. Zámerom riešiteľov je definovať štandard praktickej prípravy absolventov učiteľských študijných programov v rámci príslušných odborových didaktík.

Našou metodológiou je identifikovať kľúčové adaptívne vyučovacie stratégie uplatňujúce kognitívne orientovaný prístup pre rozvoj kritického a tvorivého myslenia žiakov a ďalšie kľúčové psychodidaktické stratégie v odborovej didaktike. Tie sa podarilo identifikovať ako:

- sebaregulácia,
- systematické a interpretatívne zručnosti,
- argumentácia,
- riešenie problémov,
- hodnotenie,
- čitateľské zručnosti [9].

Identifikované všeobecné vyučovacie stratégie pre rozvoj kritického a tvorivého myslenia sa majú následne implementovať do vzdelávania budúcich učiteľov jednotlivých vyučovacích predmetov, ktoré pozostávajú z didaktickej prípravy na univerzite a pedagogickej praxe na cvičných školách.

3 PRÍPADOVÁ ŠTÚDIA AKO VYUČOVACIA METÓDA

Holistický prístup, v ktorom je možné nájsť konštitutívne zložky prípadu, zachytiť skúmaný prípad alebo prípady v kontexte reálneho života a dospieť k jeho hlbšiemu porozumeniu, sa používa v prípadových štúdiách. Základné rozdelenie prípadových štúdií z hľadiska účelu je vedecké a edukačné. Pre naše účely používame edukačné prípadové štúdie, ktoré sú určené pre pedagogické účely, kde študentom učiteľstva informatiky predkladáme reálne situácie z vyučovania v rámci predmetu informatika.

Budúci učelia sa prostredníctvom prípadových štúdií učia, ako aplikovať teoretické koncepty do praxe a zároveň si rozvíjajú kritické myslenie. Úspešnosť metódy prípadových štúdií pri rozvoji kritického myslenia študentov opisuje Mayo [10]. Vo svojom vyučovaní sa zameriaval na rozvoj kritického myslenia, ale polovici študentov dal okrem materiálov aj prípadové štúdie a zvyšku nie. Zistil významné rozdiely na základe čoho zhodnotil, že prípadové štúdie sú vhodné pre rozvoj kritického myslenia. Popil vo svojej štúdii [11] uvádza, že konkrétne ilustrácie určitých princípov sa nachádzajú v takmer každej životnej situácii a skúmanie detailov určitých prípadov stimulujú u študentov vyššie myslenie. Kunselman a Johnson [12] odporúčajú prípadové štúdie integrovať do vyučovacieho procesu preto, že študenti sa prostredníctvom nich stávajú lepšie informovanými. Prípadová štúdia je efektívna najmä pre to, že podporuje aktívne učenie sa, rozvíja kritické myslenie a je orientovaná na študenta.

Prípady do našich prípadových štúdií vyberáme z databázy didaktických situácií vo forme audiozáznamov a fotografického materiálu, ktorú budujeme v Centre praktickej prípravy na UKF. Záznamy sme získali:

- z vyučovacích hodín didaktiky informatiky, kde sa simuloval vyučovací proces. (budúci učiteľ viedol vyučovanie a ostatní budúci učelia simulovali učiacich sa žiakov),
- z riešení didaktických zadaní pre budúcich učiteľov zaznamenaných formou screencasting s hlasovým komentárom riešenia,
- z reálneho vyučovania na základnej alebo strednej škole.

Záznamy vyučovacieho procesu doplnené o odbornú a metodickú reflexiu a spracované formou prípadových štúdií tvoria zbierku ukážok rôznych metodických postupov, ktorá sa bude používať v príprave budúcich učiteľov informatiky ako študijný materiál.

4 E-MAILY: PRÍPADOVÁ ŠTÚDIA V DIDAKTIKE INFORMATIKY

V tejto časti uvádzame príklad prípadovej štúdie, ktorá slúži ako študijný materiál pre študentov učiteľstva informatiky. Pri vypracovaní tejto a ďalších prípadových štúdií sme zvolili metodiku AAA. Je zložená z nasledovných metodických krokov:

- anotácia,
- analýza,
- alternácia [13].

Anotácia je stručný opis pozorovanej časti vyučovania, je analógiou výskumnej deskripcie a obsahuje najdôležitejšie informácie o priebehu opisovanej situácie, či už celku alebo časti vyučovacej hodiny. Tie sú: cieľ, téma, obsah výučby, činnosť učiteľa a následná činnosť žiakov. Cieľom anotácie je, aby si recipient (študent) vedel pri čítaní predstaviť obraz hodiny ako celku.

Následne naň nadväzuje analýza, teda rozbor vyučovacej hodiny, prípadne len jej časti. Ide analógiu o výskumnej interpretácie, ktorá prináša charakteristiku sledovanej situácie s ohľadom na ciele hospitácie, v našom prípade identifikovanie stratégií na rozvoj kritického myslenia a zhodnotenie dosiahnutia vzdelávacích cieľov.

Alternácia je tretou časťou metodiky AAA, kde sa recipient zamýšľa nad zmenou, t. j. návrhom na zlepšenie, úpravou alebo pozmenením výučbovej situácie. Alternácia je spojená s diskusiou. Obsahuje vyhodnotenie kvality analyzovanej situácie a návrh na zmenu spojenú s úvahou o výhodách a nevýhodách. Pre recipienta predstavuje priestor pre vlastné kreatívne návrhy, aby mohol navrhnúť iné vyučovacie metódy a postupy, ktoré vedú k zlepšeniu vyučovacej hodiny s ohľadom na dosiahnutie vzdelávacích cieľov.

Ako ukážku z našej zbierky predstavujeme prípadovú štúdiu z vyučovacej hodiny, ktorá je písaná v línii metodiky AAA. Prípad je opísaný podrobným zdokumentovaním v textovej forme a k dispozícii je aj audiozáznam a fotografický materiál. Témou vyučovacej hodiny informatiky sú E-maily. Štúdia sa zameriava na sledovanie uplatňovania vyučovacích stratégií na rozvoj čitateľských zručností žiakov.

4.1 Anotácia

Prípad opisuje dve aktivity na vyučovacej hodine informatiky v 8. ročníku základnej školy. Na hodine je prítomných 9 žiakov a učiteľka informatiky.

Aktivita A

Učiteľka rozdelila žiačky do skupín (troch dvojíc a jednej trojice) a zadala úlohu:

„Tu máme typy ľudí, ktorí posielajú e-mail (Obr. 1). Chcem počuť váš názor, ktorý z typov e-mailerov by ste nechceli stretnúť vo vašej e-mailovej schránke.“

Kryptograf: najviac dráždivý typ emailera, tento človek vyplní svoje e-maily s nevysvetliteľnými akronymami, väčšinou aby zapôsobil na šéfa.

Autor: myslí si, že píše román, nie e-mail.

Preposielateľ: prepošle každému reťazový list a vtip, ktoré dostal, ale zjavne bez toho, aby o tom porozmýšľal.

Hráč: tvrdí, že nedostal váš e-mail. Celkom to štvie; ale v týchto dňoch je to ťažko dokázateľné kvôli spamom a filtrom.

Používateľ smajlíkov: emotikony patria medzi najmenej dráždivé typy emailerov.

Stručný prehľadca: najmenej dráždivý typ emailera. Jeho e-maily sú krátke a k veci.

Obrázok 1: Typy e-mailerov – text pre zadanie úlohy

Žiačky približne 2 minúty samostatne pracovali v skupinách, potom skupina prezentovala svoj názor. Prvá skupina označila za nepríjemný typ pisateľa e-mailov Preposielateľ. V odôvodnení uviedli, že:

- Je nepríjemné, keď stále chodia správy, o ktoré nestojíte, a je ich veľa.
- Nepríjemné sú vyhrážky, ak nepošlete mail ďalej. Uvádzajú príklady vyhrážok („zomrie celá rodina“).
- Neveria lákavým sľubom, ktoré sa v mailoch objavujú. Uvádzajú príklady sľubov („prepošli a dostaneš mobilný telefón“).

Učiteľka ďalej rozvíjala diskusiu žiakov otázkou: „Prečo sa teda takéto maily posielajú?“ Žiačky odpovedali, že preto, lebo niektorí ľudia veria obsahu takýchto e-mailov. Jedna žiačka a postupne aj

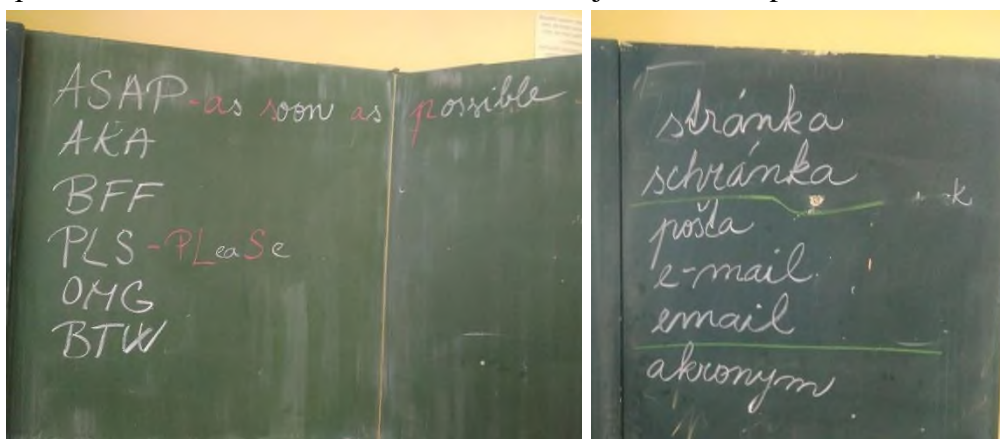
ostatné sa priznali, že tiež tomu verili. Učiteľka preto položila konkrétnu osobnú otázku: „Prečo si ty posielala takéto maily?“ Žiačka odpovedala: „Lebo som sa bála.“ Učiteľka zhrnula, že reťazové maily preposielajú ľudia zo strachu alebo preto, že chcú zažiť niečo pekné. Doplnila ešte, že ohrozenou cieľovou skupinou sú okrem detí aj starí ľudia.

Ďalšia skupina označila za nepríjemný typ e-mailera Hráča. V zdôvodnení uviedli, že keď Hráč neodpovedá alebo zatajuje prijatie pošty, odosielateľ je v strese a Hráč sa na tom zabáva. Učiteľka podporila diskusiu otázkou: „Kedy sa takéto správanie objavuje?“ Žiačky uviedli príklad: „Keď vám niekto dlhuje peniaze a nechce vám ich vrátiť, vyhýba sa vám.“ Učiteľka pridala ďalší príklad: Odosielateľ e-mailu potrebuje odpoveď, napríklad informáciu, kto sa zúčastní organizovanej akcie, a Hráč neodpovedá. Ďalej rozvíjala diskusiu na tému, ako zistíme, či príjemca naozaj náš e-mail nedostal. Vysvetlila, že pri neúspešnom pokuse doručiť e-mail príde o tom odosielateľovi správa. Ak sme takúto správu nedostali, tak pravdepodobne bola naša správa doručená do schránky príjemcu.

Zvyšné dve skupiny žiakov vybrali rovnaké dva typy e-mailerov, preto diskusia k ich výberu už ďalej nepokračovala.

Aktivita B

Učiteľka napísala na tabuľu niekoľko slov. Obráz tabule je na Obr. 2 vpravo.



Obrázok 2: Obráz tabule v Aktivite B

Najprv žiačky diskutovali o význame dvojice slov „stránka“, „schránka“. Stránku vysvetľovali vo význame webová stránka, schránku vo význame poštová schránka – fyzická a elektronická. Jedna žiačka spomenula aj kopírovanie do schránky. Ďalšia skupina diskutovaných slov bola „pošta“, „e-mail“, „email“. Slovo „email“ vo význame farba vyhľadali žiačky na internete. Počas diskusie o významoch podobných slov padla otázka žiačky: „Aký je rozdiel medzi e-mail a gmail?“ Ostatné žiačky tiež nepoznali odpoveď. Učiteľka vysvetlila, že e-mail je všeobecná služba elektronickej pošty, gmail je elektronická pošta od konkrétneho poskytovateľa Google.

Ďalšie diskutované slovo bolo akronym. Žiačky ho nepoznali, ale keď učiteľka uviedla príklad (ASAP), vedeli povedať, čo znamená, ako vznikol a prečo sa používa. Učiteľka aj žiačky uvádzali ďalšie príklady známych akronymov (Obr. 2 vľavo). Pozastavili sa pri skratke PLS (please), ktorá nevznikla zo začiatkových písmen frázy, a teda nie je akronymom.

4.2 Analýza

Témou vyučovacej hodiny bola elektronická pošta. V 8. ročníku môžeme predpokladať, že žiaci už majú zručnosti v komunikácii prostredníctvom konkrétneho e-mailového nástroja. Cieľom tejto vyučovacej hodiny bolo (využívajúc praktické skúsenosti s písaním a prijímaním e-mailov):

- uvažovať o rôznych typoch správania sa pri komunikácii v digitálnom prostredí a hodnotiť ich (Aktivita A),
- hlbšie porozumieť, vedieť interpretovať a správne používať pojmy súvisiace s témou elektronickej pošty (Aktivita B).

Cieľom boli prispôsobené vyučovacie formy a metódy. Namiesto individuálnej práce s počítačom prevládajúcej vo vyučovaní informatiky bola zvolená diskusia a práca v malých skupinách. Téma, forma aj metódy učebných aktivít poskytovali vhodné prostredie pre rozvoj kritického myslenia.

V analýze si všimneme rozvíjanie čitateľských zručností ako jednu z vyučovacích stratégií na podporu kritického myslenia žiakov. Čitateľské zručnosti ovplyvňujú schopnosť žiakov používať písané informácie v učení sa a v bežnom živote. Podľa štúdie OECD PISA [14] je pre dosiahnutie plného porozumenia písaného textu potrebné zvládnuť tieto činnosti:

- získavanie informácií z textu,
- porozumenie a interpretácia textu,
- uvažovanie o forme a obsahu textu a ich hodnotenie.

V Aktivite A žiačky v malých skupinách čítali text a uvažovali o jeho obsahu. V následnej spoločnej diskusii hodnotili obsah textu, dostali priestor na vyjadrenie názoru. Učiteľka podporovala diskusiu podnetnými otázkami, ktoré orientovala na vybavenie si konkrétnych skúseností žiačok a ich reflexiu, sama prispievala do diskusie vhodnými príkladmi a formulovala zhrnutia diskusií.

Slabou stránkou učebnej aktivity bola kvalita textu, s ktorým žiačky pracovali (Obr. 1). Text nie je neutrálny, obsahuje sugestívne vyjadrenia názoru k otázke, ku ktorej sa mali žiačky vyjadriť („najviac dráždivý typ“, „najmenej dráždivý typ“, „celkom to šťve“).

Aktivita B bola zameraná na rozširovanie slovnej zásoby žiačok, na dôkladné porozumenie a aktívnu interpretáciu významu slov. Učiteľka zvolila dobrú vyučovaciu stratégiu – analýzu významu podobne znejúcich slov alebo slov s podobným významom. Žiačky boli schopné vyjadriť rozdiely medzi pojmami, ktoré im boli známe.

Pri prvej dvojici slov „stránka – schránka“ uviedla jedna žiačka aj ďalší význam pojmu schránka okrem fyzická schránka, obal, elektronická schránka – „kopírovanie do schránky“, teda význam v zmysle odkladací priestor na krátkodobé uloženie dát počas prenosu vo vnútri aplikácie alebo medzi aplikáciami (clipboard). Učiteľka na tento význam nemyslela a žiačkinu poznámku si nevšimla. Objavenie ďalšieho významu pojmu bolo výsledkom spoločnej diskusie, škoda, že tento význam pojmu schránka nebol prediskutovaný.

Pri druhej trojici slov „mail (pošta) – e-mail – email“ našli žiačky neznámy význam pojmu „email“ v zmysle farba vyhľadáváním na webe. Aj v tomto prípade žiačky počas diskusie spontánne rozšírili skupinu pojmov o slovo „gmail“, čo bolo pre učiteľku príležitosťou na vysvetlenie rozdielu medzi všeobecným pojmom (e-mail ako služba poskytovaná prostredníctvom internetu) a jeho konkrétnou inštanciou (gmail ako služba konkrétneho poskytovateľa).

V texte z Aktivity A v opise e-mailera Kryptografa sa objavilo slovo „akronym“, význam ktorého, ako sa ukázalo, žiačky nepoznali. Napriek tomu, že opis Kryptografa obsahoval sugestívnu časť „najviac dráždivý typ e-mailera“, žiadna skupina žiačok si ho nevybrala asi z dôvodu, že textu nerozumeli. Je to ukážka toho, že porozumenie textu je dôležitým predpokladom schopnosti hodnotiť obsah textu.

Učiteľka zvolila na vysvetlenie slova akronym v Aktivite B dobrú stratégiu:

1. Učiteľka uviedla konkrétne príklady (inštalácie) pojmu akronym, žiačky si vybavovali svoje skúsenosti s pojmom uvádzaním ďalších príkladov.
2. Otázkou „Ako vznikol akronym?“ učiteľka dosiahla, že žiačky abstrahovali podstatu pojmu a vedeli sformulovať význam slova všeobecne ako „umelé slovo, ktoré vznikne zo začiatkových písmen nejakej frázy“.
3. Upevnenie nového poznatku učiteľka dosiahla uvedením príkladu, ktorý nie je akronymom. Žiačky deduktívnym myšlienkovým procesom zo všeobecného vymedzenia pojmu odvodili, že PLS nie je príkladom akronymu.

4. Zasadenie nového pojmu do kontextu vedomostí dosiahla učiteľka diskusiou o používaní pojmu – kde, kedy, prečo sa používajú akronymy. Tu učiteľka nevyužila príležitosť na prepojenie s témou predchádzajúcej Aktivite A (typy správania sa pri elektronickej komunikácii) upozornením na negatívne stránky používania akronymov v písanom texte.

4.3 Alternácia

Alternácia 1 – výmena poradia učebných aktivít

V Aktivite B sa rozširuje slovná zásoba, upevňuje sa chápanie pojmov do hĺbky, rozvíja sa abstraktné myslenie, interpretačné schopnosti žiakov. To všetko prispieva k hlbšiemu porozumeniu textu. V Aktivite A žiaci uvažujú o obsahu textu, hodnotia ho a formulujú vlastný názor. Pre tieto činnosti je dôležité rozumieť obsahu textu, preto poradie aktivít B, A považujeme za efektívnejšie.

Alternácia 2 – úprava textov

Ako slabú stránku Aktivite A sme uviedli kvalitu textu o typoch e-mailovej komunikácie. Vhodnou alternáciou je upravenie textov tak, aby neobsahovali vyjadrenia podsúvajúce odpoveď na zadanú otázku.

Alternácia 3 – rozšírenie

Vhodným rozšírením vyučovacieho postupu by mohlo byť po Aktivitách B, A zaradenie aktivity na zvládanie ďalšej čitateľskej zručnosti – získavania informácií z textu. Učiteľ pripraví ukážky textov e-mailov zodpovedajúcich rôznym typom e-mailovej komunikácie z predchádzajúcej aktivity – Kryptograf, Autor, Preposielateľ, Používateľ smajlíkov, Manažér (lepšie pomenovanie namiesto Stručný prehľadca z pôvodného textu). K ukážkam e-mailov pripraví konkrétne úlohy pre žiakov, ktorých cieľom je rozvíjanie čitateľských zručností: rozumieť textu v e-mailovej správe, získať informáciu z e-mailu rôzneho typu, rozpoznať pozitívne a negatívne stránky jednotlivých štýlov e-mailovej komunikácie, tvoriť zrozumiteľné e-mailové správy.

Príklady úloh k ukážkam textov e-mailov:

- Čo znamenajú akronymy v ukážke?
- Čo vyjadrujú emotikony v ukážke?
- Zistiť konkrétnu informáciu z textu (z dlhého jednodliateho, z krátkeho štruktúrovaného).
- Upraviť text e-mailu tak, aby bol čitateľnejší.
- Napísať odpoveď na e-mail.

Cvičenia pre tvorbu alternácií:

1. Upravte text Typy e-mailerov (Obr. 1) tak, aby neobsahoval hodnotiace vyjadrenia.
2. Vytvorte ukážky e-mailov rôznych typov komunikácie: Kryptograf, Autor, Preposielateľ, Používateľ smajlíkov, Manažér s obsahom primeraným veku žiakov.
3. Vytvorte úlohy na vyhľadanie konkrétnych informácií z ukážok e-mailov.

5 ZÁVER

Práca s prípadovými štúdiami opisujúcimi autentické didaktické situácie z vyučovania informatiky na základných alebo stredných školách je jednou z vyučovacích metód, ktorú uplatňujeme v príprave budúcich učiteľov informatiky na Univerzite Konštantína Filozofa v Nitre v rámci predmetu Didaktika informatiky. Prínos prípadových štúdií môžeme zhrnúť takto:

- Prípadové štúdie predstavujú prepojenie medzi teoretickou a praktickou prípravou študentov učiteľstva. Praktická príprava študentov počas pedagogickej praxe je založená na získavaní a reflektovaní vlastných pedagogických skúseností. Prípadová štúdia predstavuje podobné učenie sa, avšak nie z vlastnej skúsenosti, ale zo zdokumentovaného autentického prípadu z praxe.

- Analýza prípadu (autentickej didaktickej situácie) v diskusii s inými rozvíja kognitívnu dimenziu kritického myslenia študentov (analýza, hodnotenie, riešenie problémov, argumentácia, formulovanie a prezentovanie záverov), ako aj osobnostné dispozície kriticky mysliaceho človeka (otvorená myseľ, flexibilita, zvedavosť, odmietanie povrchnosti).
- Navrhovanie alternácií prípadu rozvíja tvorivosť študentov v oblasti projektovania vyučovania.

6 POĎAKOVANIE

Tento článok bol finančne podporený Univerzitnou grantovou agentúrou a projektom Prax v centre odborovej didaktiky, odborová didaktika v Centre praktickej prípravy. (APVV-15-0368).

7 BIBLIOGRAFICKÉ ODKAZY

- [1] STERNBERG, Robert J. Critical Thinking: Its Nature, Measurement, and Improvement. Washington, DC: National Institute of Education, 1986.
- [2] TIRUNEH, D. T., M. D. COCK, A. G. WELDESLASSIE, J. ELEN, R. JANSSEN. *Measuring Critical Thinking in Physics: Development and Validation of a Critical Thinking Test in Electricity and Magnetism*, International Journal of Science and Mathematics Education. 2017. s. 663–682.
- [3] RUISEL, Imrich. *Múdrosť v zrkadle vekov*. Bratislava: Ikar, 2005.
- [4] FACIONE, Peter A. Critical Thinking: What It Is and Why It Counts. Measured Reasons LLC. 2015.
- [5] MICROSOFT Corporation, McKinsey & Company's Education Practice. The class of 2030 and life-ready learning: The technology imperative. 2018.
- [6] LAAT, E. V., A. J. A. V. DEURSEN, J. A. G. M. V. DIJK a J. D. HAAN. The relation between 21st-century skills and digital skills: A systematic literature review. *Computers in Human Behavior*, vol.72, s. 577–588, 2017.
- [7] GREENE, J. A., S. B. YU a D. Z. COPELAND. Measuring critical components of digital literacy and their relationships with learning. *Computers & Education*, vol. 76, 2014.
- [8] HALPERN, Diane F. *Thought and knowledge: An introduction to critical thinking* (4th edn). Mahwah: Erlbaum. 2003.
- [9] DUCHOVIČOVÁ, Jana a Robert TOMŠIK. Critical and Creative Thinking Strategies in Teaching Internal Consistency of The Research Tool. *Slavonic Pedagogical Studies Journal*. vol. 6, 2017.
- [10] MAYO, Joseph A. *Using case-based instruction to bridge the gap between theory and practice in psychology of adjustment*. *Journal of Constructivist Psychology*. vol. 17. s. 137. 2004.
- [11] POPIL, Inna. *Promotion of critical thinking by using case studies as teaching method*. *Nurse education today*. vol. 31. s. 201–207. 2011.
- [12] KUNSELMAN, Julie. C. a Kathrine A. JOHNSON. *Using the case method to facilitate learning*. *College Teaching*, vol. 52, s. 87–92. 2004.
- [13] JANÍK, Tomáš, et al. *Kvalita (ve) vzdělávání: obsahově zaměřený přístup ke zkoumání a zlepšování výuky*. Masarykova univerzita. 2013.
- [14] KORŠŇÁKOVÁ, P. et al. *Čitateľská gramotnosť slovenských žiakov v štúdiu PISA 2003*. Bratislava: ŠPÚ, 2006. ISBN 80-85756-95-X.

Kooperativní učení ve výuce informatiky

Cooperative Learning in Teaching Informatics

Rostislav Fojtík
Ostravská univerzita
Dvořákova 7
701 03 Ostrava
Česká republika
rostislav.fojtik@osu.cz

ABSTRAKT

Článek se zabývá možnostmi uplatnění kooperativního učení ve výuce informatiky. Kooperativní učení se řadí k vyučovacím přístupům komunikativního charakteru, které využívá práci v malých skupinách s důrazem na pozitivní vzájemnou závislost a odpovědnost skupiny i jednotlivce, společné vedení i zpětnou vazbu ve skupině. V kooperativních situacích bývá obvykle zaznamenávána vyšší produktivita, než když lidé pracují samostatně. Schopnost kooperace je považována za klíčovou kompetenci i v praktickém životě. I z těchto důvodů patří kooperativní učení mezi vhodné moderní pedagogické přístupy, které se však ve školách používají méně, než by bylo vhodné. Mnohdy stále převažuje frontální výuka, případně skupinové vyučování.

Článek popisuje výsledky a zkušenosti z případových studií, které byly provedeny ve výuce informatiky na středních školách. Žáci v malých skupinách prováděli ve výuce aktivity, při kterých museli společně přistupovat k řešení problémů. Aktivity byly zaměřeny například na tvorbu dokumentů, využití cloudových služeb, zjišťování informací, párové programování. Výsledky ukazují, že při správném využití kooperativních přístupů bývají žáci obvykle více motivováni, více se soustředí na řešení úkolů a výsledky práce bývají lepší než při práci jednotlivců.

ABSTRACT

The article deals with the possibilities of using cooperative learning in computer science education. Cooperative learning is a communicative approach to teaching. This uses work in small groups with an emphasis on positive interdependence and responsibility of the group and the individual, joint leadership and feedback within the group. In cooperative situations, higher productivity is usually recorded than when people work independently. The ability to cooperate is considered a key competence even in practical life. For these reasons, cooperative learning is a suitable modern pedagogical approach, but it is less used in schools than would be appropriate. Often frontal teaching or group teaching still prevails.

The article describes the results and experience of case studies that were carried out in computer science education at secondary schools. Pupils in small groups carried out activities in the class where they had to work together to solve problems. Activities were focused on document creation, use of cloud services, information discovery, paired programming. The results show that with the right use of cooperative approaches, pupils are usually more motivated, more focused on solving tasks, and work results tend to be better than individuals.

Klíčová slova

Informatika, kooperativní učení, programování, skupinové vyučování.

Keywords

Informatics, cooperative learning, programming, group teaching.

1 ÚVOD

Současný svět prochází rychlými a velkými změnami. Škola by měla připravovat své žáky pro praxi nejen v oblasti vědomostí a dovedností, ale také v oblasti komunikace a spolupráce. Jádrem velké části profesí není individuální práce, ale naopak spolupráce v týmu. Člověk je ze své podstaty bytost kooperativní. Ve škole doposud často převládala frontální výuka založená hlavně na individualizované činnosti žáků. Kromě individualistické výukové situace můžeme při výuce využít ještě přístupu kompetitivního a kooperativního. Zatímco základními rysy kompetitivní výuky je soutěž a konfrontace, pak hlavními rysy kooperativní výuky jsou spolupráce, sdílení a podpora. Právě podpora kooperativního učení je v dnešní škole důležitá. Kooperace ve vyučování existuje, když žáci chápou, že mohou dosáhnout svého cíle společně s ostatními žáky [1]; [2]; [3].

Znaky kooperativního učení [4]:

1. Pozitivní vzájemná závislost.
2. Interakce tváří v tvář.
3. Osobní odpovědnost.
4. Formování a využití interpersonálních a skupinových dovedností.
5. Reflexe skupinové činnosti.

Kooperativní učení se dá samozřejmě využívat ve všech oblastech vzdělávání a ve všech předmětech.

2 METODIKA

Kooperativní učení je zajímavý koncept, který by měl podporovat spolupráci mezi účastníky a jejich vyšší motivaci. Podstata je založena na skutečnosti, že účastníci kooperativního učení sdílejí cíl a procedury k jeho dosažení [5].

Pro ověření vhodnosti využití kooperativního učení při výuce informatiky bylo využito kvantitativních i kvalitativních metod pedagogického výzkumu. Byly provedeny tři případové studie na dvou středních školách [6]; [7]. Dále byly využito nestrukturované pozorování a dotazník [8]; [9].

Byla stanovena výzkumná otázka: Budou žáci během kooperativní učení motivovanější a aktivnější než při individualizované výuce?

První případová studie byla prováděná v septimě na gymnáziu. Cílem žáků bylo vyhledávání informací, jejich zpracování a podle přesně stanovených pravidel zapsání do textového dokumentu. Studie probíhala ve dvou třídách. Čas na práci byl dvě vyučovací hodiny. Prvním úkolem žáků bylo rozdělit se na trojice (případně jednu čtveřici). Skupiny se měly domluvit na výběru zpracovávaného tématu a následně každá družina jej měla zpracovat tak, aby se všichni zapojili do řešení. V následující hodině byla provedena reflexe ve skupinách.

Druhá případová studie byla rovněž provedena s žáky gymnázia ze septimy. Obsah práce byl zaměřen na programování. Byla srovnávána práce dvou tříd. V jedné pracovali žáci samostatně a v druhé třídě pracovali žáci ve dvojicích. Úkolem bylo vytvořit desktopovou aplikaci pro výpočet BMI. Žáci si museli najít způsob výpočtu a zařazení do kategorií. Žáci s programováním ve WPF a programovacím jazykem C# začínali (třetí dvouhodinová). Na některé funkcionality prostředí a jazyka si museli přijít sami. Před tímto úkolem si vytvořili jen dvě jednoduché aplikace na převod jednotek. U vytvořené aplikace se hodnotila nejen funkčnost, ale rovněž zpracování uživatelského grafického rozhraní aplikace. V nižších ročnících žáci dříve programovali ve Scratchi a v Pythonu. Při vývoji aplikace museli žáci ve dvojici psát kód pouze na jednom počítači a v psaní se museli pravidelně po deseti minutách střídat.

Třetí případová studie byla provedena na střední škole ve studijním oboru se zaměřením na informatiku. Do výuky byli zařazeni žáci druhého ročníku a zpracovávali úkol v oblasti programování. Žáci pracovali ve dvojicích a vyzkoušet si párové programování jako při metodice nazvané Extrémní programování. Psát kód mohli žáci ve dvojici pouze na jednom počítači a v psaní

se museli po deseti minutách střídat. Úkolem bylo napsat devět metod v programovacím jazyce C#. Metody řešily logické problémy a vzájemně na sebe navazovaly.

Sebereflexe a zpětná vazba byly provedeny v následujících hodinách, ve kterých žáci hodnotili svou práci na základě vzorového řešení. Rovněž vyplňovali dotazník, který se týkal reflexe práce ve skupině. Dotazník obsahoval 13 otázek s uzavřenými odpověďmi. Některé otázky byly navrženy k ověření odpovědí respondentů. Reflexe se týkala rovněž hodnocení vytvořených programů, který si žáci ve skupinách hodnotili.

3 VÝSLEDKY PŘÍPADOVÝCH STUDIÍ

Kooperace může existovat jako vzájemná nápomoc nebo jako vzájemnost. Pro první typ kooperace mezi žáky je vhodné mít stanoven úkol, který mají splnit. Ukazuje se, že vliv kooperace ve výkonu i v rozvoji osobnostních atributů je téměř stejný u toho, kdo učí i u toho, kdo je učený. To znamená, že učební výsledky se zlepšují u obou [10]; [11]. Výsledky pozorování jednoznačně potvrzují, že žáci měli vyšší motivaci pro práci a byli aktivnější než při běžné individualizované výuce. Ve většině skupin sdíleli cíle i prostředky k jejich řešení. Pozitivní vzájemná závislost se dala vysledovat u většiny skupin.

3.1 První případová studie

Výsledky pozorování první třídy:

- Ve třídě bylo 11 chlapců. Jeden z nich přišel později.
- Problém nastal při rozdělování do skupin tak, aby bylo splněno pravidlo trojic a případně jen jedné čtveřice. Při rozdělování samozřejmě hrály svou roli sympatie a antipatie žáků. Nejprve se vytvořily 2 čtveřice a jedna dvojice outsiderů. Vzhledem k tomu, že se žáci nebyli schopni domluvit, muselo proběhnout losování.
- Ustavené skupiny se naopak byly schopny domluvit na výběru tématu.
- Během řešení úkolu bylo možno pozorovat různý přístup. Například jedna trojice si sedla společně, každý měl svůj počítač, ale neustále spolu komunikovali. Druhé dvě skupiny pracovaly od počátku samostatně, bez průběžného domlouvání, žáci využívali sdílený dokument.
- Žáci byli schopni se rychle domluvit na způsobu zpracování a rozdělení jednotlivých činností a úkolů. Při zpracovávání žáci, kteří byli zvyklí na samostatnou činnost, se na počátku spolu moc nebavili. To bylo dáno i povahou práce, kdy bylo potřeba informace vyhledat. Při dokončování práce se žáci ve skupině opětovně sesedli a dodělávali dokument. Tehdy byla kooperace nejintenzivnější.

Výsledky pozorování druhé třídy:

- Ve třídě bylo 6 děvčat a 6 chlapců. Žáci se bez problémů sami rozdělili na trojice (jedna skupina čistě dívčí, jedna čistě chlapecká, dvě skupiny smíšené).
- Na rozdíl od první třídy se žáci ve všech skupinkách se na začátku více domlouvali. Rovněž využívali sdílené dokumenty. Při dokončování práce byla činnost založena opět na větší spolupráci žáků.
- U některých dvojic žáci pracovali na jednom počítači a střídali se u klávesnice. Kooperace byla na vyšší úrovni než u první třídy.

Při řešení zadání museli žáci nejen hledat a zpracovat informace, ale museli využít funkcionalitu textového editoru, kterou zatím neuměli (například vytvoření rejstříku). Společně museli přijít na způsob zpracování. Všechny skupiny byly dostatečně komunikativní a žáci spolupracovali. Přestože se projeví nejaktivnější žáci, kteří se sami ujímali role vedení skupiny, nebyli mezi žáky žádní zcela pasivní jedinci, kteří by se na práci skupiny nepodíleli. Všichni žáci měli snahu úkol splnit kvalitně. Žáci nebyli pasivní nebo se nesnažili úkol splnit jen v minimálním možné podobě.

3.2 Druhá případová studie

Při tomto experimentu byla srovnávána práce jednotlivců a dvojic. V jedné třídě bylo deset chlapců, kteří programovali samostatně. V druhé třídě bylo třináct žáků, kteří se rozdělili na dvojice a jednu trojici. Obě třídy měly stejný úkol.

Výsledky pozorování první třídy:

- V první skupině bylo deset chlapců, z nichž někteří se o programování zajímali i mimo školní výuku. Chlapci pracovali samostatně, ale mohli případně komunikovat se spolužáky, pokud něco nevěděli a nebyli sami schopni najít potřebné informace.
- Komunikace byla ojedinělá, žáci se hlavně zaměřovali na tvoření kódu a s ostatními komunikovali jen, když si nevěděli rady. Žáci se občas ptali i vyučujícího.
- Žáci dobří v programování byli obvykle rychlejší než dvojice ve druhé třídě. Nikdo je nezdržoval a nikomu nemuseli vysvětlovat svůj postup. Čtyři žáci byli schopni program odevzdat již za 60 minut. Dva žáci se ve zbylém čase zaměřili na zlepšování programu nad rámec zadání.
- Naopak slabším žákům trvala práce déle než dvojicím ve druhé sledované třídě.

Výsledky pozorování druhé třídy:

- Žáci se bez problému rozdělili do dvojic a jedné trojice. Skupinky pracovaly na jednom počítači a pravidelně se po deseti minutách střídaly v psaní kódu. Druhý počítač sloužil pouze ke zjišťování informací.
- Komunikace mezi žáky byla větší než v první sledované třídě. Žáci se neustále domlouvali na dalším řešení, hledali si chyby v kódu a v postupu práce a upozorňovali na ně toho, kdo psal. Radili se o dalším řešení aplikace.
- Dvě dvojice stihli vytvořit aplikaci do 60 minut. Celkově však dvojice nebyly rychlejší než jednotlivci. Hlavně u dvojic, které byly složeny z méně schopných žáků, byl čas práce delší, protože museli část času věnovat objasňování řešení.
- Žáci v této třídě byli aktivnější než v první sledované třídě. Do práce se aktivně zapojili i žáci, kteří jsou při individualizované výuce spíše pasivní.

Hodnocení výsledků žáků obou tříd nebylo zásadně rozdílné. Rovněž čas na vytvoření aplikace byl u jednotlivců i dvojic přibližně stejný. Čas byl spíše závislý na schopnostech jednotlivých žáků než na formě výuky. Podle žáků, jim práce ve dvojici pomohla lépe pochopit tvorbu programu a práce se jim zdála zábavnější než při individuálním psaní kódu.

3.3 Třetí případová studie

Výsledky pozorování první třídy:

- Třidu tvořilo 13 žáků (11 chlapců, 2 děvčata). Žáci se sami domluvili na vytvoření dvojic a jedné trojice.
- Na začátku byla kooperace malá. Bylo možné zaregistrovat tendenci některých žáků psát kód a nekomunikovat se spolužákem. Díky povinné výměně žáků u klávesnice po deseti minutách, museli žáci začít více komunikovat, aby si vzájemně objasnili důvody vytvořeného řešení.
- Mezi důvody menší intenzity v komunikaci by se dala zařadit skutečnost, že vyučování probíhalo brzkých ranních hodinách a také to, že žáci nejsou zvyklí spolupracovat.

Výsledky pozorování druhé třídy:

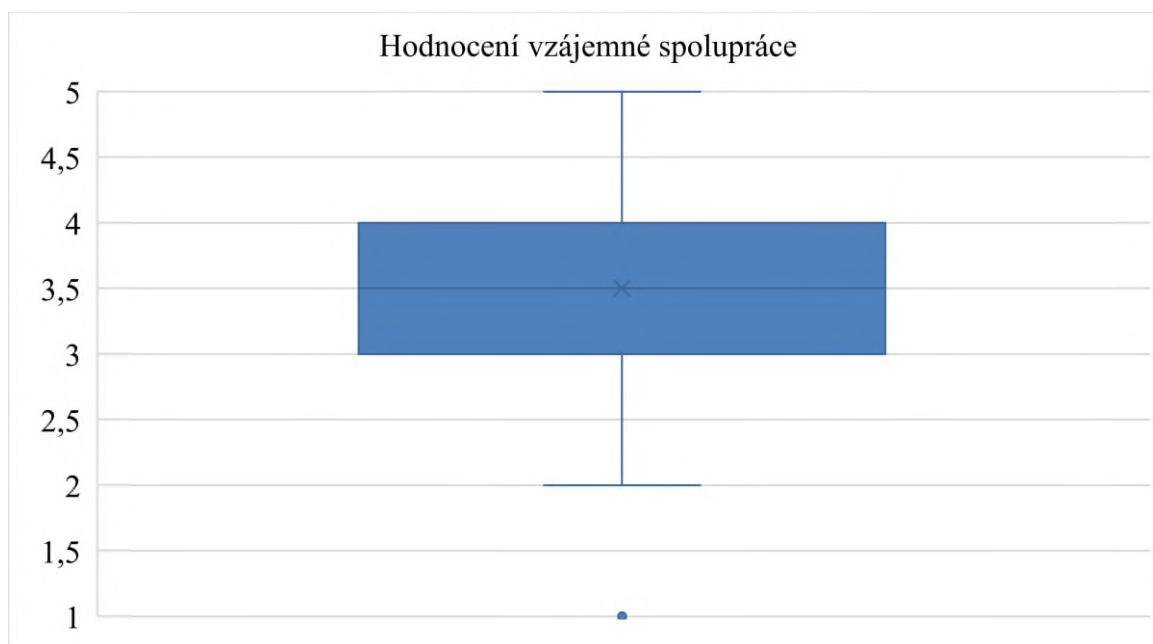
- Ve třídě bylo 14 žáků (1 děvče, 13 chlapců). Žáci jsou obvykle aktivnější než první sledovaná třída. Mají i lepší studijní výsledky.
- V průběhu pozorování byli žáci aktivnější než v první skupině. Ihned od začátku komunikovali a probíhala neustálá výměna informací a společná snaha problém vyřešit.
- Výsledky práce byly lepší než u první sledované třídy.

Zpětná vazba a reflexe skupinové činnosti:

- Někteří žáci konstatovali, že práce ve dvojici je při programování někdy zdržovala. A to tehdy, když museli své řešení druhému žákovi vysvětlovat.
- Naopak oceňovali, že díky komunikaci si ujasnil své myšlenky, mohli je případně na základě domluvy korigovat a zlepšovat navržené řešení.
- Práce ve skupině žákům přišla zábavnější a zajímavější než při programování jednotlivců.

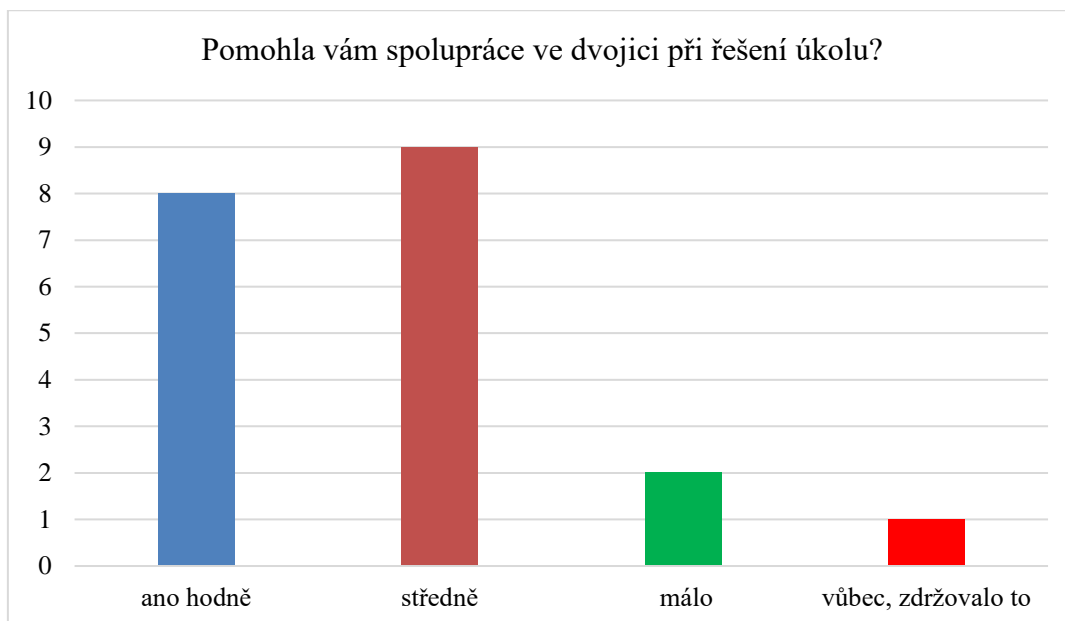
Žáci v následující hodině vyplnili dotazník, ve kterém prováděli reflexi společné práce. Devadesát pět procent respondentů uvedlo, že se plně zapojilo do práce ve dvojici. Na otázku, jak hodnotí zapojení spolužáka odpovědělo 70 % respondentů, že druhý z dvojice žáků se plně zapojil. 15 % respondentů hodnotilo zapojení spolužáka jako částečné a 10 % jako malé zapojení. Jen 5 % respondentů mělo pocit, že jejich spolužák se nezapojil do společné práce vůbec. Většina respondentů (konkrétně 65 %) si myslela, že oba žáci ve dvojici byli téměř stejně aktivní. 20 % respondentů mělo pocit, že aktivnější byl druhý žák ve dvojici a 15 % označilo jako aktivnějšího sebe. Žáci na párovém programování oceňovali i skutečnost, že práce byla zábavnější než při práci jednotlivců. Konkrétně si to myslelo 65 % respondentů.

Následující graf na obrázku číslo 1 zobrazuje hodnocení vzájemné spolupráce ve dvojici žáků. Respondenti hodnotili spolupráci body. Jeden bod znamenal špatnou spolupráci a pět bodů vynikající spolupráci.



Obrázek 1: Hodnocení vzájemné spolupráce

Graf č. 2 ukazuje, jak respondenti hodnotí spolupráci. Většina žáků si myslí, že jim spolupráce při párovém programování pomohla hodně nebo alespoň středně. Jen 5 % respondentů bylo se spoluprací nespokojeno a myslelo si, že práce ve dvojici je zdržovala a sami by to napsali rychleji a lépe. Žáci oceňovali práci ve skupině i v rozhovorech, které byly vedeny v následujících vyučovacích hodinách.



Obrázek 2: Jak vám pomohla spolupráce při řešení úkolu?

Potvrdil se předpoklad, že pro skupinovou práci je nevhodná učebna, kde lavice jsou uspořádány v řadách, jako v klasických učebnách. Žákům se hůře vyměňovalo u počítače a pro práci ve skupině bylo kolem počítače méně místa.

Důležitou roli při práci ve skupinách hrají sociální vazby. Žáci na okraji sociální skupiny a introverti se na počátku hůř zapojovali. Čím více se blížil čas odevzdání, tím více i introverti komunikovali. Potřeba společného vyřešení úkolu vedla ke zlepšení komunikace a spolupráce.

Ve všech sledovaných skupinách se téměř nevyskytovalo dlouhé diktování celých textů spolužákům. I dvojice slabších studentů byla aktivní a snažila se o vyřešení problémů. Žáci v reflexi své práce konstatovali, že objasňování svých myšlenek spolužákům jim pomohlo k ujasnění si řešení problému. Při vysvětlování byli nuceni více nad problémem přemýšlet. Přestože práce byla hodnocena známkou, vládla ve sledovaných skupinách obvykle dobrá nálada a občas byl slyšet i smích. Ukázalo se, že bylo kupodivu méně příležitosti k opisování. Jedním z důvodů bylo potřeba naprogramovat hodně metod, skupiny musely pracovat relativně rychle, neměly moc času komunikovat s ostatními a radit jim.

Další částí reflexe se týkala sebehodnocení dosažení cílů. Žáci opětovně ve dvojicích hodnotili kvalitu vytvořeného programu. Každou vytvořenou metodu hodnotili 0 až 3 body. Žáci měli k dispozici příklad možného řešení. Své hodnocení mohli případně konzultovat s vyučujícím, který jim například potvrdil, zda jejich řešení lišící se od vzoru je správné. Srovnání žákovského hodnocení a hodnocení vyučujícím se zásadně nelišilo. Velkým přínosem procesu hodnocení kvality programu bylo rovněž to, že při jeho vytváření žáci diskutovali o použitém řešení, ale také o procesu vzájemné spolupráce.

Potvrdilo se, že i role učitele je při kooperativním učení jiná než v běžné frontální výuce. Učitel nezprostředkovává hotové informace, ale nechává žáky, aby společným úsilím sami přicházeli na nové poznatky a získávali nové kompetence. Učitel například pomáhá s vytvářením skupin, monitoruje činnost skupin, případně pomáhá žákům tam, kde jejich schopnosti nestačí a dává žákům prostor pro reflexi [12]; [13].

4 SHRNUTÍ

Z uvedených případových studií je patrné, že kooperativní učení má svou nezastupitelnou roli a jednoznačně obohacuje vzdělávací proces, a to rovněž ve výuce informatiky. Přináší nový pohled na práci učitele i žáků. Výsledky potvrzují, že většinu žáků práce v kooperujících skupinách těší a často jim umožňuje větší rozvoj než při individualizované výuce. Důležitou roli v kooperativním učení hraje reflexe skupiny.

Výsledky pozorování a dotazníkového šetření potvrdily, že žáci ve skupinách pracovali aktivněji a byli více motivováni než při individualizované výuce. Respondenti výzkumu potvrdili, že se jim práce ve skupině zdála zábavnější a pomohla jim k lepšímu dosažení cíle. Až na výjimky účastníci kooperativního učení potvrzovali, že díky této formě dosáhli lepších výsledků než při individualizované práci. Výuka ve skupinách však přináší některé obtíže. Problém může nastat například ve skupinách, kde je velký rozdíl ve schopnostech členů. Některým žákům práce ve skupinách vyhovovala méně a kladla na ně větší nároky než samostatná práce. Mezi nevýhody kooperativního učení je skutečnost, že obvykle není možné hodnotit jednotlivce, ale pouze celou skupinu.

5 BIBLIOGRAFICKÉ ODKAZY

- [1] JOHNSON, David W. & JOHNSON, Roger T. Making cooperative learning work. *Theory Into Practice*. 2009. ISSN 0040-5841.
- [2] KASÍKOVÁ, Hana. *Kooperativní učení, kooperativní škola*. Praha: Portál. 2016. ISBN 978-80-262-0983-6.
- [3] KASÍKOVÁ, Hana. Proměna vysokoškolské výuky: zkušenost s využitím kooperativních skupin. *Studia Pedagogica*. Roč. 20. č. 2. 2015. DOI 10.5817/SP2015-2-5.
- [4] BOSÁKOVÁ, Alice. *Sociálně psychologické aspekty kooperativního učení a význam skupinové práce pro dynamiku školní třídy*. [online] <https://www.szs-ruska.cz/userdata/files/oppa/10-vyuziti-skupinove-prace-a-dynamika-skolni-tridy/10-socialne-psych-aspekty.pdf>
- [5] KASÍKOVÁ, Hana. Kooperativní učení ve výuce: teorie – výzkum – realita. *Pedagogika*. Roč. 67. č. 27. 2017. Pages 106–125. ISSN 2336-2189.
- [6] MAREŠ, Jiří. Tvorba případových studií pro výzkumné účely. *Pedagogika*. Roč. 65. č. 2. 2015. Pages 113–142. ISSN 2336-2189.
- [7] KLASSEN, Renate G. Interdisciplinary education: a case study. *European Journal of Engineering Education*. Volume 43. Pages 842–859. 2017. DOI 10.1080/03043797.2018.1442417.
- [8] ŠVARÍČEK, Roman & ŠEĐOVÁ, Klára. *Kvalitativní výzkum v pedagogických vědách*. Praha: Portál. 2007. ISBN 978-80-7367-313-0.
- [9] CHRÁSKA, Miroslav. *Metody pedagogického výzkumu*. Praha: Grada. 2007. ISBN 978-80-247-1369-4.
- [10] NÁBĚLKOVÁ, Jitka. *Využití modelu kooperativního učení pro rozvoj klíčových kompetencí žáka*. 2017. [online] https://www.pdf.upol.cz/fileadmin/userdata/PdF/VaV/2017/odborne_seminare/vyuziti_modelu_kooper_uceni.pdf.
- [11] GILLIES, Robyn M. Development in Cooperative Learning: Review in Research. *Australian Journal of Teacher Education*. Volume 41. Issue3. 2014. ISSN 1835-517X.
- [12] GILLIES, Robyn M. BOYLE, Michael. Teachers' reflections on cooperative learning: Issues of implementation. *Teaching and Teacher Education*. Volume 26. Issue 4. Pages 933–940. 2010. ISSN 0742-051X.
- [13] SLAVIN, Robert E. Cooperative Learning, Success for All, and Evidence-based Reform in education. *Éducation et didactique*. Volume 2. Number 2. 2009. ISSN 2111-4838.

Rovnaké alebo rôzne? Čo spôsobila zmena motivácie v dvoch úlohách v súťaži iBobor

Are They the Same or Are They Different? What Effect Had the Change of Motivation on Two Bebras Task

Lucia Budinská
KDMFI FMFI UK
Mlynská dolina 1
842 48 Bratislava
Slovensko

Lucia.Budinska@fmph.uniba.sk

Karolína Miková
KDMFI FMFI UK
Mlynská dolina 1
842 48 Bratislava
Slovensko

mikova@fmph.uniba.sk

ABSTRAKT

V informatickej súťaži iBobor sa každoročne objavujú nové úlohy, mnohé sú však nejakým spôsobom podobné tým minulým. Ukazuje sa, že napriek veľkej podobnosti, nemajú vždy rovnakú úspešnosť. V našom výskume nás preto zaujímalo, čo spôsobuje, že problémovo rovnaké úlohy majú rôznu úspešnosť. V článku bližšie opisujeme dve vybrané úlohy zamerané na tému grafy. Úlohy boli použité v kategóriách Drobcí (2. a 3. ročník ZŠ) a Bobríci (4. a 5. ročník ZŠ). Tieto úlohy boli zaradené do dvoch respektíve troch ročníkov súťaže, pričom im bola zmenená iba motivácia. Ukázalo sa, že motivácia úlohy ovplyvňuje úspešnosť jedného z pohlaví a taktiež sa zmenou motivácie zmenila aj voľba nesprávnych odpovedí, takzvaných distraktorov.

ABSTRACT

Every year a lot of new tasks are used in the informatics contest Bebras, but many of them are somehow like tasks used in previous years. Despite the similarities these tasks are not always of the same difficulty. Therefore, the main idea of our research was to find out what factors influence the difficulty of tasks dealing with the same problem. In this paper we discuss two chosen graph tasks, which were used in Mini (2nd and 3rd grade of primary school) and Little Beavers (4th and 5th grade of primary school). Both were used in two or three years of contest only with slight changes in task motivation. It was showed that the change of motivation is a factor of gender specific success and in choosing of wrong answers, so called distractors.

Kľúčové slová

Súťaž iBobor, vplyv motivácie, grafové úlohy, úspešnosť, rozdielnosť v úspešnosti podľa pohlavia.

Keywords

Bebras contest, motivation impact, graph tasks, success, gender specific success differences.

1 ÚVOD

Súťaž iBobor je slovenskou verziou medzinárodnej súťaže Bebras, ktorá sa zameriava na popularizáciu informatiky, rozvíjanie informatického myslenia a zoznamovanie žiakov s technológiami. Každá z krajín, ktoré v súťaži participujú (v roku 2018/19 to bolo 54 krajín¹), každoročne posieľa niekoľko úloh do medzinárodnej databázy. Tieto úlohy si krajiny vzájomne recenzujú a na spoločnom workshope sa ich snažia úpravami prepracovať do vhodných podôb, aby

¹ zdroj <https://www.bebas.org/?q=statistics>

najlepšie z nich vybrali do výslednej série. Ide o jeden pomerne veľký kolos, za ktorým je množstvo práce. Tieto úlohy potom môžu krajiny použiť vo svojich súťažiach. Slovensko patrí medzi krajiny, ktoré majú súťažné kategórie už pre žiakov na prvom stupni základných škôl (konkrétne od druhého ročníka ZŠ). Takýchto krajín však nie je veľa a preto pri zostavovaní súťažných úloh do týchto nižších kategórií často v medzinárodnej databáze chýbajú úlohy, z ktorých by sa dalo vyberať úlohy pre mladších žiakov. Úlohy si z tohto dôvodu často vytvárame sami, podľa potreby. Pri skúmaní úloh v kategórii Bobríci (v čase výskumu bola táto kategória určená pre žiakov druhého až štvrtého ročníka ZŠ), sme zistili, že až 50 % použitých úloh z rokov 2012/13 až 2016/17 bolo slovenských [1]. To znamená, že úlohy tvoríme prioritne s ohľadom na vek žiakov a ich vedomosti podľa slovenského štátneho vzdelávacieho programu [2]. Keďže vieme, aký cieľ úlohou chceme sledovať, vieme tomu prispôbiť aj ideu, resp. celkový zámer úlohy.

Ukazuje sa, že nielen cieľ a zámer úlohy sú dôležité, ale pre mladších žiakov je veľmi dôležitá motivácia alebo takzvaná rozprávka či príbeh, ktorý sa vinie témou úlohy. Vhodne zvolený príbeh vie napomôcť porozumieť zadaniu a tomu, čo má žiak robiť, ako a prečo. V nižších kategóriách sa však snažíme používať čo najmenej textu, aby sme žiakov zbytočne nezaťažovali dlhým čítaním, preto sa veľká časť motivácie prenáša do obrázku, z ktorého by mala byť zjavná a text už doplní len dôležité informácie. Typickým príkladom úloh, v ktorých je základným nositeľom informácií obrázok, s ktorým majú žiaci nejako pracovať, sú takzvané grafové úlohy. Grafom v týchto úlohách myslíme buď dátovú štruktúru graf alebo graf z teórie grafov – samozrejme, v podobe vhodnej pre danú vekovú kategóriu.

Náš predošlý výskum sa venoval kategorizácii grafových úloh [3], pri ktorej sme našli niekoľko rôznych typov úloh a pri analýze žiackych výsledkov sme zistili, že kategórie súvisia aj s úspešnosťou chlapcov, respektíve dievčat. Tu spomenieme len niekoľko základných faktorov, ktoré v našom výskume ukázali, že ovplyvňujú úspešnosť riešenia úlohy v prospech chlapcov alebo dievčat. Pri kategorizácii sme zohľadňovali okrem iného: stratégiu riešenia, dĺžku textového zadania a vizualizáciu grafu. Dievčatá boli štatisticky významne úspešnejšie v úlohách, ktoré mali stanovený postup riešenia, alebo stav, z ktorého mali začať riešiť. Tak isto boli úspešnejšie v úlohách, kde bol graf reprezentovaný pomerne komplikovanou štruktúrou, ale mali na nej spraviť jednoduché operácie, alebo mali otestovať malý počet možností. Naopak, ukázalo sa, že chlapci boli úspešnejší v úlohách, ktoré vyžadovali objavenie stratégie riešenia, alebo nebol presne daný bod, odkiaľ mali postupovať. Lepšie výsledky dosahovali pri úlohách s viac textom, alebo pri zložitých operáciách na jednoduchých, často abstraktných, grafových štruktúrach.

Pri ďalšom skúmaní sme však zistili, že dievčatá sú ovplyvňované aj vizuálnou stránkou úlohy a motiváciou, ktorá je v úlohe použitá. To potvrdzuje aj výskum Hubweisera a kolektívu [4], ktorý bol robený na úlohách z nemeckej súťaže Bebras. Dievčatá boli úspešnejšie v úlohách, ktoré zobrazovali situáciu, s ktorou sa mohli stretnúť v reálnom živote; ktoré neboli veľmi náročné a ktoré mali pekné farebné obrázky so zvieratkami, osobami a podobne. Na druhej strane, chlapci boli úspešnejší v úlohách s abstraktnými obrázkami a technickými nákresmi; radšej riešili problém pre problém (nepotrebovali relevantný príklad zo života) a častejšie to boli náročné problémy.

Pri vyučovaní žiakov v tomto veku sme videli, že často po prvom pohľade na úlohu povedali, či je „pre dievčatá“ alebo „pre chlapcov“ a podľa toho k nej pristupovali – chlapci tie „pre dievčatá“ rýchlo preleteli, dievčatá si zas pri úlohách „pre chlapcov“ verili menej ako pri tých „pre dievčatá“, ktoré riešili dlhšie. Samozrejme, toto je len naše subjektívne pozorovanie. Výskum na talianskych stredoškôlkoch, ktorí riešili úlohy zamerané na mentálne otáčanie objektov (Mental Rotation Test) [5], ale ukázal, že ak je pred testom povedané, že v ňom budú vynikať muži resp. ženy, tak u tejto skupiny stúpne úspešnosť. U opačnej klesne, v porovnaní s výsledkami bez zásahu pred testom. To naznačuje, že aj spôsob, akým úlohy prezentujeme a ako k ich riešeniu motivujeme, môže mať vplyv na úspešnosť chlapcov a dievčat.

Môže ale aj podprahové naznačenie toho, komu je úloha určená, ovplyvniť výsledky? Tento fenomén sme chceli otestovať a zistiť, či motivácia úlohy (a jej prispôsobenie dievčatám či chlapcom) dokáže

ovplyvniť jej náročnosť. A to jak celkovú, tak aj pre jedno z pohlaví. Využili sme preto dve úlohy zo súťaže, pričom jedna bola pôvodne neutrálna a druhá chlapčenská, a zmenili sme im motiváciu. Jeden typ úlohy sme použili dokonca dvakrát, aby sme mali ešte lepšie porovnanie.

2 VÝSKUMNÁ STRATÉGIA

V tomto článku opisujeme výskum, ktorého hlavným cieľom bolo zistiť, či motivačná téma úloh zo súťaže iBobor vie ovplyvniť úspešnosť žiakov v závislosti od pohlavia, prípadne súvisí aj s voľbou distraktorov. A ak áno, tak akým spôsobom.

Pri spracovaní dát, ktoré sme mali k dispozícii, čím boli v našom prípade zadania úloh (texty, obrázky, možnosti odpovedí) a výsledky riešiteľov, sme využívali v prvej fáze kvalitatívne a v druhej fáze kvantitatívne metódy analýzy dát. Najprv sme kategorizovali úlohy, pričom sme používali kvalitatívne nástroje ako otvorené kódovanie úloh viacerými výskumníkmi podľa [6], pričom komplexné výsledky kategorizácie sme publikovali v [3].

Potom, v druhej fáze výskumu sme žiakov rozdelili do rôznych skupín – na základe ročníka v škole, pohlavia, súťažnej kategórie, respektíve kombinácie predchádzajúcich. Na zisťovanie rozdielov v úspešnosti medzi jednotlivými skupinami žiakov sme používali chi-square test [7]. Pre každú skupinu a pre každú úlohu v súťaži sme zo súťažnej databázy zistili počet správnych odpovedí, počet nesprávnych odpovedí a počet súťažiacich, ktorí danú úlohu neriešili. Každá skupina tak mala tri kvalitatívne znaky, a aby sme zistili, či je početnosť týchto znakov v oboch skupinách rovnaká a ak nie, v ktorom znaku nastal rozdiel, použili sme zisťovanie z-score [7]. Keďže obe skúmané úlohy bolo vo forme výberu zo štyroch odpovedí, druhý test sme previedli na 5-tich znakoch (4 odpovede a neriešenie úlohy). Opäť sme použili z-score, tentoraz aby sme zistili, či jednotlivé skupiny súťažiacich vyberali rovnaké nesprávne odpovede, takzvané distraktory.

Hladiny významnosti z-score sme určovali podľa štatistických tabuliek a určili sme tri hranice – a to $z = 1,96$ pre $\alpha = 0,05$, $z = 2,58$ pre $\alpha = 0,01$ a $z = 3,3$ pre $\alpha = 0,001$. Hodnoty z-skóre nižšie ako 1,96 sme považovali za nevýznamný rozdiel medzi skupinami, t. j. že skupiny úlohy riešili s rovnakou úspešnosťou.

3 ÚLOHY

Ako sme písali v úvode, pre náš výskum sme vybrali dve úlohy, ktoré podávajú riešiteľom podstatnú časť informácie v obrázku a to konkrétne na obrázku grafu. V nasledujúcich kapitolách opisujeme tieto úlohy: aké mali zadanie, ako vyzeral graf, ako sa s meniacou motiváciou menili dosahované výsledky riešiteľov, ako volili distraktory, atď.

3.1 Červíček a ZOO

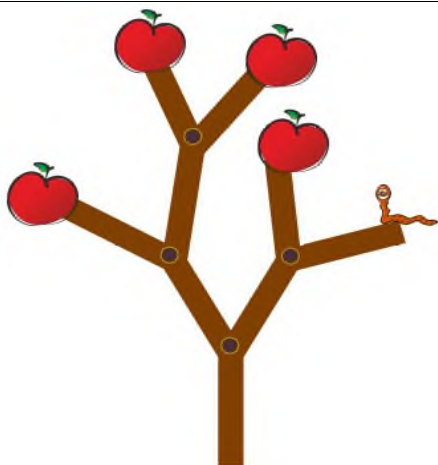
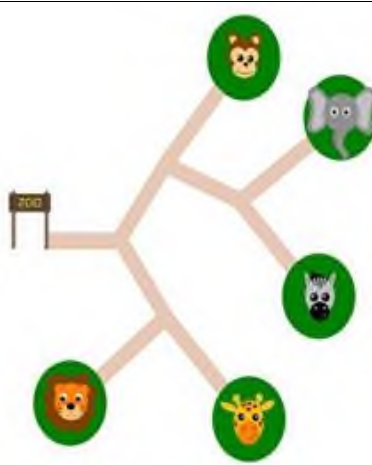
Úloha Červíček bola pôvodne úlohou zadanou v súťaži v školskom roku 2017/18. Zadanie sa nachádza na Obrázku 1 vľavo. Úloha bola zaradená v kategórii Drobci (t. j. druháci a tretiaci základných škôl), a jej úspešnosť bola 24,29 %, čo znamená, že sa zaradila k tým ťažším. Najčastejšia nesprávna odpoveď bola odpoveď A, ktorú zvolilo 38,22 % súťažiacich. V tejto verzii úlohy boli výrazne úspešnejší chlapci ($\alpha = 0,001$), dievčatá výrazne častejšie volili odpoveď A alebo neodpovedali. Rozdiel vo voľbe odpovedí B a D nebol štatisticky významný.

Nízka úspešnosť úlohy nás podnietila zamyslieť sa, či bola úloha nedostatočne vysvetlená, alebo existuje príbeh, ktorý deťom jasnejšie ozrejmi, resp. napovie, ako postupovať pri riešení. Podľa najčastejšie zvolenej (nesprávnej) odpovede v úlohe Červíček sme usúdili, že pre súťažiacich bolo pravdepodobne náročné uvedomiť si, ako sa červíček hýbe po strome a že sa rátajú všetky časti jeho cesty (aj keď sa vracia). Preto sme úlohu upravili na cestičky a beh leva po nich. Taktiež sme otočili graf. Tak vznikla úpravou motivácie z úlohy Červíček úloha ZOO. Zadanie je na Obrázku 1 vpravo.

Úlohu sme zadali v školskom roku 2018/19 opäť kategórii Drobci. Úspešnosť úlohy ZOO bola 22,58 %, čo je menej ako v úlohe Červíček, napriek tomu, že súťažili aj takí žiaci, ktorí rok predtým videli úlohu Červíček. Najčastejšou nesprávnou odpoveďou bola odpoveď A, ktorú zvolilo 31,35 %

súťažiacich. V tejto úlohe je rozdiel medzi chlapcami a dievčatami štatisticky nevýznamný a ani nevidíme výrazné rozdiely pri voľbe jednotlivých odpovedí. Štatisticky významné je však neodpovedanie dievčat.

Keď porovnáme tieto úlohy, tak už z uvedených úspešností je zrejmé, že úloha Červíček dopadla štatisticky významne lepšie ($\alpha = 0,01$), a zároveň v tejto úlohe súťažiaci výrazne viac neodpovedali. Zaujímavé je však aj rozdelenie zlých odpovedí. Hoci odpoveď A bola najčastejšie zvolenou zlou odpoveďou v oboch úlohách, v úlohe ZOO ju vybralo výrazne menej súťažiacich ako v úlohe Červíček. Naopak, odpoveď D zvolilo výrazne viac súťažiacich v úlohe ZOO. Pri voľbe odpovede B nebol rozdiel významný.

Červíček sedí na konci vetvy stromu. Prejsť každú časť vetvy mu trvá 1 minútu.	Lev Filip utiekol zo svojej kletky a chce pozdraviť svojich štyroch kamarátov. Prebehnúť časť cestičky po najbližšiu križovatku mu trvá 1 minútu.
	
Koľko minút mu bude trvať najkratšia cesta, keď chce zjesť všetky jablká? A) 4 B) 8 C) 11 D) 12	Najmenej koľko minút mu to bude trvať? A) 4 B) 8 C) 11 D) 12





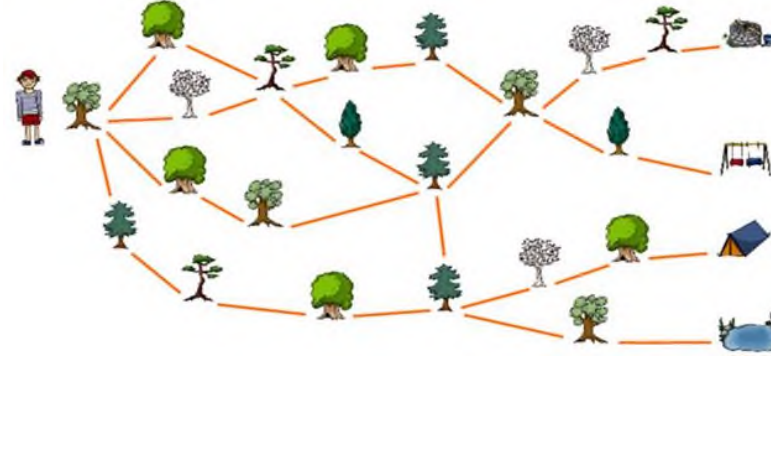

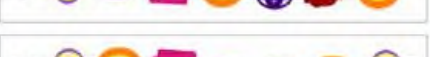


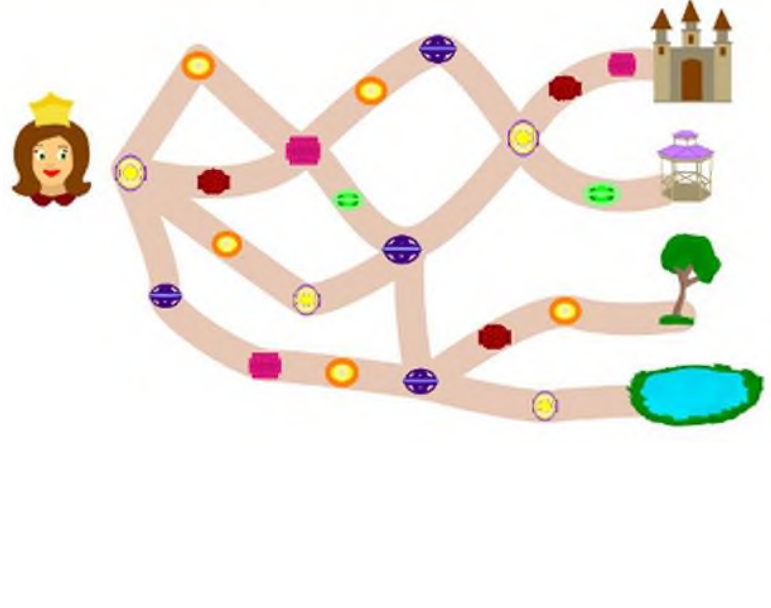




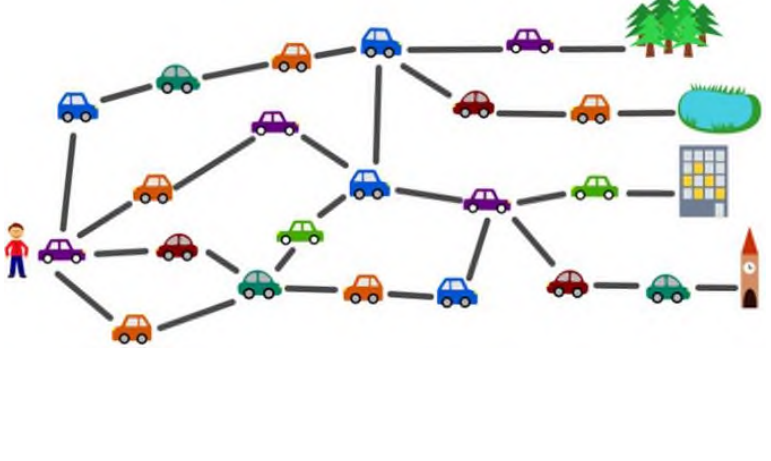
Obrázok 1: Vľavo: Kompletné zadanie úlohy Červíček (2017/18) Vpravo: Kompletné zadanie úlohy ZOO (2018/19), ktoré vzniklo úpravou úlohy Červíček.

Ak pre každé pohlavie porovnáme úspešnosť medzi úlohami, tak zistíme, že pre dievčatá je rozdiel nevýznamný. Takže úlohu riešili rovnako úspešne v oboch verziách, samozrejme s rozdielom v nesprávnych odpovediach. Pre chlapcov však vidíme významný rozdiel – úlohu Červíček riešili štatisticky významne lepšie ako úlohu ZOO. Rovnako ako pri dievčatách, aj u chlapcov sú ďalšie rozdiely v nesprávnych odpovediach, čo je ale zaujímavé tiež aj pri neriešení úlohy – výrazne častejšie chlapci neriešili úlohu Červíček. Celkovo vieme povedať, že zmena motivácie nepomohla k vyššej úspešnosti úlohy. Vyššiu voľbu distraktora D nemôžeme chápať ako lepšie porozumenie úlohe, keďže v úlohe ZOO volili súťažiaci približne s rovnakým rozdelením odpovede B a D. Tu by sme chceli len vysvetliť, že odpovede sa žiakom pri súťaži miešajú, takže nejde o voľbu prvej možnosti.

3.2 V lese, Princeznin náramok a Elektrické autá

Úloha V lese bola zadaná v školskom roku 2013/14 kategórii Bobríci, ktorá bola v tom čase určená žiakom druhého až štvrtého ročníka ZŠ. Zadanie úlohy je na Obrázku 2 hore. Úspešnosť žiakov v tejto úlohe bola 58 %, čo znamená strednú náročnosť. Najčastejšou nesprávnou odpoveďou bola odpoveď A, ktorú zvolilo 21,8 % súťažiacich. Úloha bola rovnako náročná pre chlapcov aj dievčatá

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitní a pedagogická
a nezaznamenali sme žiaden štatisticky významný rozdiel, či už z pohľadu úspešnosti pohlaví
v jednotlivých ročníkoch, alebo rozdielov medzi voľbou konkrétnej odpovede.

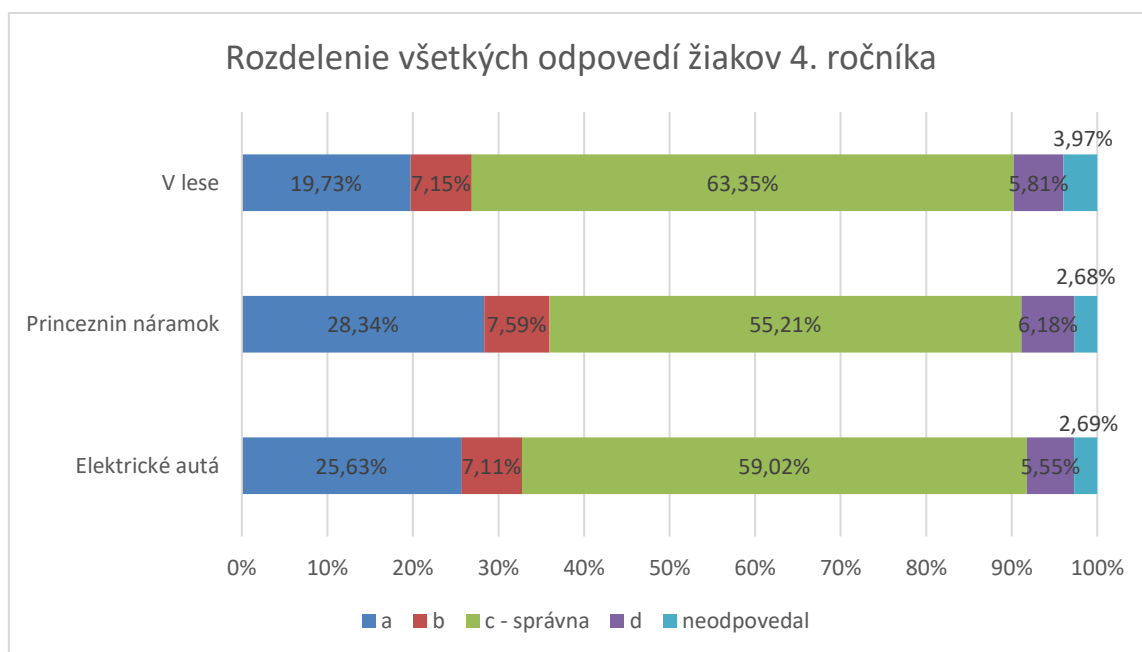
<p>V lese. Tomáš išiel na prechádzku. Prechádzal okolo rôznych stromov, až prišiel k jazierku.</p> <p>Okolo ktorých stromov postupne prechádzal?</p> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">a)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">b)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">c)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">d)</div>  </div>	
<p>Princeznin náramok. Princezná si pri prechádzke v hradnom parku všimla na zemi rozhádzané koráliky. Cestou k jazierku ich zbierala a navliekala na šnúрку.</p> <p>Ktorý náramok vytvorila?</p> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">A)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">B)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">C)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">D)</div>  </div>	
<p>Elektrické autá. Na mape mesta sú znázornené autá, ktoré sa práve nabíjajú na nabíjačkách popri ceste.</p> <p>Martin išiel do lesa. V akom poradí videl autá?</p> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">A)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">B)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">C)</div>  </div> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">D)</div>  </div>	

Obrázok 2: Hore: Kompletne znenie úlohy *V lese* (2013/14). **V strede:** Kompletne zadanie úlohy *Princeznin náramok* (2018/19). **Dole:** Kompletne zadania úlohy *Elektrické autá* (2019/20)

Úlohu V lese sme v školskom roku 2018/19 upravili tak, aby motivácia úlohy bola dievčenská. Stromy sme nahradili korálikmi a zápis cestičky predstavoval náramok, na ktorý princezná navliekala koráliky podľa toho, ako ich cestou našla, viď. Obrázok 2 v strede. Úlohu sme zaradili do kategórií Drobci (druhý a tretí ročník ZŠ) a taktiež Bobríci (štvrtý a piaty ročník ZŠ). Úspešnosť úlohy v kategórii Drobci bola 44 %, v kategórii Bobríci 59 % a celková úspešnosť pre všetky štyri ročníky bola 53 %. Najčastejšou nesprávnou odpoveďou v oboch kategóriách bola odpoveď A, ktorú zvolilo 32 % Drobčov a 26 % Bobříkov, čo je výrazne viac ako v úlohe V lese. Z pohľadu rozdielov medzi chlapcami a dievčatami je táto úloha pomerne zaujímavá. V druhom a štvrtom ročníku sme nezaznamenali rozdiel medzi chlapcami a dievčatami vo voľbe správnej odpovede. V treťom a piatom ročníku boli ale dievčatá štatisticky významne úspešnejšie ($\alpha = 0,05$). Rozdiely existovali aj medzi voľbou nesprávnych odpovedí – chlapci častejšie vyberali odpoveď D ($\alpha = 0,01$). Dokopy, pre všetkých súťažiacich, vychádza úloha ako dievčenská ($\alpha = 0,01$).

Motiváciu úlohy sme do ročníka 2019/20 ešte raz upravili. Tentoraz sme sa snažili prispôbiť motiváciu chlapcom, preto sú v grafe autíčka, pozri Obrázok 2 dole. Aby úloha vizuálne vyzerala inak ako úloha Princeznin náramok, použili sme rovnaké značenie ciest ako v úlohe V lese, a graf sme zrkadlovo otočili. Úlohu sme použili iba v kategórii Bobríci (štvrtý a piaty ročník ZŠ). V štvrtom ročníku sa úloha ukázala ako neutrálna, jediný rozdiel bol v neodpovedaní (chlapci neodpovedali viac $\alpha = 0,05$). V piatom ročníku bola úloha dievčenská ($\alpha = 0,05$) opäť s väčším neodpovedaním chlapcov. Celkovo pre oba ročníky dokopy bola úloha výrazne dievčenská s výrazným neodpovedaním chlapcov ($\alpha = 0,01$).

Na porovnanie sme vybrali žiakov štvrtého ročníka, keďže to je jediný rovnaký ročník vo všetkých troch verziách úlohy. Kompletne výsledky sú v Grafe 1. Najúspešnejší boli štvrtáci v úlohe V lese, v tejto úlohe je výraznejšie aj neodpovedanie v porovnaní so zvyšnými dvoma verziami. Výraznejšie častejšie štvrtáci volili odpoveď A v úlohách Princeznin náramok a Elektrické autá. Táto odpoveď predstavuje zápis cesty k inému cieľu, ako je zadané v úlohe. Ciele nie sú vzájomne zameniteľné, v neskorších úpravách úlohy ale možno viac podobné (les a jazierko, resp. jazierko a strom) ako v pôvodnej úlohe V lese (jazierko a stan).



Graf 1 – Odpovede žiakov 4. ročníka na jednotlivé verzie úlohy

4 DISKUSIA A ZÁVER

Naše výsledky ukazujú, že motivácia úlohy vplýva na žiacke riešenia a to i z pohľadu úspešnosti a aj voľby nesprávnej odpovede. Príbeh použitý v úlohe ovplyvňuje spôsob, akým žiaci nad úlohou rozmyšľajú a ako ju riešia, takže má priamy vplyv na ich miskoncepcie. Nepotvrdila sa ale úplne naša hypotéza, že chlapčensky ladená úloha prispieva k vyššej úspešnosti chlapcov. Predpokladáme, že existuje viacero faktorov, ktoré vplývajú na rozdiely medzi žiakmi. Na druhej strane je možné, že iba umiestnenie autíčok na mapu nespraví motiváciu chlapčenskou. Z pohľadu výsledkov Hubweisera a kolektívu [4] vyplýva, že dievčatá sa úspešnejšie v úlohách, ktoré pripomínajú reálnu situáciu, ktorú mohli zažiť – a to táto úloha spĺňa, rovnako ako aj farebný zaujímavý obrázok. Ten sa vyskytuje vo všetkých verziách úlohy. Pri úlohe ZOO sa obrázok a zadanie upravili viac na dievčenskú verziu podľa [4], ale úspešnosť dievčat nestúpila, naopak, klesla úspešnosť chlapcov, takže výsledky sa vyrovnali.

Pri porovnaní úspešnosti štvrtákov v druhej úlohe sme zistili, že pre žiakov bola najľahšia prvá verzia úlohy, t. j. úloha V lese. To však môže mať viacero dôvodov – v tomto ročníku súťaže súťažilo menej žiakov a zároveň títo žiaci mali povinnú informatiku aj v druhom ročníku základnej školy. Čo nám môže naznačovať, že učitelia žiakov vyberali, na rozdiel od aktuálneho stavu, kedy je bežné, že súťažia celé triedy. Úloha Princeznin náramok mala z týchto úloh najnižšiu úspešnosť. Vyššia úspešnosť úlohy Elektrické autá mohla byť ovplyvnená tým, že niektorí žiaci sa s touto úlohou už stretli v predchádzajúcom ročníku súťaže alebo v archíve úloh. Ale to, že by použitie podobnej úlohy automaticky zvýhodnilo žiakov, ktorí riešili v minulom roku predchádzajúcu verziu, vyvracia aj príklad úloh Červíček a ZOO, kde celkové výsledky žiakov klesli.

Ukazuje sa, že rozdiely medzi chlapcami a dievčatami sa líšia medzi jednotlivými školskými ročníkmi – sú ročníky, v ktorých je úloha neutrálna, a o ročník vyššie je už výrazne dievčenská či chlapčenská. Tento fenomén sa chystáme ďalej skúmať a zisťovať jeho príčiny.

5 POĎAKOVANIE

Výsledky tohto výskumu môžu byť publikované aj vďaka projektu VEGA 1/0602/20.

6 BIBLIOGRAFICKÉ ODKAZY

- [1] BUDINSKÁ, Lucia, MAYEROVÁ, Karolína a Michaela VESELOVSKÁ. Bebras Task Analysis in Category Little Beavers in Slovakia. In: *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2017*. s. 91–101. Springer, Cham, 2017.
- [2] ŠPU: 2015, *Inovovaný štátny vzdelávací program: Informatika. Vzdelávacia oblasť: Matematika a práca s informáciami. Príloha ISCED 2*. Bratislava: Štátny Pedagogický Ústav. [cit. 2019-12-27]. Dostupné na: http://www.statpedu.sk/sites/default/files/dokumenty/inovovany-statny-vzdelavaci-program/informatika_nsv_2014.pdf
- [3] BUDINSKÁ, Lucia, a Karolína MAYEROVÁ. Graph Tasks in Bebras Contest: What does it have to do with gender? In *Proceedings of the 6th Computer Science Education Research Conference*, s. 83–90. ACM, 2017.
- [4] HUBWIESER, Peter, HUBWIESER, Elena a Dorothee GRASWALD. How to Attract the Girls: Gender-Specific Performance and Motivation in the Bebras Challenge. In *Informatics in Schools: Improvement of Informatics Knowledge and Perception. ISSEP 2016*. s. 40–52. Springer, Cham. 2016.
- [5] MOE, Angelica a Francesca PAZZAGLIA. Following the instructions!: Effects of gender beliefs in mental rotation. In: *Learning and Individual Differences vol. 16, issue 4*. s. 369–377. 2006. ISSN 1041-6080.

- [6] ŠVARŤÍČEK, Roman a Klára ŠEĎOVÁ: *Kvalitativní výzkum v pedagogických vědách*. Praha: Portál, 2014. ISBN 978-80-7367-313-0.
- [7] WIMMER, Gejza. *Štatistické metódy v pedagogike*. Hradec Králové: GAUDEAMUS, 1993. 154 s. ISBN 80-7041-864-8.

Kategorizácia teoretických úloh algoritmickej súťaže PRASK

The Algorithmic Tasks Categorization in the Middle School Competition PRASK

Michal Anderle

UK FMFI KI

Mlynská dolina

842 48 Bratislava

Slovensko

anderle@dcs.fmph.uniba.sk

ABSTRAKT

PRASK je algoritmická súťaž určená žiakom základných škôl, ktorej teoretické úlohy sa skladajú z postupne sa graduujúcich podúloh, ktoré vedú riešiteľov k návrhu algoritmu bez toho, aby sa opierali o predchádzajúce znalosti algoritmizácie či programovacieho jazyka.

Doposiaľ bolo pre súťaž PRASK vytvorených viac ako 30 originálnych teoretických úloh. V našom prvotnom výskume sme sa rozhodli kategorizovať tieto úlohy na základe kompetencií určenými informatickým myslením a Bloomovou taxonómiou. V tomto článku prinášame výsledky tejto kategorizácie, pomocou vhodne zvolených príkladov prezentujeme často sa opakujúce vzorce a predkladáme kostru typickej teoretickej úlohy v súťaži PRASK.

ABSTRACT

Theoretical tasks from middle school algorithmic competition PRASK presents complex algorithms and data structures to pupils regardless of their previous knowledge. These tasks use a sequence of graduating subtasks that direct contestants towards the design of the final algorithm.

So far, more than 30 original theoretical tasks were created for this competition. In this paper, these tasks will be categorized based on computational thinking skills and also according to the levels of the cognitive domain of Bloom's taxonomy. Emerging patterns will be presented and described using fitting examples. Finally, the general archetype of PRASK's theoretical task will be proposed.

Kľúčové slová

súťaž PRASK, základné školy, kategorizácia úloh

Keywords

PRASK competition, middle schools, tasks categorization

1 ÚVOD

Jednou z veľkých otázok súčasného výskumu vyučovania informatiky je, akým spôsobom vieme rozvíjať informatické myslenie žiakov. Pomerne bežný je prístup prostredníctvom programovania, nás však zaujal pohľad, ktorý prezentovali Lu a Fletcher [1], v ktorom tvrdia, že programovanie by nemalo byť esenciálne pri rozvoji informatického myslenia. Zápis formálnych postupov v programovacom jazyku je totiž veľmi náročný, ak žiaci nerozumejú základným princípom, ktoré majú používať.

Alternatívu poskytujú rôzne aktivity bez priameho zapojenia programovania, medzi ktoré vieme zaradiť napríklad CS Unplugged [2], ale aj rôzne informatické súťaže. Práve tie môžu viesť k tomu,

že žiakom atraktívnou formou prezentujeme dôležité informatické koncepty. V našom výskume sme sa preto na jednu takúto súťaž rozhodli zamerať.

V roku 2015 vznikla na Slovensku algoritmická súťaž PRASK [3] určená pre žiakov druhého stupňa základných škôl (primárne ročníky 7 až 9), ktorej cieľom je rozvíjať informatické myslenie jej účastníkov bez ohľadu na to, či sú začiatočníci alebo skúsení riešitelia, a popularizovať medzi nimi informatiku.

Súťaž pozostáva zo štyroch kôl, ktoré sú zverejňované počas roka. V každom kole je 5 príkladov a žiaci majú na riešenie aspoň mesiac času. Kolá na seba nenadväzujú a je možné sa zapojiť kedykoľvek pred termínom odovzdania. Po tomto termíne organizátori opravujú a obodujú odovzdané riešenia a spolu so slovným komentárom ho pošlú späť riešiteľom. Za úlohu je možné získať aj čiastkové body. Dvakrát do roka sa potom pre najlepších riešiteľov organizujú vzdelávacie tábory.

Úlohy v súťaži PRASK sú rôzne, formou zadania aj spôsobom riešenia. Každé kolo obsahuje dve teoretické, jednu praktickú a dve programátorské úlohy. Pri programátorských úlohách žiaci odovzdávajú odladený program, ktorý sa automaticky otestuje na pripravených vstupných dátach. Vďaka tomu majú okamžitú spätnú väzbu a svoje riešenie môžu na jej základe ďalej upravovať. Pre žiakov, ktorí programovať nevedia, je určený tutoriál, ktorého riešením môžu získať body priamo do súťaže.

Praktické úlohy sa zameriavajú na konkrétne technológie či nástroje, mnohokrát sú prezentované štýlom logických úloh a obsahujú interaktívne prostredie. Prostredníctvom týchto úloh sa žiaci stretli s SQL, HTML, prácou s obrázkami či šifrovaním.

Teoretické úlohy, a práve na tie sa v tomto výskume zameriavame, najčastejšie zoznamujú žiaka so známymi dátovými štruktúrami či algoritmami. Sú však prezentované vo forme príbehu, ktorý odhaľuje všetky dôležité pravidlá a vzťahy, a potom necháva žiakov, aby sami prišli na to, ako tieto pravidlá využiť a navrhli formálny postup riešenia. Pre uľahčenie sú teoretické úlohy rozdelené do viacerých podúloh s postupne rastúcou zložitou.

Riešenia k teoretickým úlohám žiaci slovne spisujú a tieto popisy sú následne opravované. Spísané riešenia však nemajú obsahovať iba správny výsledok, ale aj postup riešenia a častokrát aj popis algoritmu, akurát v prirodzenom jazyku.

V tomto článku sme sa rozhodli bližšie pozrieť na teoretické úlohy súťaže PRASK. Ukážeme si, ako sme ich kategorizovali na základe informatických zručností a Bloomovej taxonómie a aké výsledky nám z tejto kategorizácie vyšli. Nakoniec zhrnieme všetky získané výsledky a odprezentujeme kosťu typického príkladu v tejto súťaži.

2 METODOLÓGIA

2.1 Ciele výskumu

PRASK je algoritmická súťaž pre žiakov základných škôl, ktorej cieľom je rozvíjať informatické myslenie a programátorské schopnosti jej riešiteľov. Je ťažké povedať, nakoľko je v tejto snahe úspešná. Bez ohľadu na to však majú organizátori pomerne bohaté skúsenosti s prípravou úloh a zadaní. Treba poznamenať, že jedným z dlhoročných organizátorov je aj autor tohto príspevku. Organizátori súťaže PRASK nemajú žiadne formálne didaktické vzdelanie, pri vytváraní súťažných úloh sa riadia najmä intuíciou. Bolo by preto zaujímavé sa pozrieť na to, ako vyzerajú takto vytvorené súťažné zadania a zistiť, či sa v nich napríklad nachádzajú často sa opakujúce vzorce. Tieto poznatky by totiž mohli byť prínosné pri tvorbe obdobných úloh.

Pre tento výskum sme si stanovili nasledovné ciele:

- Kategorizovať teoretické úlohy súťaže PRASK na základe dvoch zvolených kategorizácií.
- Nájsť opakujúce sa vzorce a popísať, ako vyzerá typické zadanie úlohy v súťaži PRASK.

2.2 Zvolené kategorizácie

Ako prvé sme si museli zvoliť kritériá, na základe ktorých budeme úlohy kategorizovať. Prirodzene sme sa snažili pozrieť na kategorizácie iných súťaží. Problémom ale bolo nájsť súťaž s podobnými úlohami, ktorá by vhodnú kategorizáciu mala vytvorenú. Najčastejšie boli úlohy zaradované podľa ich témy a nie zručností potrebných na ich riešenie.

Ako najvhodnejšia alternatíva sa javila súťaž Bebras. Úlohy v nej sa zameriavajú na podobné témy ako v súťaži PRASK, aj keď väčšinou idú menej do hĺbky. Navyše pre ňu existuje viacero kategorizácií, ktoré sú publikované a naďalej sa upravujú. Posledný významnejší príspevok k tejto téme predstavila v roku 2017 Dagiene s kolektívom [4]. Uvedená kategorizácia definovala dva rozmery, v ktorých má byť úloha hodnotená, prvým boli informatické koncepty a druhým informatické zručnosti.

Informatické zručnosti, ktoré sa ukázali byť pre našu prácu dôležitejšie, vychádzali z článku autorov Selby a Woollard z roku 2013 [5], ktorí sa snažili zjednotiť definíciu pojmu informatické myslenie skúmaním viacerých článkov. V tomto článku zhrnuli, že informatické myslenie je proces zameraný na riešenie problému, ktorý zahŕňa abstrakciu, dekompozíciu, algoritmické myslenie, hodnotenie a zovšeobecnenie. Kategorizácia založená na týchto zručnostiach nám prišla byť vhodná pre náš výskum.

Napriek tomu dvojrozmerná kategorizácia spomenutá vyššie mala isté obmedzenia. Pre získanie väčšieho množstva dát a tým pádom informatívnejších výsledkov sme sa rozhodli využiť ešte druhú kategorizáciu postavenú na kognitívnom obore Bloomovej taxonómie [6], ktorá sa bežne využíva na hodnotenie vzdelávacích cieľov a s ktorou sme už mali predchádzajúcu skúsenosť.

V našom výskume sme použili pôvodné názvy aj poradie jednotlivých úrovní, ktoré sa neskôr zmenili v revidovanej Bloomovej taxonómii. Dôvodom bolo, že pôvodná taxonómia lepšie zodpovedala úlohám v súťaži PRASK, ktoré sú zamerané hlavne na vytváranie vlastných riešení.

2.3 Metódy výskum

Ako prvé sme zhromaždili všetky zadania teoretických úloh zo súťaže PRASK, ktoré boli zverejnené medzi januárom 2015 (začiatok súťaže) a marcom 2019, čo bolo 31 unikátnych úloh, z ktorých sme ale 2 úlohy vylúčili, kvôli príliš špecifickému formátu zadaniu.

Po prvom prečítaní zadaní sme zistili, že celé zadania sú príliš obsiahle na to, aby sme ich mohli zmysluplne kategorizovať. Našťastie sa však všetky zadania skladali z niekoľkých podúloh, ktoré vytvárali menšie celky, ktoré boli na analýzu vhodnejšie. Dokopy sme pracovali so 135 podúlohami (nie každá úloha mala rovnaký počet podúloh, medián ich počtu bol 5).

V našom výskume sme využili ukotvenú teóriu s objavujúcim sa dizajnom [7]. Tento prístup nám pomohol vytvárať teóriu na základe prístupných dát a prispôbovať si ju aktuálnym požiadavkám. Pri analýze podúloh sme využili kvantitatívne metódy, tie však boli doplnené kvalitatívnymi, keď sme objavené teórie dodatočne verifikovali opätovnou analýzou zadaní.

3 VÝSLEDKY

V tomto článku uvádzame analýzu, na ktorú boli použité dve vyššie uvedené kategorizácie – jedna na základe článku Dagiene et al. [5] a druhá na základe Bloomovej taxonómie. Hoci oboje viedli k relevantným výsledkom, ich použitie nebolo bezproblémové. V tejto časti opíšeme vzniknuté problémy a takisto dáta získané oboma kategorizáciami.

3.1 Limitácia dvojrozmernej kategorizácie

Článok popisujúci dvojrozmernú kategorizáciu poskytoval užitočné usmernenia k tomu, ako na základe neho hodnotiť úlohy. Bolo však zjavné, že táto kategorizácia bola určená na menšie problémy, ktoré sú také časté v súťaži Bebras. Preto napríklad daná kategorizácia hovorila o tom, že jedna úloha by mala spadať do najviac troch kategórií.

Aj menšie podúlohy v súťaži PRASK však boli častokrát príliš komplexné. Zoberme si napríklad nasledovné zadanie jednej podúlohy (zadanie bolo mierne upravené pre účely tohto článku).

V predchádzajúcej podúlohe ste dostali dva rady 16 vojakov zoradených od najsilnejšieho po najslabšieho a prišli ste na spôsob, ako ich zoradiť do jednej usporiadanej postupnosti použitím čo najmenšieho počtu porovnaní.

V tejto podúlohe však máte 32 vojakov, o ktorých sile neviete nič. Navrhnete postup, ktorý ich zoradí do jedného, podľa sily usporiadaného, radu použitím čo najmenšieho počtu porovnaní.

Dvojrozmerná kategorizácia charakterizuje päť informatických zručností: abstrakciu; algoritmické myslenie; dekompozíciu; hodnotenie; zovšeobecnenie. Riešenie tohto problému jendoznačne vyžaduje zovšeobecnenie, keďže je potrebné opakovat' postup tvoriaci kratšie usporiadané postupnosti. To zároveň zahŕňa aj dekompozíciu, teda rozloženie problému na menšie časti. Hodnotenie je potrebné k určení optimálneho počtu porovnaní a samotná tvorba procesu spadá do algoritmického myslenia. A v nejakej miere žiaci potrebujú aj abstrahovať, keďže si pravdepodobne nahradia vojakov číslami z intervalu 1–32, keď si uvedomia, že potrebujú iba relatívnu hodnotu sily.

Kategorizáciu navyše komplikuje aj fakt, že jednotlivé podúlohy nemôžeme hodnotiť nezávisle, keďže častokrát závisia od predchádzajúcich podúloh v zadaní. Hoci žiaci v príklade vyššie potrebujú využiť abstrakciu, takáto abstrakcia pravdepodobne nastane už v predchádzajúcich podúlohách a do tejto by nemala byť zaradená.

Síce sa táto kategorizácia viac hodí na úlohy zo súťaže Bebras, vieme na základe nej odvodiť zaujímavé závery aj pre súťaž PRASK. Jednotlivé podúlohy sme teda hodnotili na základe najpravdepodobnejších riešení, berúc do úvahy predchádzajúce podúlohy a neobmedzujúc sa počtom troch kategórií na podúlohu. Je nám navyše jasné, že takáto kategorizácia je aspoň do nejakej miery subjektívna, vo výslednej analýze sme sa však na to snažili brať ohľad.

3.2 Kategorizácia na základe informatických zručností

Dvojrozmerná kategorizácia definuje dve oblasti, do ktorých sa úloha zaraďuje – informatické koncepty a informatické zručnosti. Čo sa týka konceptov, zo 135 podúloh, ktoré sme analyzovali, sa 21 zaradilo medzi „Dátové štruktúry a reprezentácia“ a zvyšok medzi „Algoritmy“. Zvyšné tri koncepty (procesy a hardvér; komunikácia a sieť; interakcie v spoločnosti) neboli vôbec využité a v skutočnosti sme veľmi podobný výsledok, vzhľadom na typ súťaže, aj očakávali. Tento predpoklad sa potvrdil a ďalej sme s využitými konceptami nepracovali.

Informatické zručnosti sú, ako už bolo spomenuté vyššie, abstrakcia, algoritmické myslenie, dekompozícia, hodnotenie a zovšeobecnenie. Spočítali sme výskyt týchto zručností naprieč všetkými podúlohami a takisto na základe toho, v ktorej podúlohe sa nachádzali. Úlohy majú rôzne počty podúloh, najčastejšie tri (7 úloh) alebo päť (10 úloh). Pri analýze sme sa teda pozerali najmä na celkový trend, teda či sa podúlohy na začiatku líšia od tých na konci.

V Tabuľke 1 sú uvedené výsledky tejto analýzy. Percentá sú zaokrúhlené na najbližšie celé číslo a vyjadrujú zastúpenie zručnosti medzi danými podúlohami. Pripomíname, že jedna podúloha mohla obsahovať viacero zručností. V Tabuľke 1 navyše neuvádzame údaje pre podúlohy f), g) a h), keďže mali iba 7, 3 a 2 výskytov, v časti spolu sú však zohľadnené.

Tabuľka 1: Percentuálny výskyt informatických zručností naprieč jednotlivými podúlohami

Podúloha	Počet	Abstrakcia	Alg. myslenie	Dekompozícia	Hodnotenie	Zovšeobecnenie
a	29	76 %	72 %	10 %	55 %	31 %
b	28	68 %	79 %	14 %	54 %	61 %
c	28	57 %	75 %	25 %	46 %	64 %
d	21	48 %	81 %	29 %	43 %	76 %
e	17	35 %	82 %	29 %	35 %	94 %
Spolu	135	59 %	78 %	20 %	46 %	64 %

Najviac zastúpené bolo algoritmické myslenie, najmenej dekompozícia, ktorá sa vyskytovala iba v 20 % podúloh, čo je výrazne menej oproti ostatným zručnostiam. Po opätovnej analýze zadání sa však nazdávame, že táto nezrovnalosť sa dá vysvetliť štruktúrou zadání. Podúlohy totiž rozdeľujú zadanie do menších logických častí, ktoré sú ľahšie na riešenie. To znamená, že dekompozíciu problému spravia už zadávateľia, aby sa riešiteľ mohol sústrediť na tvorbu algoritmu.

V Tabuľke 1 si môžeme všimnúť ešte dva výrazné trendy. Výskyt abstrakcie naprieč zadániami klesá a potreba zovšeobecňovania stúpa. Dôvodom je opäť koncepcia zadání. Zadanie prezentuje úlohu vo forme príbehu a prvé podúlohy dávajú žiakovi možnosť odstrániť nadbytočné detaily, vytvoriť si vlastnú reprezentáciu a sústrediť sa na dôležité spojitosti. Neskoršie podúlohy potom častokrát slúžia na to, aby riešiteľ spojil všetky nadobudnuté vedomosti a vytvoril všeobecný algoritmus, alebo ho použil v mierne upravenej verzii.

3.3 Kategorizácia na základe Bloomovej taxonómie

Kognitívny obor Bloomovej taxonómie definuje šesť úrovní: znalosť, pochopenie, aplikáciu, analýzu, syntézu a hodnotenie. Najčastejšie sa používa na určovanie vzdelávacích cieľov aktivít či úloh, a preto sa hodí do našej analýzy.

Tak ako predtým sme spočítali všetky výskyty jednotlivých úrovní naprieč podúlohami a takisto v závislosti na podúlohe, v ktorej sa vyskytovali. Z analýzy sme však vylúčili úroveň znalosti, keďže na tú sa úlohy v súťaži PRASK nezameriavajú a žiadne špeciálne vedomosti nad rámec IŠVP neboli vyžadované. Údaje v Tabuľke 2 opäť vyjadrujú zastúpenie jednotlivých zručností v rámci podúlohy, percentá sú zaokrúhlené na najbližšie celé číslo a údaje pre podúlohy f), g) a h) nie sú uvedené.

Tabuľka 2: Percentuálny výskyt úrovní Bloomovej taxonómie naprieč jednotlivými podúlohami

Podúloha	Počet	Porozumenie	Aplikácia	Analýza	Syntéza	Hodnotenie
a	29	97 %	90 %	31 %	0 %	7 %
b	28	43 %	89 %	96 %	21 %	7 %
c	28	11 %	61 %	100 %	57 %	14 %
d	21	0 %	57 %	100 %	76 %	19 %
e	17	0 %	59 %	100 %	100 %	24 %
Spolu	135	33 %	74 %	84 %	47 %	15 %

V Tabuľke 2 vidíme, že najčastejšie sa vyskytujúcou úrovňou bola úroveň analýzy, čo je najdôležitejší krok pri objavovaní nových závislostí či využití a práve to bolo častokrát podstatou úlohy. Najmenej sa vyskytuje úroveň hodnotenia, čo je najnáročnejšia úroveň Bloomovej taxonómie a vyskytovala sa najmä v úlohách, v ktorých mali riešitelia dokázať nejaké tvrdenie, či ohodnotiť efektívnosť ich riešenia. Navyše si môžeme všimnúť, že prvé podúlohy sú najmä na úrovni porozumenia a aplikácie a syntéza sa objavuje až pri neskorších podúlohách.

Navyše sa môžeme pozrieť aj na to, či sa jednotlivé úrovne vyskytujú v jednej podúlohe spoločne. Výsledky takejto analýzy sú uvedené v Tabuľke 3, pre danú množinu kategórií je uvedený percentuálny výskyt naprieč všetkými podúlohami.

Ako prvý typ podúlohy môžeme označiť podúlohu nachádzajúcu sa na úrovniach porozumenie-aplikácia, aplikácia-analýza či porozumenie-aplikácia-analýza. Tento typ podúloh sa vyskytoval na začiatku zadania, častokrát v ňom boli riešitelia vyzvaní vyriešiť konkrétnu malú inštanciu problému, pri ktorom si mali vybudovať intuitívne porozumenie úlohy. Ako príklad môžeme uviesť hľadanie najväčšieho prvku dvoch usporiadaných postupností, navrhnutie postupu hľadajúce najmenšie číslo v binárnom vyhľadávacom strome či hľadanie najväčšieho spoločného deliteľa čísel 111 a 9 pomocou odčítavacieho Euklidovho algoritmu.

Druhý typ podúloh sa dá charakterizovať úrovňami aplikácia-analýza-syntéza, analýza-syntéza či analýza-syntéza-hodnotenie. V týchto podúlohách mali riešitelia sformulovať všeobecný postup riešiaci danú úlohu. Príkladom takýchto podúloh je popis postupu na spojenie dvoch usporiadaných postupností, hľadanie prvku so zadanou hodnotu v binárnom vyhľadávacom strome či popis zrýchlenej verzie Euklidovho algoritmu.

Tabuľka 3: Najčastejšie sa vyskytujúce skupiny úrovní Bloomovej taxonómie objavujúce sa spoločne v jednej podúlohe

Porozumenie	Aplikácia	Analýza	Syntéza	Hodnotenie	Výskyt
0	1	1	1	0	24 %
0	1	1	0	0	17 %
1	1	0	0	0	16 %
1	1	1	0	0	12 %
0	0	1	1	0	12 %
0	0	1	1	1	7 %

3.4 Typický príklad v súťaži PRASK

Na základe vyššie uvedených pozorovaní sa môžeme pokúsiť definovať, ako vyzerá skladba typického príkladu v súťaži PRASK. Takýto príklad je algoritmický, snaží sa riešiteľovi prezentovať nový koncept či dátovú štruktúru a doviesť ho k zostaveniu algoritmu na riešenie. Prvé podúlohy sú určené na to, aby sa žiak zoznámil s prezentovaným konštruktom, často vyriešil konkrétnu malú verziu všeobecnej úlohy a vybuďoval si abstraktný model, s ktorým bude následne pracovať. V týchto podúlohách je žiak taktiež vedený k tomu, aby objavil závislosti a vzťahy, ktoré musí neskôr využiť. Potom je riešiteľ vyzvaný k tomu, aby sformuloval všeobecný postup riešiaci daný problém a odhadol efektívnosť tohto riešenia. Prezentované podúlohy pritom tvoria dekompozíciu pôvodnej úlohy na menšie, na seba nadväzujúce časti.

Z 29 zadania, ktoré sme analyzovali až 21 spĺňalo túto všeobecnú definíciu.

4 ZÁVER

V roku 2015 bola na Slovensku založená algoritmická súťaž PRASK, ktorá je špecifická tým, že je určená pre žiakov na druhom stupni základnej školy, ktorých sa snaží zoznámiť aj s náročnejšími konceptami teoretickej informatiky. Táto súťaž je organizovaná skupinou dobrovoľníkov, študentami stredných a vysokých škôl, ktorí nemajú formálne vzdelanie v didaktike informatiky, ovplyvňujú však množstvom skúseností so vzdelávaním talentovaných detí. Tieto ich skúsenosti môžu poskytovať cenné náhľady do problematiky vzdelávania.

V tomto príspevku sme sa zamerali na teoretické úlohy súťaže PRASK a kategorizovali sme jej zadania na základe dvoch zvolených kategorizácií. Prvá bola pôvodne navrhnutá pre súťaž Bebras a určovala vyžadované informatické zručnosti, druhá zodpovedala kognitívneho oboru Bloomovej taxonómie. Zo získaných dát sme vyvodili a pomenovali najčastejšie sa vyskytujúce vzorce a definovali ako vyzerá typická úloha tejto súťaže.

Zaujímavé bolo, že dekompozícia, ktorá je niektorými autormi pokladaná za najnáročnejšiu informatickú zručnosť [8] bola do značnej miery odstránená pomocou existencie podúloh, ktoré zložitejšiu úlohu delili na menšie, ľahšie zvládnuteľné časti. Takisto sme videli, že úlohy najskôr od žiaka vyžadovali zapojenie abstrakcie, pri ktorom si vybudoval model, s ktorým sa ďalej pracovalo a upriamili sa na kľúčové prvky úlohy a až vo vyšších podúlohách sa požadovalo zovšeobecnenie.

Výrazne bolo vidieť aj to, ako sa s postupne rastúcimi podúlohami vyžadujú vyššie úrovne Bloomovej taxonómie. Kým prvé podúlohy boli zaradené najmä na úrovne porozumenia, aplikácie poprípade analýzy, posledné podúlohy väčšinou vyžadovali od žiaka syntézu či hodnotenie. Toto všetko viedlo k pomerne jasnému definovaniu typického príkladu súťaže PRASK.

Náš príspevok poskytuje základ pre nadväzujúce výskumy. Zapojenie ďalších výskumníkov by pomohlo zredukovať subjektivitu vnesenú do kategorizácie a pomohlo potvrdiť vystupujúce vzorce. Takisto, okrem sústreďovania sa výlučne na zadania úloh, sa v budúcnosti plánujeme pozrieť aj na úspešnosť žiackych riešení, poprípade vykonať experimenty na základných či stredných školách, keďže úlohy tejto súťaže môžu byť zaujímavým východiskom pre to, akým spôsobom viesť žiakov bežných škôl k objavovaniu konceptov teoretickej informatiky.

V neposlednom rade, výsledky nášho výskumu môžu prispieť aj k ďalšiemu rozvoju súťaže PRASK, keďže organizátori, ktorí navrhujú zadania úloh, sa môžu v širšom merítku pozrieť na výsledky vlastnej práce a brať ich do úvahy pri ďalších ročníkoch.

5 BIBLIOGRAFICKÉ ODKAZY

- [1] LU, J. J., FLETCHER, G. H.: Thinking about computational thinking. In: ACM SIGCSE Bulletin, Vol. 41, No. 1, pp. 260–264. ACM (2009).
- [2] CSUnplugged. Computer Science Without a Computer, www.csunplugged.org (2010)
- [3] ANDERLE, M.: PRASK – an Algorithmic Competition for Middle Schoolers in Slovakia. In: Olympiads in Informatics, Vol. 12, pp. 147–157 (2018).
- [4] DAGIENE, V., SENTENCE, S., STUPURIENE, G.: Developing a two-dimensional categorization system for educational tasks in informatics. Informatica, 28(1), pp. 23–44 (2017).
- [5] SELBY, C., WOOLLARD, J.: Computational thinking: the developing definition (2013).
- [6] BLOOM, B.: Taxonomy of Educational Objectives: The Classification of Educational Goals. Handbook 1 Cognitive Domain McKay. New York (1956).

- [7] CRESWELL, J. W.: Educational research: Planning, conducting, and evaluating quantitative, pp. 146–166. Upper Saddle River, NJ: Prentice Hall (2002).
- [8] SELBY, C.: Relationships: computational thinking, pedagogy of programming, and Bloom's Taxonomy. In: Proceedings of the workshop in primary and secondary computing education, pp. 80–87. ACM (2015).

Úroveň informatického myslenia žiakov zapojených do Národného projektu IT Akadémia

The Level of Computer Thinking of Pupils Involved in the National IT Academy Project

Ján Guniš
UPJŠ v Košiciach,
Prírodovedecká fakulta
Jesenná 5
041 54 Košice
Slovensko
jan.gunis@upjs.sk

Ľubomír Šnajder
UPJŠ v Košiciach,
Prírodovedecká fakulta
Jesenná 5
041 54 Košice
Slovensko
lubomir.snajder@upjs.sk

Veronika Jurková
UPJŠ v Košiciach,
Prírodovedecká fakulta
Jesenná 5
041 54 Košice
Slovensko
veronika.kopцова@upjs.sk

Zuzana Tkáčová
UPJŠ v Košiciach,
Prírodovedecká fakulta
Jesenná 5
041 54 Košice
Slovensko
zuzana.tkacova1@upjs.sk

ABSTRAKT

V Národnom projekte IT Akadémia sa okrem aktualizácie obsahu, metód a foriem výučby zameriavame aj na rozvoj informatického myslenia žiakov. Pre potreby vyhodnotenia efektívnosti výučby podľa nami navrhnutých inovatívnych metodík sme vyvinuli vlastný test informatického myslenia. V príspevku uvádzame výsledky vstupného testovania žiakov základných a stredných škôl v oblasti informatického myslenia. Výsledky ukazujú na štatisticky významné rozdiely medzi žiakmi z rôznych cieľových skupín (vek, typ školy, pohlavie). Výsledky považujeme za užitočné aj na úrovni jednotlivých škôl, tried a tiež jednotlivcov. Učiteľom umožnia lepšie spoznať svojich žiakov a podľa toho zaradiť vhodné pedagogické intervencie.

ABSTRACT

In the National IT Academy project, in addition to updating the content, methods and forms of teaching, we also focus on the development of pupils' computational thinking. In order to evaluate the effectiveness of teaching according to our innovative teaching materials, we have developed our own computational thinking test. In the paper we present the results of the initial testing of primary and secondary school pupils in the field of computational thinking. The results show statistically significant differences between pupils from different target groups (age, type of school, gender). The results are useful at the level of particular schools, classes and also individuals. They enable teachers to get to know their pupils better and to include appropriate pedagogical interventions accordingly.

Kľúčové slová

informatické myslenie, hodnotenie žiakov, porovnanie populácií, koncepty informatického myslenia

Keywords

computational thinking, pupils' assessment, population comparison, computational thinking concepts

1 ÚVOD

Súčasťou informatického vzdelávania je meranie výsledkov učenia sa žiakov. Popri meraní odborných informatických poznatkov žiakov rôznymi nástrojmi hodnotenia (napr. didaktickými testami, rubrikami či škálami pre hodnotenie projektov) vyvstáva v súčasnosti potreba merať aj na predmete nezávislé schopnosti žiakov nevyhnutné pre vzdelávanie a život v 21. storočí. V našom výskume sme sa zamerali na meranie úrovne **informatického myslenia** žiakov (angl. computational thinking, ďalej IM), ktoré podľa [1] môžeme vymedziť ako efektívne riešenie problémov s pomocou počítača alebo bez neho. IM je komplexná schopnosť tvorená viacerými **konceptmi**. Viaceré významné vzdelávacie spoločnosti (napr. Computer Science Teachers Association, International Society for Technology in Education, Computing at School) navrhli vlastné rámce konceptov IM. Pre naše výskumné účely sme vybrali šesticu konceptov navrhnutých Computing at School, a to konkrétne:

- **Logika** – predpovedaj, analyzuj
(logicky vyvodzovať závery z pozorovaní a experimentov, analyzovať nejaký systém/program a predpovedať jeho správanie sa)
- **Algoritmy** – vytváraj kroky a pravidlá
(zostavovať postupy činnosti pre nejakého vykonávateľa (algoritmy, programy, scenáre, storyboardy) a tiež využívať, upravovať, vylepšovať vytvorené postupy (algoritmy, programy, scenáre, storyboardy))
- **Dekompozícia** – rozdeľ na časti
(rozdeľovať problém na ľahšie podproblémy, riešenie ktorých bude využiteľné pri riešení pôvodného problému)
- **Hľadanie vzorov** – rozpoznaj a využívaj podobnosti
(rozpoznávať a určovať v systéme rovnaké/podobné časti/vlastnosti/pravidlá a využívať nájdené vzory pri rôznych činnostiach/riešení problému)
- **Abstrakcia** – vyber podstatné, odhliadni od menej podstatného
(určovať, ktoré detaily/prvky/vlastnosti/vzťahy systému sú v danej situácii podstatné a ktoré môžeme zanedbať)
- **Vyhodnotenie** – rob rozhodnutia
(určovať relevantné kritériá hodnotenia a podľa nich vyhodnocovať projekt/program/algoritmus)

Popri rôznych rámcoch IM existujú aj rôzne **spôsoby merania IM**:

- Rubriky pre hodnotenie žiackych projektov [2]
- Analýza projektového portfólia žiakov [3]
- Interview k vlastným projektom [3]
- Posúdenie návrhových scenárov pripravených cudzích projektov [3]
- Testy [4]
- Kompetenčne zamerané meranie [5]

V národnom projekte IT Akadémie (skr. NPITA) [6] vyvíjame obsahovo aj metodicky inovatívne metodiky pre rôzne témy školskej informatiky, ktoré majú zároveň ambíciu rozvíjať IM. Pre vyhodnotenie efektívnosti týchto metodík sme potrebovali použiť nejaký nástroj na meranie IM. Po analýze existujúcich nástrojov sme sa rozhodli vzhľadom na špecifiká NPITA navrhnuť vlastný test IM.

2 TEST INFORMATICKÉHO MYSLENIA

Na začiatku vývoja testu IM sme si stanovili niekoľko požiadaviek [7], medzi ktoré patria hlavne: nezávislosť od konkrétnych predmetových poznatkov, záber pre všetky vekové skupiny žiakov (2. stupeň ZŠ a SŠ), administrovateľnosť testu v rámci jednej vyučovacej hodiny a možnosť

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitná a pedagogická strojového vyhodnotenia riešenia testu. Test IM je testom relatívneho výkonu žiaka. Úspešnosť žiaka vyhodnocujeme v rámci jeho kategórie (vek, typ školy) a vyjadrujeme ho percentilom. Test IM sme vyvíjali vo viacerých na seba nadväzujúcich etapách. Z banky 30 kandidátskych úloh sme zostavili tlačnú beta-verziu testu s 2 uzavretými a 10 otvorenými úlohami, ktorá bola v novembri 2017 administrovaná so skupinou 26 študentov rozširujúceho štúdia informatiky. Na základe výsledkov tohto testovania sme zostavili verziu 1.0 testu IM s 8 uzavretými a 3 otvorenými úlohami, ktorý sme administrovali v decembri 2017 až januári 2018 na 862 žiakoch z 11 pilotných škôl (4 ZŠ, 4 SOŠ a 3 gymnázií). Napokon sme po analýze výsledkov testu 1.0 zostavili verziu 1.1 testu IM s 10 uzavretými a 2 otvorenými úlohami, ktorý sme administrovali v marci 2018 až v decembri 2019 na 31 771 žiakoch zo škôl zapojených do NPITA.

Tento iteratívny vývoj testu IM bol nevyhnutý pre zabezpečenie kvality výslednej verzie testu IM. Medzi najpodstatnejšie úpravy verzie 1.1 testu IM oproti jeho verzii 1.0 patria:

- K samotnému testu sme zaradili aj krátky postojový dotazník k programovaniu na zistenie vzťahu medzi úrovňou IM žiakov a ich postojom k programovaniu.
- Do testu sme doplnili 1 uzavretú úlohu pre testovanie konceptu algoritmy, čím sme v teste lepšie vyvážili zastúpenie jednotlivých konceptov IM.
- Pre každú testovaciu úlohu sme doplnili aj možnosť odpovede „neviem“, aby sme eliminovali možnosť náhodných odpovedí.
- Pri vybraných úlohách sme upravili distraktory pre mapovanie viacerých žiackych miskoncepcií.
- V zadaní testovacích úloh sme farebne zvýraznili podstatné informácie a zredukovali synonymá (eliminujeme vplyv nižšej úrovne čitateľskej gramotnosti žiakov).
- Do zadania niektorých testovacích úloh s nižšou úspešnosťou sme doplnili ilustračný príklad, aby sme pomohli žiakom lepšie pochopiť zadanie.
- Redukovali sme množstvo zobrazovaných informácií v jednotlivých testovacích úlohách. Pôvodne použitý prepínač (HTML element radio) sme nahradili rozbaľovaním zoznamom (HTML element select).
- Testovaciu úlohu nezobrazujeme naraz, ale jednotlivo na jednej samostatnej rolovateľnej obrazovke, čím eliminujeme nechcenú spoluprácu žiakov.
- Na vyhodnotenie aktuálnej verzie testu sme namiesto tabuľkového kalkulátora použili Python skripty.

3 METODIKA VYHODNOCOVANIA TESTU INFORMATICKÉHO MYSLENIA

3.1 Kategorizácia žiakov

Keďže test IM je testom relatívneho výkonu, vyhodnocovali sme úspešnosť riešenia testu pre každú vekovú kategóriu zvlášť. Naším pôvodným zámerom bolo kategorizovať žiakov podľa školských ročníkov. Vzhľadom na voľnosť pomenovania tried (napr. Dúha, Lúka, Vesmír) to nebolo možné. Zvolili sme preto kategorizáciu podľa veku. Pri tejto kategorizácii sa môže stať, že dvaja spolužiaci z rovnakej triedy budú zaradení v rôznych kategóriách. Vek žiaka je však údaj, ktorý vieme určiť presne.

Rovnako sme odlišovali aj typ školy, ktorý žiak v čase riešenia testu navštevoval. Keďže nevieme odlíšiť či žiak navštevuje osemročné alebo štvorročné gymnázium (obidva typy škôl majú rovnaký číselník), rozdelili sme gymnaziálnych žiakov podľa veku. Kategóriu gymnázium (10–14) zastupujú prevažne žiaci osemročných gymnázií. V kategórii gymnázium (15–19) sú žiaci osemročných aj štvorročných gymnázií.

3.2 Typy úloh

Samotný test obsahuje 12 úloh testujúcich koncepty IM a 2 postojové otázky. Úlohu testujúce koncepty IM sú nasledovných typov:

- **Typ 1:** Úloha, kde musí žiak vyhodnotiť rôzne ponúkané odpovede. Každú z odpovedí žiak hodnotí jednou z možností správne/nesprávne/neviem, prípadne iné ekvivalentné formulácie podľa kontextu úlohy. Test obsahuje 7 úloh tohto typu.
- **Typ 2:** Úloha, kde má žiak pre vybrané objekty vystupujúce v úlohe vybrať jednu z ponúkaných možností alebo môže vybrať možnosť neviem. V teste sú 3 úlohy typu 2.
- **Typ 3:** Otvorená úloha v ktorej žiak odpoveď vytvára. V teste sme použili 2 úlohy tohto typu.

V odpovediach sme žiakom dali možnosť odpovede „neviem“¹. Takáto odpoveď bola tiež hodnotená počtom bodov 0. Pri štatistickom vyhodnotení však vieme odlíšiť nesprávnu odpoveď od možnosti, keď žiak sám usúdil, že odpovedať nevie.

Rozdelenie úloh podľa jednotlivých typov a ich mapovanie na koncepty [1] informatického myslenia je uvedené v tabuľke (Tabuľka 1).

Tabuľka 1: Mapovanie úloh na koncepty informatického myslenia a typ úlohy

atribút/úloha		1	2	3	4	5	6	7	8	9	10	11	12
Koncepty informatického myslenia	logika					✓					✓		
	algoritmy								✓	✓			✓
	vzory						✓	✓					
	dekompozícia	✓	✓							✓			
	vyhodnotenie	✓						✓	✓			✓	
	abstrakcia			✓	✓								
Typ úlohy	typ 1	✓	✓	✓	✓		✓			✓	✓		
	typ 2					✓						✓	✓
	typ 3							✓	✓				

3.3 Vyhodnocovanie riešení úloh

Počet respondentov predstavuje rádovo desať tisíce. Ručné vyhodnocovanie preto nepripadalo do úvahy. Riešenia úloh sme preto vyhodnocovali pomocou skriptov (ako programovací jazyk sme použili jazyk Python). Pre vyhodnotenie každej úlohy (resp. odpovedí v danej úlohe) sme vytvorili samostatnú funkciu, ktorá odpovede žiaka v úlohe analyzovala a vrátila bodové hodnotenie danej úlohy, resp. bodové hodnotenie konceptov IM, ktoré sme v úlohe testovali. Bodový zisk žiaka za každú úlohu bol normovaný na maximálny zisk 1 bodu. Každá úloha v teste k celkovému hodnoteniu prispela rovnakou mierou. V tomto prípade mal test 12 samostatne hodnotených položiek.

Pri ďalšej analýze odpovedí žiakov sme zvlášť vyhodnocovali aj úspešnosť jednotlivých konceptov IM. Bodové zisky žiakov z jednotlivých konceptov v úlohách sme sčítali pre každý koncept zvlášť a znormovali na maximálny zisk z daného konceptu na 1 bod. V tomto prípade sme test hodnotili ako 6položkový.

Úlohy **typu 1** sme vyhodnocovali nasledovne. Každú odpoveď žiaka sme vyhodnocovali samostatne. Ak žiak označil všetky správne odpovede ako správne a všetky nesprávne odpovede ako nesprávne získal plný počet bodov. Ak úloha mapovala viac konceptov IM, za nesprávne označenie odpovede sme žiakovi nepriznali body len v tom koncepte IM, v ktorom týmto nesprávnym označením zlyhal. Ak sa v úlohe dali označiť viaceré odpovede ako správne, ale vzájomne sa vylučovali, tak za ich súčasné označenie získal žiak celkové hodnotenie 0.

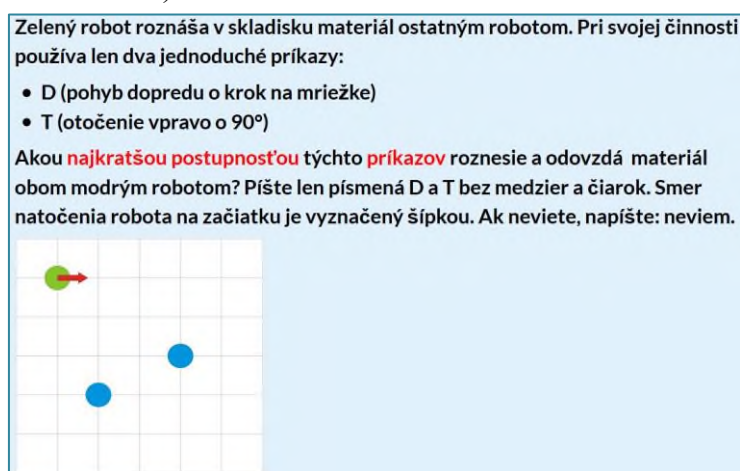
¹ Ako odpoveď neviem sme akceptovali aj rôzne modifikácie tohoto slova, napr. „neveim“, „nevim“, „neveimto“.

Úlohy **typu 2** sme vyhodnocovali analýzou celkovej situácie vyplývajúcej zo žiackeho výberu možností pre jednotlivé objekty. Čím viac sa žiacke priradenie priblížilo k očakávanému, tým viac bodov žiak dostal. Ak výsledné priradenie možností pre jednotlivé objekty bolo konfliktné, bodový zisk žiaka sme krátili.

Úlohy **typu 3** sme vyhodnocovali interpretáciou žiackeho riešenia, pričom sme sledovali, či sa podarilo splniť požadované ciele zadané v úlohe. Na základe množstva dosiahnutých cieľov sme navyšovali zisk žiaka. Keďže išlo o otvorenú úlohu, akceptovali sme s nejakým krátením bodového zisku žiaka aj odpovede, ktoré nespĺňali požadované formálne parametre.

3.4 Ukážka vyhodnotenia a analýza riešenia úlohy

Pre lepšiu predstavu uvádzame konkrétnu úlohu (v teste úloha číslo 8) a postup pri jej vyhodnocovaní. Úlohou žiakov v tejto úlohe bolo nájsť a zapísať algoritmus pre robota tak, aby splnil požadovanú úlohu (Obrázok 1). Z výsledkov pilotného overovania testu sme usúdili, že požiadavku na efektívnosť riešenia žiaci dostatočne nereflektovali. Podobne aj očakávanú formu odpovede, postupnosť znakov D a T. Tieto prvky sme v zadaní úlohy vo výslednej verzii testu zvýraznili (červená farba) alebo doplnili (požiadavka len na D a T).



Obrázok 1: Ukážka zadania úlohy v teste informatického myslenia

Táto úloha je typu 3. V riešení úlohy sme sledovali dva koncepty IM, ktoré sa na celkovom hodnotení riešenia podieľali rovnakou mierou:

- **algoritmy** – vytvorenie algoritmu, realizáciou ktorého sa zelený robot postupne dostane na pozície oboch modrých robotov,
- **vyhodnotenie** – postupnosť príkazov by mala byť čo najkratšia.

Pre vyhodnotenie odpovede žiaka sme naprogramovali jednoduchý interpret, ktorý sa pokúsil realizovať žiakov algoritmus a sledoval pritom splnenie požiadaviek. Napriek tomu, že inštrukcia pre žiaka znie: „Píšte len písmená D a T bez medzier a čiarok“, ide o otvorenú úlohu.

Pri vyhodnotení sme sa rozhodli akceptovať a vyhodnotiť aj odpovede, ktoré nie celkom spĺňali požiadavku na formu odpovede. Ak odpoveď žiaka obsahovala niektoré zo znakov: `0`, `1`, `2`, `3`, `4`, `5` alebo `biely znak`, tak v odpovedi sme malé písmená nahradili veľkými a ostatné z uvedených znakov sme odstránili. Za takúto formálnu chybu sme celkový bodový zisk žiaka z úlohy násobili koeficientom 0,9. Ak odpoveď obsahovala nejaké iné znaky, vyhodnotili sme ju ako nesprávnu s počtom bodov 0.

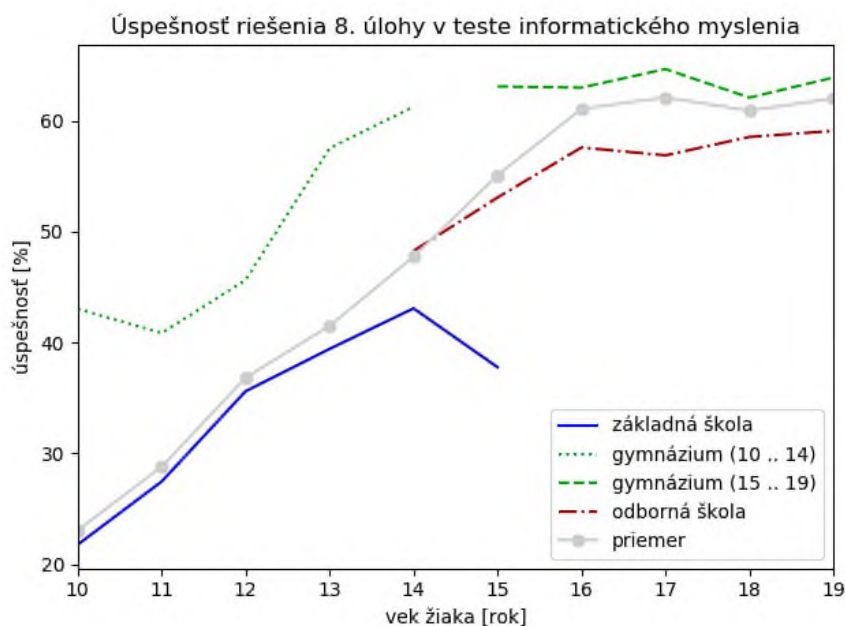
Koncept algoritmy sme hodnotili nasledovne:

- Ak sa zelený robot dostal na pozíciu jedného z modrých robotov, žiak získal 0,25 bodu.
- Ak sa robot dostal aj na pozíciu druhého modrého robota, žiak získal ďalších 0,25 bodu.

Koncept vyhodnotenie sme hodnotili nasledovne:

- Ak sa zelený robot zo štartovacej pozície dostal k modrému robotu (nerozlišujeme ku ktorému) najkratšou možnou postupnosťou príkazov, žiak získal 0,1 bodu.
- Ak sa zelený robot od navštíveného modrého robota dostal k druhému modrému robotu najkratšou možnou postupnosťou príkazov, žiak získal 0,1 bodu.
- Ak zelený robot po návšteve druhého modrého robota už nevykonával žiaden príkaz navyše, žiak získal 0,1 bodu.
- Ak zelený robot navštívil modré roboty v správnom poradí a použitím najmenšieho možného počtu príkazov, žiak získal 0,2 bodu.

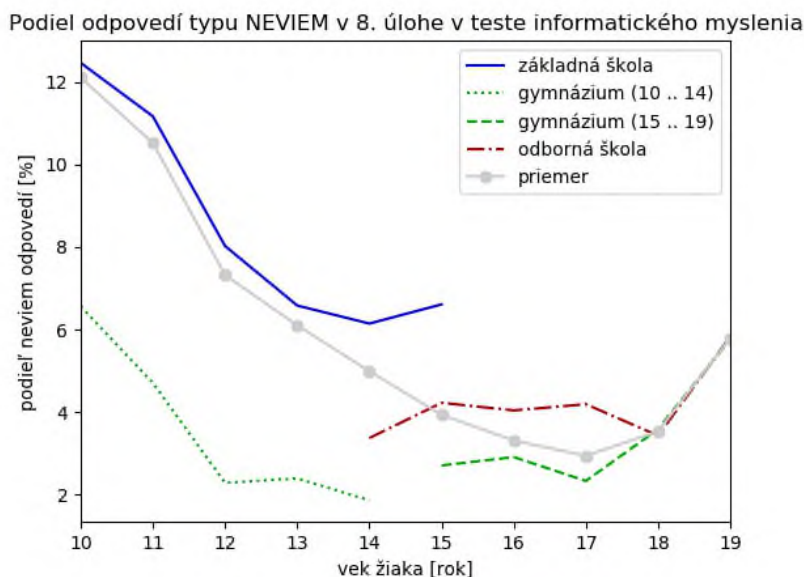
Úspešnosť riešenia tejto úlohy sme vyhodnotili pre každú vekovú kategóriu žiakov a typ školy, ktorú žiaci navštevujú (Obrázok 2).



Obrázok 2: Výsledky riešenia úlohy 8

Z grafu vidno, že úspešnosť riešenia úlohy rastie s vekom žiaka. Rovnako vidno ako osemročné gymnázia odoberajú „lepších“ žiakov zo ZŠ. Zaujímavá je aj zmena smernice úspešnosti gymnázií po nástupe žiakov ZŠ na štvorročné gymnázia. Výrazne záporná smernica úspešnosti sa objavuje na ZŠ medzi 14 a 15 vekom žiakom. Zdôvodniť to „uvoľnenosťou“ žiakov v 9. ročníku ZŠ by bolo bez ďalšieho skúmania unáhleným záverom.

Ak sa pozrieme na zastúpenie odpovedí typu NEVIEM (Obrázok 3), ich počet s vekom klesá. Podobný trend sme očakávali. Zaujal nás nárast od veku 17 rokov. Nepredpokladáme, že to súvisí so skutočnou vedomostnou úrovňou žiakov. Skôr si myslíme, že odpoveď neviem si žiaci vybrali ako „skratku“ pre rýchle ukončenie testu.



Obrázok 3: Podiel odpovedí typu NEVIEM v úlohe 8

4 VÝSLEDKY TESTU INFORMATICKÉHO MYSLENIA

4.1 Výskumná vzorka

Test IM sme administrovali na populácii 31 771 žiakov ZŠ a SŠ. Z tohto počtu žiakov test ukončilo a odoslalo na hodnotenie 28 617. Aby sme z nameraných dát dostali čo najmenej skreslené výsledky, očistili sme dáta o extrémne hodnoty. Pomocou Dixonovho testu extrémnych hodnôt sme vylúčili odpovede účastníkov s celkovým časom administrácie testu menším ako 10 minút a väčším ako 61 minút. Týmto sa znížil počet vyhodnocovaných testov na 22 926. Celkové rozloženie výskumnej vzorky uvádzame v nasledujúcej tabuľke (Tabuľka 2).

Tabuľka 2: Absolútne početnosti jednotlivých skupín žiakov

typ školy\vek	10	11	12	13	14	15	16	17	18	19	spolu
základná škola	939	2302	2505	2583	1725	575	0	0	0	0	10629
gymnázium (10–14)	61	255	350	334	591	0	0	0	0	0	1591
gymnázium (15–19)	0	0	0	0	0	1441	2337	1758	1116	295	6947
odborná škola	0	0	0	0	89	781	1287	859	554	189	3759
spolu	1010	2568	2867	2930	2419	2812	3640	2634	1688	503	22926

4.2 Výsledky testu IM

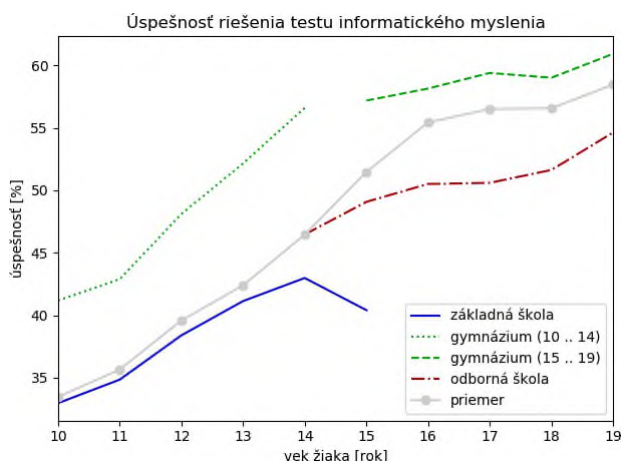
Pri vyhodnotení testu sme sa zamerali na porovnanie výsledku testu pre rôzne kategórie žiakov. Zaujímala nás rozdielnosť výsledkov testu vzhľadom na vek, pohlavie a typ školy pre jednotlivé úlohy testu, jednotlivé koncepty IM a test ako celok.

Na štatistické vyhodnocovanie (pomocou programu SPSS) boli použité neparametrické testy z dôvodu, že dáta nepodliehajú normálnemu rozdeleniu. Mann-Whitney test bol použitý pre porovnania dvoch kategórií (pohlavie) a Kruskal Wallis test pre porovnanie troch a viac kategórií (typ školy).

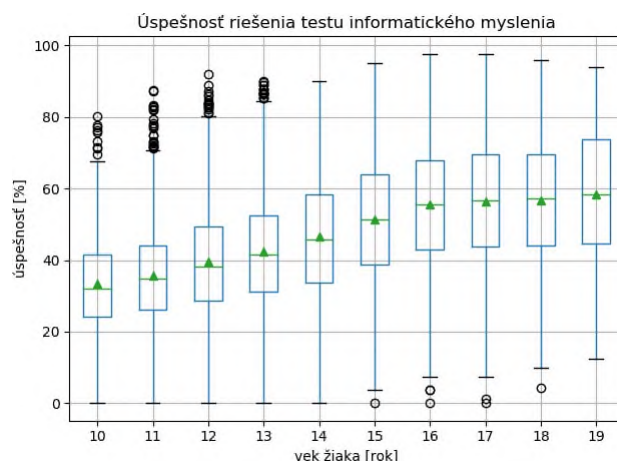
Úspešnosť riešenia každej úlohy testu v celej populácii bola v akceptovateľnom rozmedzí 25 % až 65 %.

Z pohľadu celkovej úspešnosti sa štatisticky významná rozdielnosť na hladine významnosti $\alpha = 0,01$ preukázala pre pohlavie v prospech chlapcov a pre typ školy v poradí od najslabšej ZŠ, SOŠ a GYM.

Na základe Spearmanovho korelačného koeficientu na hladine významnosti $\alpha=0,01$ bola preukázaná mierna priama závislosť medzi vekom žiakov a úspešnosťou testu s hodnotou 0,441, čo potvrdzuje zistenie, že žiaci ZŠ dosahujú nižšie skóre ako žiaci SŠ. Táto závislosť ale nie je vysoká, čo môže byť odôvodnené rozdielnosťou skóre pre SOŠ a GYM. Závislosti medzi vekom a úspešnosťou pre jednotlivé typy škôl sú nasledovné: ZŠ 0,208; SOŠ 0,064; GYM(15–19) 0,055; GYM(10–14) 0,326. Na obrázku (Obrázok 4) je znázornená priemerná úspešnosť žiakov podľa veku a typu školy. Na obrázku (Obrázok 5) je znázornená charakteristika polohy a variability výsledkov žiakov pre jednotlivé vekové kategórie.



Obrázok 4: Priemerná úspešnosť žiakov podľa veku a typu školy



Obrázok 5: Charakteristiky polohy a variability úspešnosti žiakov podľa veku

Na hladine významnosti $\alpha=0,01$ sa preukázala štatisticky významná rozdielnosť v pohlaví v prospech chlapcov pre úlohy 1, 5, 6, 8, 10, 11; v prospech dievčat pre úlohy 3, 7, 9, 12. Úlohy 2, 4 sú vzhľadom na pohlavie vyrovnané.

Podobne boli otestované jednotlivé koncepty IM, kde sa štatisticky významná rozdielnosť medzi pohlaviami preukázala pre koncepty logika a vyhodnotenie v prospech chlapcov a abstrakcia v prospech dievčat. Ostatné koncepty sú vzhľadom na pohlavie vyrovnané.

Pri konceptoch bola navyše otestovaná rozdielnosť medzi typom školy. Pre všetky koncepty sa preukázala štatisticky významná rozdielnosť medzi typom školy v poradí od najslabšej ZŠ, SOS, GYM.

Zisťovali sme tiež závislosti testovacích úloh a závislosti konceptov IM. Pre tieto kvantitatívne dáta sme použili Pearsonov korelačný koeficient. Na hladine významnosti $\alpha=0,01$ bola otestovaná štatistická významnosť týchto korelácií. Štatisticky významná korelácia sa preukázala medzi všetkými úlohami v rozmedzí 0,110–0,329. Najvyššia korelácia je medzi úlohami 5 a 8, a 7 a 8. V prípade konceptov sa medzi všetkými preukázala štatisticky významná korelácia v rozmedzí 0,225–0,643. Najvyššia závislosť (okolo 0,60) je medzi konceptami algoritmy a vyhodnotenie, algoritmy a dekompozícia, vyhodnotenie a vzory, vyhodnotenie a dekompozícia.

Na výpočet spoľahlivosti testu – reliability sme použili Cronbachov koeficient alfa. Po vyhodnotení testu po úlohách (12 položiek) sme dostali uspokojivé výsledky reliability pre jednotlivé typy škôl a pohlavia. Výsledky uvádzame v tabuľke (Tabuľka 3).

Tabuľka 3 Reliabilita testu z pohľadu úloh pre jednotlivé typy škôl a pohlavie

cieľová skupina	všetci	chlapci	dievčatá	ZŠ	SOŠ	GYM
Cronbachov koeficient alfa	0,756	0,775	0,738	0,673	0,701	0,710

Ešte vyššiu hodnotu Cronbachovho koeficientu alfa sme dostali pri vyhodnotení testu po konceptoch IM (6 položiek), a to 0,812.

5 DISKUSIA

Pri tvorbe testu sme uvažovali vytvoriť rôzne testy pre jednotlivé vekové kategórie. Z praktických dôvodov sme vytvorili jeden test IM. Žiakov vždy vyhodnocujeme v danej kategórii v závislosti od veku alebo typu školy.

Náš test ukázal štatisticky významný rozdiel v úspešnosti riešenia úloh medzi chlapcami a dievčatami v prospech chlapcov. Meraním podobnej schopnosti sa zaoberala aj PISA 2012. Podľa [8] „Schopnosť riešenia problémov znamená, že jednotliviec využíva svoje kognitívne procesy na porozumenie a vyriešenie problémovej situácie, kde spôsob riešenia nie je okamžite jednoznačný.“ Z Národnej správy PISA 2012 vyplýva, že chlapci na Slovensku dosahovali štatisticky významne lepšie výsledky ako dievčatá (v priemere o 22 bodov). Najlepšie výsledky v riešení problémov dosahovali žiaci 8-ročných a 4-ročných gymnázií. Rovnaké závery konštatujeme aj na základe výsledkov testu IM. Pre úplnosť uvádzame, že testovanie PISA sa týka len 15 ročných žiakov. Do nášho testu sme zahrnuli žiakov vo veku 10 až 19 rokov.

V roku 2018 IEA (International Association for the Evaluation of Educational Achievement) uskutočnila druhý cyklus štúdie ICILS (International Computer and Information Literacy Study), v ktorej popri hodnotení počítačovej a informačnej gramotnosti žiakov doplnili aj hodnotenie IM [9]. Do tejto unikátnej medzinárodnej štúdie sa zapojilo 46000 žiakov zo 14 krajín a regiónov (mimo SR a ČR). V tejto štúdii v teste počítačovej a informačnej gramotnosti dosiahli dievčatá (so skóre 505 bodov) lepšie výsledky ako chlapci (so skóre 488 bodov). Ale v teste IM, podobne ako v našom teste, dosiahli chlapci (so skóre 502 bodov) lepšie výsledky ako dievčatá (so skóre 498 bodov).

Ak v našom teste poskytneme žiakom možnosť odpovede „neviem“, tak na jednej strane nenútime žiaka odpovedať na otázku na ktorú odpovedať nevie (náhodná odpoveď), ale na druhej strane poskytujeme žiakom príležitosť „rýchlo“ sa zbaviť povinnosti riešenia testu. Koľko žiakov odpovedalo neviem práve z druhého dôvodu nevieme zistiť. Z výsledkov (Obrázok 3) však môžeme usudzovať, že najmä vyššie vekové kategórie odpoveď neviem vyberali z tohto dôvodu. Túto možnosť sme poskytli aj z dôvodu, že systém sme nastavili tak, aby odpoveď bola povinná.

Pre ďalší výskum je zaujímavé zistiť skutočný dôvod poklesu úspešnosti na konci ZŠ a nárast počtu odpovedí typu NEVIEM na konci SŠ. V rámci projektu NPITA boli niektoré školy vybavené laboratórnym vybavením pre prírodné vedy, matematiku a informatiku. Toto vybavenie získali vybrané školy na základe hodnotenia ich predchádzajúcich aktivít a záujmu o zapojenie sa do aktivít projektu. Predpokladali sme, že aktívni učitelia z týchto škôl svojou predchádzajúcou aktivitou ovplyvnili aj úroveň žiakov a teda, že žiaci z týchto škôl budú v teste úspešnejší. Tento predpoklad sa však na základe štatistického vyhodnotenia nepreukázal. Pre ďalšie skúmanie bude zaujímavé zistiť, ako sa prítomnosť dodaného laboratórneho vybavenia podpíše pod úspešnosť žiakov v teste IM na konci projektu.

Pomerne veľké množstvo testov, ktoré sme vyradili z hodnotenia je zrejme spôsobené procesom administrácie cez viacero stupňov (autori testu → garant aktivity v NPITA → koordinátor na škole → učiteľ → žiak) a tiež procesom spracovania výsledkov cez viacero stupňov (odpovede žiakov + dáta z informačného systému školstva → vyhodnotenie riešenia úloh → štatistické spracovanie).

Zaujímavé bude zistiť, ako sa líšia žiaci medzi sebou na základe porovnania výsledkov v jednotlivých konceptoch IM. Napríklad u koľkých žiakov je dominantný 1 či 2 koncepty voči ostatným konceptom. Koľkí žiaci majú dobre vybalansované všetky koncepty IM?

Zatiaľ nevieme presne povedať, do akej miery k rozvoju jednotlivých konceptov IM prispieva vek (vývoj psychiky) a do akej miery samotné vzdelávanie.

6 ZÁVER

Výsledky testu IM na vzorke 22 296 žiakov ZŠ a SŠ zapojených do NPITA ukázali, že úspešnosť riešenia testu je závislá od veku žiaka (rastie s vekom), typu školy (v poradí ZŠ, SOŠ, GYMN) aj pohlavia (v prospech chlapcov). Zaujímavé zistenie je, že žiaci GYM(10–14) dosahujú porovnateľnú úroveň ako žiaci SOŠ, ktorí sú o 4 až 5 rokov starší.

Preto porovnávať žiakov navzájom má zmysel len v rámci vybranej vekovej kategórie a typu školy a navyše len relatívne pomocou percentilov.

Štatistické hodnotenie ukázalo aj na uspokojivú reliabilitu testu vzhľadom k 6 konceptom IM. Priemerný čas administrovania testu (cca 27 minút) ukazuje na jeho praktickú použiteľnosť v rámci jednej vyučovacej hodiny.

Na jar 2020 budeme administrovať retest IM a po jeho štatistickom vyhodnotení spolu s testom IM vyhodnotíme mieru dopadu našich inovatívnych metódik.

Výsledky testu (percentil) poskytujeme učiteľom a žiakom až po spracovaní všetkých v danom čase dostupných dát.

7 POĎAKOVANIE

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

8 BIBLIOGRAFICKÉ ODKAZY

- [1] BCS Barefoot: Computational Thinking Concepts and Approaches [online]. [vid. 19. 12. 2019]. Dostupné na: <https://www.barefootcomputing.org/concept-approaches/computational-thinking-concepts-and-approaches>.
- [2] SHERMAN, Mark and Fred MARTIN. The assessment of mobile computational thinking. *Journal of Computing Sciences in Colleges*. 2015, roč. 30, č. 6., p 53–59 [online]. [vid. 19. 12. 2019]. Dostupné na: <https://dl.acm.org/citation.cfm?id=2753037>.
- [3] BRENNAN, Karen and Mitchel RESNICK. New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association*. 2012, p. 1–25 [online]. [vid. 19. 12. 2019]. Dostupné na: http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf.
- [4] GOUWS, Lindsey, BRADSHAW, Karen and Peter WENTWORTH. First year student performance in a test for computational thinking. In *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*. 2013, p. 271–277.
- [5] DENNING, Peter J. Remaining Trouble Spots with Computational Thinking. *Communications of the ACM*. 2017, roč. 60, č. 3, p. 33–39 [vid. 19. 12. 2019]. ISSN:0001-0782. Dostupné tiež na: <https://dl.acm.org/citation.cfm?id=2998438>.
- [6] Kolektív autorov. Národný projekt IT Akadémia – vzdelávanie pre 21. storočie [online]. [vid. 19. 12. 2019]. Dostupné na: <http://itakademia.sk/sk/domov/>.
- [7] ŠNAJDER, Ľubomír, GUNIŠ, Ján, TKÁČOVÁ, Zuzana, KOPČOVÁ, Veronika, HANESOVÁ, Angelika a Mária SPIŠÁKOVÁ. Mária Test informatického myslenia – priebežné výsledky. In: *DidInfo 2018: mezinárodní konference o vyučování informatiky*: 4.–6.4.2018, Liberec. – Liberec: Technická univerzita v Liberci, 2018. ISBN 9788074944246. S. 260–269, [online]. [vid. 19. 12. 2019]. Dostupné na: http://www.didinfo.net/images/DidInfo/files/Didinfo_2018.pdf.

- [8] Kolektív autorov. Národná správa PISA 2012 [online]. [vid. 19. 12. 2019]. Dostupné na: https://www.nucem.sk/dl/3491/N%C3%A1rodn%C3%A1_spr%C3%A1va_PISA_2012.pdf.
- [9] FRAILLON, Julian, John AINLEY, Wolfram SCHULZ, Tim FRIEDMAN a Daniel DUCKWORTH, 2019. *Preparing for life in a digital world: the IEA International Computer and Information Literacy Study 2018 International Report*. Amsterdam, The Netherlands: International Association for the Evaluation of Educational Achievement (IEA). ISBN 978-3-030-38781-5. Dostupné tiež na: https://research.acer.edu.au/ict_literacy/23.

Závislost na internetu, počítačových hrách a on-line komunikaci u studentů učitelských studijních programů

Addiction to Internet, Computer Games and Online Communication of Students of Teacher Studies Programs

Miroslav Chráska

Katedra technické a informační výchovy PdF UP v Olomouci
Žižkovo nám. 5, 771 40 Olomouc, Česká republika
miroslav.chraska@upol.cz

ABSTRAKT

Príspevok popisuje výsledky výzkumu, který zjišťoval míru závislosti na internetu, PC hrách a online komunikaci u studentů učitelství. Jako první výzkumná metoda pro posouzení závislosti na internetu byl použit přeložený, v zahraničí standardizovaný, dotazník CIAS. Jako druhá metoda k identifikaci závislostního chování studentů, ve vztahu ke hraní počítačových her, byl použit vlastní dotazník. Výzkumný vzorek činilo 226 studentů učitelských studijních programů různého zaměření. Bylo zjištěno, že závislost na internetu se objevuje u cca 3,1 % všech studentů. Vyšší závislost na internetu se objevuje u mužů (skóre 44) než u žen (40) a mladších studentů do 30 let (42) než u starších studentů (37) nad 30 let. Rozdíl je v obou případech statisticky významný ($p = 0,05$; $p = 0,003$). Závislost na počítačových hrách je vyšší u mužů než u žen ($p < 0,001$) a nezávisí na věku studentů ($p = 0,64$). Studenti technického a informatického zaměření přitom nevykazují vyšší závislost na internetu oproti studentům jiného zaměření ($p = 0,21$), avšak jsou více závislí na hraní počítačových her ($p < 0,001$) než studenti jiných zaměření. V délce on-line aktivit v závislosti na pohlaví byl zjištěn významný rozdíl jen v délce denní komunikace pomocí mobilu (volání, SMS, chat), kdy ženy touto aktivitou tráví 126 minut denně oproti 70 minutám u mužů. V délce on-line aktivit v závislosti na věku byl zjištěn významný rozdíl v délce aktivit na sociálních sítích, v délce komunikace pomocí mobilu (volání, SMS, chat) a délkou využívání internetu mimo školní potřeby, kdy mladší studenti do 30 let tráví těmito aktivitami statisticky významně více času než studenti starší 30 let.

ABSTRACT

This paper describes the results of a research study that examined the degree of addiction to the internet, computer games and online communication tools among the students of teacher training courses. The first research method was the translated version of the international standardized CIAS scale designed for the assessment of internet addiction. The other research instrument was designed by the authors of the study and analysed the students' addictive behaviour in relation to playing computer games. The research sample consisted of 226 students of teacher training courses of various specializations. Internet addiction was observed in 3.1 % of all the students. A higher degree of internet addiction was observed among men (score 44) compared with women (score 40) and in younger students up to 30 years of age (score 42) compared with older students over 30 years of age (score 37). The difference was statistically significant in both cases ($p = 0.05$; $p = 0.003$). Computer game addiction was higher in men than in women ($p < 0.001$) and was independent of the students' age ($p = 0.64$). The students of technical and computer science did not show higher internet addiction compared with the students of other specializations ($p = 0.21$) but they were more addicted to playing computer games ($p < 0.001$) than the students of other specializations. A significant difference was observed in the length of online activities depending on the students' gender only in the length of daily mobile communication (calls, text, chat), where women spent 126 minutes a day compared with 70 minutes for men. In terms of the length of online activities by age, a significant difference was observed in the length of social networking, mobile communication (calls, text, chat) and internet use for purposes other than school requirements. The students up to 30 years of age spent statistically significantly more time on these activities than the students over 30 years of age.

Klíčová slova

Závislost na internetu, závislost na počítačových hrách, on-line komunikace, student učitelských studijních programů.

Keywords

Internet addiction, computer game addiction, online communication, students of teacher training courses.

1 ÚVOD

Obecně se závislost na internetu definuje jako nadměrné používání internetu, které s sebou přináší do života jedince psychologické, sociální, pracovní nebo školní komplikace. Přitom jde o závislost na internetu ve všech jeho formách, tzn. hraní webových her, chatování, sledování e-mailů, běžné surfování apod. V širším pojetí pak tzv. netolismus zahrnuje i závislost na tzv. virtuálních drogách, kam lze zařadit mobilní telefon a televizi [1]. Panuje však shoda na tom, že pojem internetová závislost není příliš přesný, lépe je hovořit o závislém, či závislostním chování [2]. Obvykle potom rozlišujeme 3 základní typy závislosti na internetu. 1.) Závislost na on-line hrách 2.) Závislost na online komunikaci. 3.) Závislost na stránkách s pornografickým obsahem [2].

Vysokoškolští studenti jsou obecně považováni, s ohledem na závislostní chování v souvislosti s používáním informačních technologií, zvláště pak internetu a počítačových her, za vysoce rizikovou skupinu. Vysoký výskyt závislostí je obvykle vysvětlován snadným přístupem studentů k internetu, pružným denním programem, a také tím, že po příchodu na vysokou školu se zároveň velká část z nich ocitá v novém sociálním prostředí, kde přišli o své dosavadní vazby a začínají si budovat nové vztahy a postavení a tyto úkoly jim internet usnadňuje [2], [3]. Prevalence závislostního chování v souvislosti s používáním internetu mezi vysokoškolskou populací se v zahraničí pohybuje mezi 6–13 %. Problémem je ale skutečnost, že jen velmi malá část uživatelů internetu identifikuje negativní vliv tohoto chování na své životy. Studenti s rizikovým chováním ve vztahu k internetu nejčastěji využívají aplikace, jako jsou chatroomy, emaily a interaktivní počítačové hry, které umožňují obousměrnou komunikaci [4].

V souvislosti s negativními dopady závislostního chování spojeného s používáním internetu se nejčastěji mluví o zhoršených studijních výsledcích, pozdních příchodech nebo absencích ve výuce, zhoršeném zraku, spánkové deprivaci, bolestech zad, rukou, prstů a celkové únavě [4]. Na dopad závislostního chování v sociální oblasti je však nejednotný pohled, neboť některé studie uvádějí pozitivní přínos užívání internetu na mezilidské vztahy, jiné zase mluví o negativních následcích, jakými jsou větší sociální izolace a snížení počtu navázaných nových vztahů [2], [3].

2 CÍL VÝZKUMU

Základní výzkumný problém, který jsme si stanovili po prostudování relevantních zdrojů [1], [2], [3], [4], [9], [12], [13] byl: Je míra závislosti na internetu (včetně on-line aktivit) u studentů českých učitelských studijních programů odlišná než u studentů jiného zaměření? Je tato závislost ovlivněna obecně uváděnými intervenujícími proměnnými? Jako další problém jsme stanovili: Je míra závislostního chování studentů českých učitelských studijních programů v souvislosti s hraním počítačových her ovlivněna pohlavím studentů, jejich věkem a studijním zaměřením? Přínos výzkumu spatřujeme v hledání určitých specifik, kdy by studenti učitelského zaměření měli, oproti vrstevníkům, vzhledem ke své budoucí edukační roli, vykazovat odlišné „závislostní“ charakteristiky.

3 POPIS VÝZKUMNÝCH METOD A VÝZKUMNÉHO VZORKU

3.1 Dotazník závislostního chování v souvislosti s hraním počítačových her

Jako výzkumná metoda pro měření závislostního chování v souvislosti s hraním počítačových her byl ve výzkumu použit zatím nestandardizovaný dotazník [5], který vycházel z dotazníku použitého již

v roce 2015 [6]. Na základě odpovědí studentů na dvanáct otázek v dotazníku, což byla kritéria závislostního chování definovaná především podle APA [7], bylo pro každého studenta spočítáno celkové skóre míry jeho závislostního chování v souvislosti s hraním počítačových her. Toto skóre bylo vypočítáno podle míry souhlasu (na škále 0-5), kdy odpověď „zcela mne vystihuje“ byla kódována hodnotou 5 a další odpovědi „vystihuje mne“ (4), „spíše mne vystihuje“ (3), „spíše mne nevystihuje“ (2), „nevystihuje mne“ (1), „vůbec mne nevystihuje“ (0), hodnotami uvedenými v závorce. Studenti posuzovali následujících 12 tvrzení: Počítačové hraní je pro mě odraťování, najednou zapomenu na problémy nebo stres, odpočinu si a cítím se fajn. Dříve jsem měl/a i jiné zájmy, než je počítačové hraní, ale dnes většinu volného času hraji hry. I když vím, že budu psát ve škole písemku, tak raději hraji hry. Když zrovna nehraji, často přemýšlím o tom, jak se mi hra dařila, čeho všeho dosáhnu nebo jak budu postupovat, až si zase zahraji. Kvůli hraní počítačových her ponocuji. Nedovedu si představit život bez počítačových her. Mám pocit, že se počítačové hry pro mne stávají čím dál více důležité. Musím trávit stále více času hraním počítačových her, abych se cítil spokojen. Cítím, že nemůžu kontrolovat množství času, které trávím hraním počítačových her. Snažil/a jsem se snížit množství času stráveného hraním, ale neuspěl/a jsem. Vzhledem k mému častému hraní se někdy dostanu do problémů ve škole, v práci nebo doma. Trávím více času hraním počítačových her, než jsem schopen/na přiznat ostatním. Ostatním neříkám pravdu, abych skryl/a, kolik času jsem strávil/a hraním počítačových her. Už jsem ztratil/a nebo riskoval/a důležitý vztah nebo přátelství kvůli počítačovému hraní. Kvůli hraní počítačových her jsem riskoval/a nebo ztratil/a své příležitosti ve škole nebo práci. Rodiče mi často zakazují nebo mě omezují v hraní počítačových her, jsem poté naštvaný nebo hraji za jejich zády. Maximální celkový počet bodů, které bylo takto možné získat činil 60. Pokud student získal alespoň 30 bodů, vykazoval pro potřeby našeho výzkumu již převažující znaky závislostního chování podle APA [7]. Před vyhodnocením dotazníku byla pomocí Cronbachova alfa určena reliabilita dotazníku, která činila 0,94, což značí vysokou reliabilitu měření. Žádná z položek dotazníku přitom nesnižovala jeho celkovou reliabilitu pod hodnotu 0,89. Výsledky výzkumu závislostního chování v souvislosti s hraním počítačových her uvádí kapitola 4.2.

3.2 Chenova škála závislosti na internetu CIAS

Pro posouzení závislosti studentů na internetu byla použita škála závislosti na internetu The Chen Internet Addiction Scale (CIAS). Ta byla vyvinuta v roce 2003 Dr. Sue-Huei Chenem a jeho spolupracovníky (Národní Taiwanská univerzita Taipei) za účelem zhodnocení závažnosti závislosti na internetu [8]. Jedná se o dotazník o 26 položkách, každá položka se skóruje na 4 bodové škále. Respondent zaškrtnává nejvhodnější odpověď (škála nikdy – téměř nikdy – často – téměř vždy) podle výskytu za posledních 6 měsíců. Dosažené skóre se pohybuje od 26 do 104 bodů. Čím vyšší skóre respondent dosáhne, tím závažnější je závislost na internetu. Skóre 64 a více se přitom již považuje za závislost. Jednotlivé škály dotazníku CIAS se vážou na 5 dimenzí, které pokrývají syndrom kompulzivního konání, narušenou kontrolu nad vlastním chováním, abstinenci příznaky, míru tolerance, problémy v interpersonálních vztazích a zdravotní problémy. Pro účely výzkumu jsme použili českou verzi, kterou přeložila Mgr. Petra Vondráčková (viz <https://www.adiktologie.cz/chenova-skala-zavislosti-na-internetu-chen-internet-addiction-scale-cias>). Před vyhodnocením dotazníku CIAS byla ověřena jeho konstruktová validita (viz Rizika a omezení zjištěných závěrů výzkumu). Dále byla vypočítána pomocí Cronbachova alfa i jeho reliabilita, která dosáhla hodnoty 0,92, přičemž žádná z položek dotazníku CIAS tuto hodnotu nesnižovala. Dále byly také zjišťovány vybrané on-line aktivity studentů (ze subjektivního pohledu studentů), jejich volnočasové aktivity a další demografické údaje.

3.3 Popis výzkumného vzorku respondentů

Výzkumné šetření bylo provedeno anonymně u studentů Pedagogické fakulty Univerzity Palackého v Olomouci v průběhu roku 2019. Celkový výzkumný vzorek tvořilo 226 studentů různých učitelských studijních programů v prezenční (označeno písmenem P) a kombinované (K) formě studia. Složení výzkumného vzorku popisuje tab. 1.

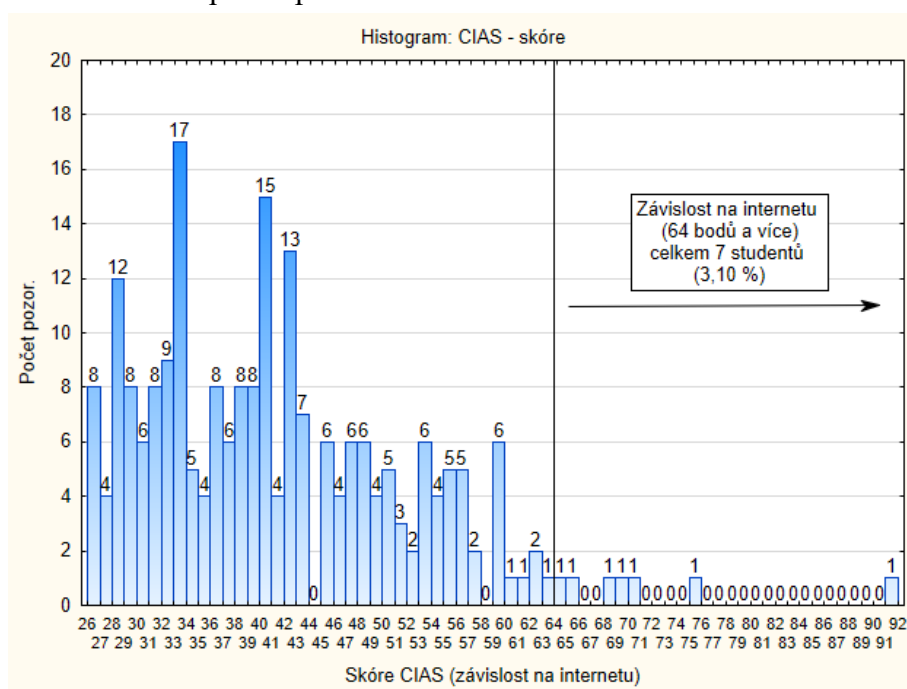
Tabulka 1: Složení výzkumného souboru studentů učitelských studijních programů

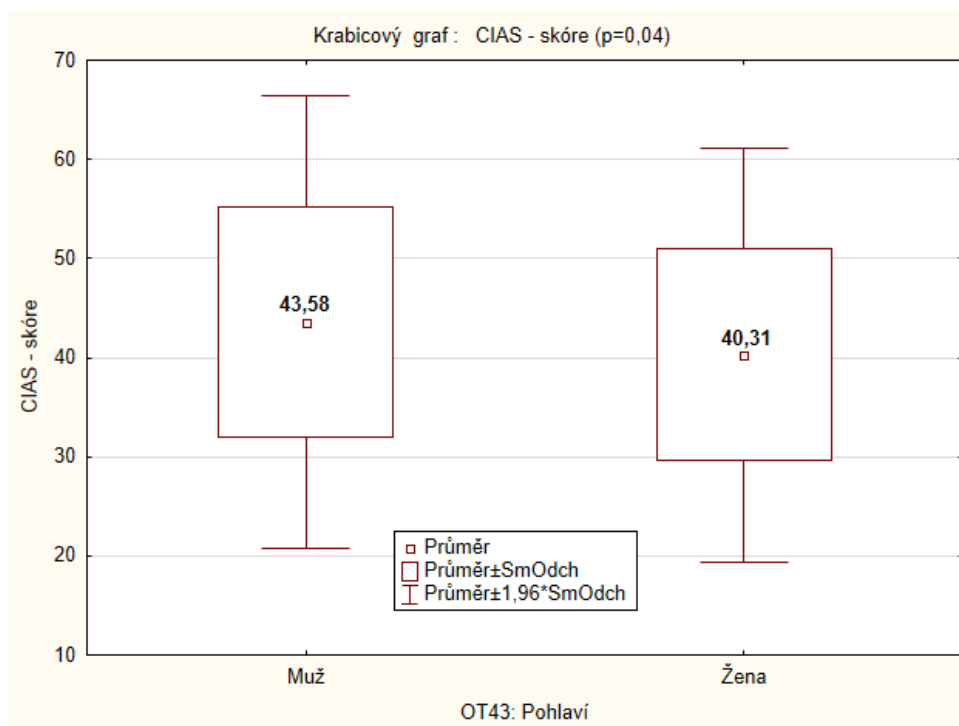
Forma studia	Pohlaví	Věk (do/nad 30 let)	Zaměření studia Technika/informatika	Zaměření studia Jiné	Celkem
Prezenční	Muž	Do 30 let	20	15	35
Prezenční	Muž	30 let a více	0	0	0
Celkem			20	15	35
Prezenční	Žena	Do 30 let	16	85	101
Prezenční	Žena	30 let a více	0	0	0
Celkem			16	85	101
Kombinovaná	Muž	Do 30 let	9	2	11
Kombinovaná	Muž	30 let a více	11	2	13
Celkem			20	4	24
Kombinovaná	Žena	Do 30 let	4	28	32
Kombinovaná	Žena	30 let a více	5	29	34
Celkem			9	57	66
Sloupce celkem			65	161	226

4 VÝSLEDKY VÝZKUMU

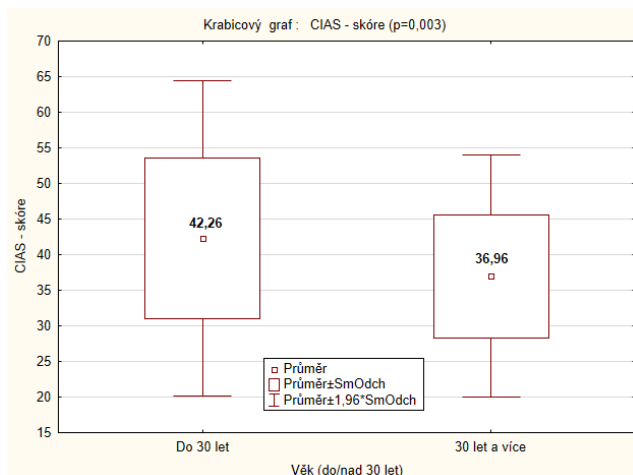
4.1 Míra závislosti na internetu u studentů učitelských studijních programů

V první části výzkumu nás zajímalo, jaká část studentů vykazuje závislost na internetu a dosáhla tedy minimálního skóre závislosti 64 bodů. Výsledky uvádí obr. 1. Z něj je patrné, že závislost na internetu vykazuje celkem 7 studentů, z toho jeden velmi výrazně – dosažené skóre 91. Závislost na internetu se tedy objevuje u 3,1 % studentů učitelských studijních programu, což ve srovnání s obecně udávanou prevalencí 6–10 % [9] není mnoho. Je to zřejmě způsobeno zaměřením studentů na pomáhající profesi – učitelství. Dále nás zajímalo, zda mezi mírou závislosti na internetu u studentů je rozdíl podle pohlaví studentů, jejich věku a zaměření studia. Srovnání závislosti bylo provedeno pomocí t-testu [10]. v programu STATISTICA 12 CZ [11] a uvádí jej obr. 2–4. Pomocí t-testu byla provedena i další srovnání skupin respondentů v tomto článku.

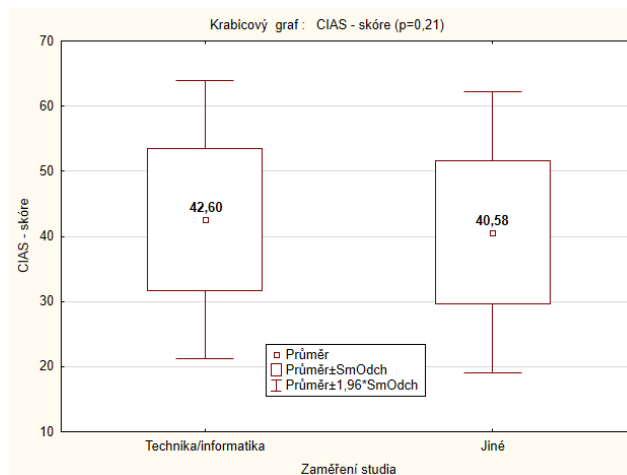
**Obrázek 1: Závislost na internetu u studentů učitelských studijních programů**



Obrázek 2: Závislost na internetu u studentů učitelských studijních programů v závislosti na jejich pohlaví



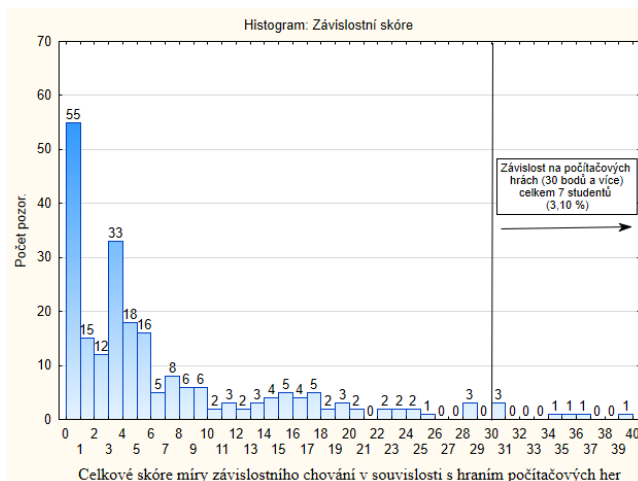
Obrázek 3: Závislost na internetu u studentů učitelských studijních programů v závislosti na jejich věku



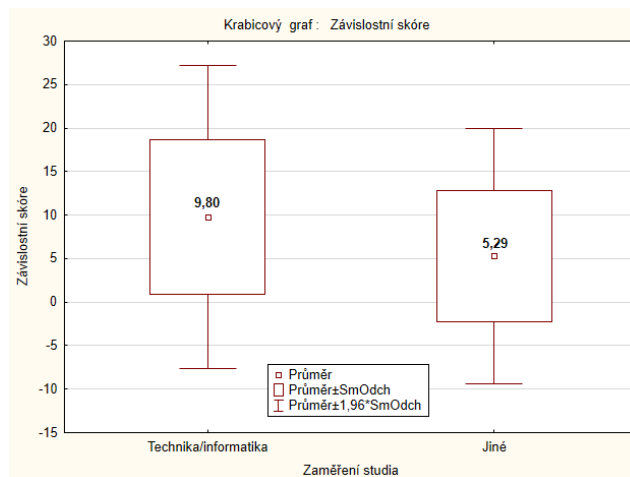
Obrázek 4: Závislost na internetu u studentů učitelských studijních programů v závislosti na jejich studijním zaměření

4.2 Míra závislostního chování v souvislosti s hraním počítačových her u studentů učitelských studijních programů

Celkový počet studentů, kteří získali 30 bodů a více byl 7 – viz obr. 5, jedná se o 3,10 % z celkového počtu studentů. Při porovnání míry závislostního chování v souvislosti s hraním počítačových her u studentů různého zaměření bylo zjištěno, že studenti technického a inženýrského zaměření vykazují vyšší míru závislosti než studenti jiných zaměření ($p = 0,0001$) – viz obr. 6. Srovnání bylo provedeno pomocí t-testu v programu STATISTICA 12 CZ [11].



Obrázek 5: Skóre závislostního chování v souvislosti s hraním počítačových her u studentů učitelských studijních programů



Obrázek 6: Skóre závislostního chování v souvislosti s hraním počítačových her u studentů učitelských studijních programů v závislosti na jejich studijním zaměření

4.3 Délka jednotlivých on-line aktivit u studentů učitelských studijních programů v závislosti na pohlaví, věku a zaměření studia

Ve výzkumu nás dále zajímal vliv vybraných proměnných na délku on-line aktivit studentů. Konkrétně šlo o obecně uznávaný vliv pohlaví [9] studentů, věku studentů [2] a také zaměření studia [17]. Jako věkovou hranici pro srovnání studentů jsme zvolili 30 roků, protože z hlediska psychologického členění vývoje jedince, do 20 roků probíhá v rámci období adolescence psychické vyzrání a mezi 20–30 rokem v rámci období mladší dospělosti již sociální, pracovní a partnerské dozrání jedince. Studijní zaměření jsme rozdělili na technické/informatické a jiné, kdy jsme u první skupiny předpokládali, vzhledem k zaměření studia, jiný „pohled“ a zkušenosti s on-line technologiemi. Srovnání bylo opět provedeno pomocí t-testu a uvádí jej tab. 2–4 (otázka 37 obsahovala 4 podotázky týkající se délky jednotlivých on-line aktivit studentů, studenti zde uváděli počet hodin a minut, které těmto aktivitám věnují, souhrnné srovnání bylo ale provedeno v minutách).

5 ZÁVĚR

Bylo zjištěno, že závislost na internetu se objevuje u 3,10 % studentů učitelských studijních programů, což je méně než např. v Japonsku [12] nebo u srovnatelné populace dospělých v Evropě [9]. Vyšší závislost na internetu se objevuje u mužů (skóre 43,58) než u žen (40,31) což bylo vzhledem k předpokládaným výsledkům překvapivé. Podle srovnání v 9 evropských zemích [13] se totiž prevalence problematického používání internetu (PIU) pohybovala od 14 % do 55 % a byla častější u žen. Tento rozpor může být způsoben učitelským zaměřením studentů v našem výzkumném vzorku. Typicky je ale uváděno, že rizikovější skupinou pro závislost na internetu jsou muži. Většina studií uvádí do různé míry významné výsledky, které většinou poukazují na vyšší průměrnou míru závislosti u mužů [14]. Zjištění však zvláště u univerzitních studentů jsou ambivalentní. Některé studie uvádí, že rozdíl nebyl prokázán [15], popřípadě byl pouze mírný [16], nebo že se rozdíl neprokázal [17]. Bylo také zjištěno, že statisticky vyšší míra závislost na internetu se objevuje u mladších studentů (42,26) oproti starším studentům (36,96). Naopak nebyly zjištěny rozdíly v míře závislosti na internetu u studentů technického/informatického zaměření, kdy jsme předpokládali, že oproti studentům jiných studijních programů budou tyto vykazovat vyšší míru závislosti.

Tabulka 2: Délka jednotlivých on-line aktivit u studentů učitelských studijních programů v závislosti na jejich pohlaví – srovnání pomocí t-testu

Proměnná	t-testy; grupováno: OT43: Pohlaví (Data DIDINFO 2020.sta)								
	Skup. 1: Muž Skup. 2: Žena								
	Průměr Muž	Průměr Žena	t	sv	p	Poč.plat Muž	Poč.plat. Žena	Sm.odch. Muž	Sm.odch. Žena
OT37: Kolik minut v průměru za jeden den trávíte na sociálních sítích?	114,69	147,54	-1,43	224	0,155	59	167	128,10	159,58
OT37: Kolik minut v průměru za jeden den trávíte komunikací pomocí mobilu (volání, SMS, chat)?	70,39	125,84	-2,46	224	0,015	59	167	89,41	164,44
OT37: Kolik minut v průměru za jeden den trávíte využíváním internetu (mimo školní potřeby)?	164,41	131,66	1,53	223	0,127	59	166	157,60	134,71
OT37: Kolik minut v průměru za jeden den trávíte využíváním internetu (pro školní potřeby)?	86,97	103,14	-1,10	223	0,271	59	166	89,72	99,05
OT38: Kolik času během všech vyučovacích hodin v jednom dni věnujete on-line aktivitám (využívání sociálních sítí, mobilu a hraní počítačových her): (minut)	61,41	72,12	-0,69	223	0,489	59	166	108,22	99,83

Tabulka 3: Délka jednotlivých on-line aktivit u studentů učitelských studijních programů v závislosti na jejich věku – srovnání pomocí t-testu

Proměnná	t-testy; grupováno: Věk (do/nad 30 let) (Data DIDINFO 2020.sta)								
	Skup. 1: Do 30 let Skup. 2: 30 let a více								
	Průměr Do 30 let	Průměr 30 let a více	t	sv	p	Poč.plat Do 30 let	Poč.plat. 30 let a více	Sm.odch. Do 30 let	Sm.odch. 30 let a více
OT37: Kolik minut v průměru za jeden den trávíte na sociálních sítích?	160,64	56,43	4,33	224	0,000	179	47	162,08	56,21
OT37: Kolik minut v průměru za jeden den trávíte komunikací pomocí mobilu (volání, SMS, chat)?	128,85	44,77	3,50	224	0,001	179	47	162,74	48,62
OT37: Kolik minut v průměru za jeden den trávíte využíváním internetu (mimo školní potřeby)?	157,25	75,85	3,60	223	0,000	178	47	150,93	66,52
OT37: Kolik minut v průměru za jeden den trávíte využíváním internetu (pro školní potřeby)?	103,75	80,53	1,47	223	0,144	178	47	98,21	89,69
OT38: Kolik času během všech vyučovacích hodin v jednom dni věnujete on-line aktivitám (využívání sociálních sítí, mobilu a hraní počítačových her): (minut)	74,41	50,00	1,46	223	0,145	178	47	106,03	82,96

Tabulka 4: Délka jednotlivých on-line aktivit u studentů učitelských studijních programů v závislosti na jejich studijním zaměření – srovnání pomocí t-testu

Proměnná	t-testy; grupováno: Zaměření studia (Data DIDINFO 2020.sta)								
	Skup. 1: Technika/informatika Skup. 2: Jiné								
	Průměr Technika/i nformatika	Průměr Jiné	t	sv	p	Poč.plat Technika/i nformatika	Poč. plat. Jiné	Sm.odch. Technika/i nformatika	Sm.od ch. Jiné
OT37: Kolik minut v průměru za jeden den trávíte na sociálních sítích?	128,80	143,07	-0,64	224	0,525	65	161	130,30	160,66
OT37: Kolik minut v průměru za jeden den trávíte komunikací pomocí mobilu (volání, SMS, chat)?	95,97	117,58	-0,98	224	0,329	65	161	137,02	155,39
OT37: Kolik minut v průměru za jeden den trávíte využíváním internetu (mimo školní potřeby)?	169,37	128,66	1,96	223	0,051	64	161	156,17	133,91
OT37: Kolik minut v průměru za jeden den trávíte využíváním internetu (pro školní potřeby)?	95,88	100,11	-0,30	223	0,768	64	161	85,63	101,07
OT38: Kolik času během všech vyučovacích hodin v jednom dni věnujete on-line aktivitám (využívání sociálních sítí, mobilu a hraní počítačových her): (minut)	88,56	61,66	1,79	223	0,074	64	161	139,78	81,57

Rizika a omezení zjištěných závěrů výzkumu

Vzhledem k zatím ne zcela úplným výsledkům našeho výzkumu, nebyly do našeho výzkumného vzorku zatím zahrnuti studenti všech studijních zaměření, což je další předpokládaná intervenující proměnná. Dalším omezením interpretace výzkumu je použitý dotazník CIAS z hlediska mezikulturního přenosu použitých škál. V zahraničí již totiž byla učiněna revize nástroje – CIAS-R [18], která ukazuje spíše na čtyři dimenze závislosti na internetu. Nicméně v ČR se CIAS používá jako orientační – základní zkouška závislosti na internetu (např. na Klinice adiktologie 1. lékařské fakulty Univerzity Karlovy a Všeobecné fakultní nemocnice v Praze viz: https://poradna.adiktologie.cz/otestujte-se/?poll_id=5). Ověření konstruktové validity měrného nástroje CIAS bylo u studentů PdF UP v Olomouci provedeno pomocí faktorové analýzy – viz tab. 5. Z ní je patrné, že shoda vypočítané faktorové struktury dotazníku CIAS s teoretickou není zcela ideální, avšak pro základní posouzení závislosti na internetu je evaluační nástroj postačující. Nicméně zjištěné nedostatky nástroje by bylo vhodné dále eliminovat ve spolupráci s psychology. Podobné výsledky, týkající se konstruktové validity, přinesla také analýza dotazníku CIAS v Malaisii [19].

Tabulka 5: Ověření konstruktové validity měrného nástroje CIAS u studentů PdF UP pomocí faktorové analýzy v programu STATISTICA 12 CZ [11]

Faktor. zátěže (Varimax normaliz.) Extrakce: Max. věrohodné faktory (Označené zátěže jsou > 0,40)					
Proměnná	F1	F2	F3	F4	F5
O1: Bylo mi vícekrát než jednou řečeno, že na internetu trávím příliš mnoho času.	0,28	0,06	0,26	0,51	0,23
O2: Po surfování na internetu mě bolívá v zádech, případně pociťuji jiné fyzické potíže.	0,09	0,06	0,68	0,12	-0,05
O3: Zjišťuji, že se vyžívám na internetu na úkor toho, abych se stýkal(a) se svými kamarády.	0,16	0,20	0,43	0,43	0,21

O4: Myšlenka na to, že se připojím na internet, je první věc, která mě každý den ráno hned po probuzení napadne.	0,58	0,04	0,00	0,22	0,03
O5: Nedaří se mi zvládat své nutkání připojit se na internet.	0,58	0,09	0,18	0,29	0,05
O6: Více než jednou se mi stalo, že jsem kvůli internetu spal/a méně než čtyři hodiny.	0,16	0,23	0,21	0,26	0,12
O7: Surfování na internetu se negativně projevuje na mém tělesném zdraví.	0,20	0,19	0,70	0,10	0,02
O8: Přestože se užívání internetu negativně odrazilo na mých vztazích s jinými lidmi, množství času, které na internetu trávím, se nezmenšilo.	0,13	0,30	0,50	0,24	0,27
O9: Když jsem nějakou dobu bez internetu, mám pocit, že mi něco chybí.	0,71	0,13	0,26	0,12	0,12
O10: Kvůli internetu trávím méně času s rodinou.	0,30	0,38	0,41	0,23	0,15
O11: Mé odpočinkové aktivity se omezují na užívání internetu.	0,38	0,16	0,29	0,43	0,06
O12: Cítím se deprimovaný/a a skleslý/a, když na nějakou dobu přestanu internet užívat.	0,61	0,32	0,24	0,11	0,20
O13: Doba, kterou týdně trávím na internetu, se za poslední pololetí (resp. za posledních 6 měsíců) výrazně zvýšila.	0,30	0,01	0,17	0,40	0,13
O14: Trávení času na internetu má určitý negativní dopad na mé výsledky ve škole/můj pracovní výkon.	0,06	0,32	0,24	0,56	0,11
O15: Nedaří se mi zvládat své nutkání připojit se na internet poté, co jsem se již odhlásil/a, abych mohl/a dělat nějakou jinou plánovanou práci.	0,38	0,33	0,31	0,36	0,05
O16: Kdyby nebylo internetu, neměl/a bych v životě mnoho radosti.	0,29	0,12	0,05	0,22	0,38
O17: Snažím se trávit na internetu méně času, ale mé snahy jsou marné.	0,33	0,25	0,17	0,48	0,19
O18: Zůstávám online po delší dobu, než jsem měl/a v úmyslu, i když jsem měl/a původně v plánu jen se na něco rychle podívat.	0,38	0,15	0,13	0,51	-0,12
O19: Jakmile se dostanu na internet, vjede do mě nová energie bez ohledu na únavu, kterou cítím.	0,45	0,42	0,00	0,22	0,13
O20: Během dne se cítím unavený/a, protože dlouho do noci surfuji na internetu.	0,18	0,79	0,25	0,23	-0,11
O21: Kvůli užívání internetu nepravidelně jím.	0,15	0,66	0,21	0,17	-0,02
O22: Musím na internetu trávit stále více času, abych dosáhl/a stejného uspokojení jako předtím.	0,17	0,59	0,03	0,15	0,47
O23: Zvykám si méně spát, abych mohl/a trávit více času na internetu.	0,10	0,74	0,12	0,10	0,23
O24: Jsem neklidný/a a podrážděný/a, když připojení na internet nefunguje nebo když není takové připojení k dispozici.	0,49	0,23	0,22	0,24	-0,03
O25: Zjišťuji, že trávím na internetu stále více času.	0,23	0,19	-0,02	0,59	0,05
O26: Když po určitou dobu nechodím na internet, začnu se cítit nesvůj/nesvá.	0,66	0,20	0,09	0,25	0,33

6 PODĚKOVÁNÍ

Příspěvek vznikl v rámci řešení projektu IGA_PdF_2019_043 „Analýza vybraných aktuálních problémů ve výuce informatiky zaměřených předmětů s akcentem na on-line technologie“.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] YOUNG, KS. The evolution of Internet addiction. In *Addictive Behaviors*. Volume 64, 2017, pp. 229–230, ISSN 0306-4603.
- [2] BLINKA, L. et al. *Online závislosti*. Praha: Grada, 2015.
- [3] ŠKAŘUPOVÁ, K. *Internet: From Excess to Addiction* (Disertation). Brno: Masarykova univerzita, Fakulta sociálních studií, 2015. http://is.muni.cz/th/43275/fss_d_b1/.
- [4] CERNIGLIA, L. et al. IA in adolescence: Neurobiological, psychosocial and clinical issues. In *Neuroscience & Biobehavioral Reviews*. Volume 76, Part A, May 2017, Pages 174–184. <http://dx.doi.org/10.1016/j.neubiorev.2016.12.024>.
- [5] CHRÁSKA, M. jr. a J. BASLER. Research of Computer game addiction among the 18-year old students of general upper secondary schools in the Czech Republic. In *ICERI 2016 Proceedings. Seville, 9th International Conference of Education, Research and Innovation*, 2016, pp. 69–78.
- [6] CHRÁSKA, M. jr. Grammar School Students and Their Typology According to Dependence on Computer Games. In *SGEM 2016 Conference Proceedings*. Sofia: STEF92 Technology Ltd., 2016, pp. 795–802. ISBN 978-619-7105-70-4. ISSN 2367-5659. Retrieved from DOI 10.5593/SGEMSOCIAL2016/B11/S03.101.
- [7] APA (American Psychiatric Association). *DSM-5® classification*. Arlington, VA: American Psychiatric Association, 2016.
- [8] CHEN, SH. et al. Development of Chinese Internet addiction scale and its psychometric study. In *Chinese Journal of Psychology*. 45, 2003, pp. 279–294.
- [9] WENLIANG, S. et al. Are males more likely to be addicted to the internet than females? A meta-analysis involving 34 global jurisdictions. *Computers in Human Behavior*. Volume 99, 2019, pp 86–100, ISSN 0747-5632, <https://doi.org/10.1016/j.chb.2019.04.021>.
- [10] CHRÁSKA, M. sr. *Metody pedagogického výzkumu*. 2. aktual. vyd. Praha: Grada, 2016.
- [11] STATSOFT, INC. *STATISTICA (data analysis software system)*. Version 12, 2013. www.statsoft.com.
- [12] SEKI, T. et al. Relationship between internet addiction and depression among Japanese university students. In *Journal of Affective Disorders*. Volume 256, 2019, pp. 668–672, ISSN 0165-0327, <https://doi.org/10.1016/j.jad.2019.06.055>.
- [13] LACONI, S. et al. Psychometric Evaluation of the Nine-Item Problematic Internet Use Questionnaire (PIUQ-9) in Nine European Samples of Internet Users. *Front. Psychiatry*. 10:136, 2019, doi: 10.3389/fpsyt.2019.00136.
- [14] WEINSTEIN, A. et al. Internet addiction is associated with social anxiety in young adults. In *Annals of Clinical Psychiatry: Official Journal of the American Academy of Clinical Psychiatrists*. 27(1), 2015, pp. 4–9.
- [15] KUSS, D. J. et al. (2013). Internet addiction in students: Prevalence and risk factors. *Computers in Human Behavior*. 29(3), 2013, pp. 959–966. <http://doi.org/10.1016/j.chb.2012.12.024>.
- [16] SERVIDIO, R. Exploring the effects of demographic factors, Internet usage and personality traits on Internet addiction in a sample of Italian university students. *Computers in Human Behavior*. 35, 2014, pp. 85–92. <http://doi.org/10.1016/j.chb.2014.02.024>.
- [17] CAPETILLO-VENTURA, N. a M. JUÁREZ-TREVIÑO. Internet addiction in university medical students. *Medicina Universitaria*. 17(67), 2015, pp. 88–93. <http://doi.org/10.1016/j.rmu.2015.02.003>.
- [18] MAK, KK. et al. Psychometric Properties of the Revised Chen Internet Addiction Scale (CIAS-R) in Chinese Adolescents. In *Journal of abnormal child psychology*. 42, 2014, 1237. DOI 10.1007/s10802-014-9851-3.
- [19] RADEEF, A. et al. Translation and validation study of the Chen Internet Addiction Scale (CIAS) among Malaysian college students. *Journal of International Dental and Medical Research*. 11, 2018, pp. 32–37.

Možnosť implementácie ochrany kultúrneho dedičstva do vyučovania informatiky

The Possibility of Implementing the Protection of Cultural Heritage in the Teaching of Informatics

Ondrej Takáč
Univerzita J. Selyeho
Bratislavská cesta 3322
945 01 Komárno
Slovenská republika
takac.ondrej@gmail.com

ABSTRAKT

V súčasnosti je veľmi aktuálna otázka ochrany kultúrneho dedičstva. Moderné technológie sú v týchto snahách veľmi nápomocné. V našom príspevku sa zameriame na tvorbu 3D modelov sakrálnych objektov. Vytvorený 3D model nielenže hodnoverne odzrkadľuje reálny vzhľad objektu, ale je aj jeho metrickým odrazom. To znamená, že ho môžeme presne umiestniť do globálneho alebo lokálneho súradnicového systému a získať jeho všetky rozmery. Tieto postupy sú celkom dobre implementovateľné aj do vyučovania informatiky, čím nielenže vieme zvýšiť záujem žiakov o informatiku, ale tento predmet sa pre žiakov môže stať aj zaujímavejší.

ABSTRACT

At present, the issue of cultural heritage protection is very topical. Modern technologies are very helpful in these efforts. In these papers we focus on 3D models of sacral objects. The created three-dimensional model not only plausibly mirrors the real appearance of the object, but is also its metric reflection. This means that it is possible to accurately locate it in a global or local coordinate system and get all its dimensions. These procedures are quite well implementable in the teaching of informatics, which not only can increase pupils' interest in informatics, but this subject can also become more interesting for students.

Kľúčové slová

Kultúrne dedičstvo, modelovanie, UAV, fotogrametria.

Keywords

Cultural heritage, modeling, UAV, photogrametry.

1 ÚVOD

V súčasnosti evidujeme rozmach bezpilotných leteckých prostriedkov (UAV – unmanned aerial vehicle) vo viacerých oblastiach nielen robotiky ale aj filmu, podnikania či každodenného života. Tieto sú ideálne na snímkovacie činnosti či už s cieľom získania filmu alebo fotozáberov. Súčasný vývoj systémov plánovania letu umožňuje udržanie optimálnej geometrie letu a tým vylúčenie zisku záberov so slabým alebo žiadnym prekrytom. Optimálny prekryt z hľadiska presnosti, úplnosti, počtu snímkov a kvality aerotriangulácie je 60% v pozdĺžnom smere (smere letu) a 20–30% v priečnom smere. Bežne sa však v súčasnosti použitím digitálnych kamier realizuje až 80–90% pozdĺžny a 50–60% priečny prekryt. [3][4]

2 METÓDY A POSTUPY

Základom tvorby modelov sú fotozábery – fotografie alebo video objektu, ktorého model chceme vytvoriť. Pre získanie fotografií je potrebné splniť minimálne zákonné podmienky na získanie takýchto záberov. Môžeme využiť fotografie získané fotoaparátmi pri fotografovaní zo zeme, ale aj zo vzduchu – pomocou bezpilotných lietadiel (UAV). Je zrejmé, že pri ich získavaní sa vykonáva letecká činnosť, ktorá je viazaná na patričné pilotské oprávnenie príslušnej kategórie UAV. Tím samozrejme nechceme nikoho domotivovať, ale presne naopak. Existujú firmy a spoločnosti, ktoré takéto práce vykonávajú aj s odtajnením. To znamená, že okrem povolenia vlastníka objektu, dotknutých osôb a pod., získané zábery sa musia odtajniť, lebo pri fotografovaní objektu z rôznych uhlov sa zároveň vykonáva mapovanie zemského povrchu a môže sa stať, že na niektorom so záberov budú aj utajované miesta a skutočnosti. Musíme mať zreteľ aj zákon o ochrane osobných údajov! Získané fotografie po odtajnení už môžeme voľne používať a využiť ich aj na hodinách informatiky opakovane. Celkovo môžeme povedať, že v súčasnosti existuje dostatok prostriedkov s dostatočnými parametrami na získanie fotografií dostatočnej kvality.

Chápeme, že môžeme mnohé deti motivovať k získaniu fotozáberov amatérskymi dronmi, ale zastávame názor, že správnou informovanosťou tomu vieme aj zabrániť – ak budeme chcieť.

2.1 Získanie fotozáberov

Pri tvorbe každého modelu sú najpodstatnejšie detailné fotozábery danej sakrálny pamiatky s prekrytom v jednotlivých záberoch. Literatúra uvádza požiadavku aspoň 2/3 ale my s cieľom dosiahnuť lepšie výsledky sme zvolili 3/4. V tomto konkrétnom prípade sme použili výhradne zábery z UAV, konkrétne UAV DJI Phantom, ktorý sa nám už osvedčil aj pri iných prácach. Nie je síce najšpičkovejší, ale ľahko sa ovláda, má dobrú stabilizáciu obrazu a dobré letové vlastnosti s dobrou ovládateľnosťou aj pri bočnom vetre. Podrobnejšie parametre stabilizovanej kamery ktorú UAV nesie uvádzame nižšie:

Rozlíšenie	4K (3 840 × 2 160)
Fps (4K)	30/60
Fps (Full HD)	120
Rozlíšenie	12 Mpx / 20 Mpx
Svetelnosť	f2,8
Zorný uhol	94°

Je zrejmé, že na splnenie nutnej požiadavky – existencia prekrytia v jednotlivých záberoch, počet záberov rýchlo môže dosiahnuť aj niekoľko desiatok až stoviek. Tento počet je však potrebný na korektnú tvorbu modelu. My sme si ako sakrálnu stavbu vybrali rotundu Dvanástich apoštolov v Bíni. Na vytvorenie 3D modelu sme použili 253 geodeticky lokalizovaných záberov (teda záberov so známou polohou kamery pri každej snímke – GPS dáta), ktoré sme získali v troch letových výškach. Hoci na vytvorenie 3D modelu to nie je nutná podmienka, tieto dáta však výrazne dopomôžu k precízite modelu a samotnej tvorby modelu pomocou automatizovaných postupov.

2.2 Tvorba modelu

Prvým krokom je prvotná filtrácia fotozáberov, ktorá je potrebná k softvérovému spracovaniu. To znamená, že žiaci a študenti vyradia tie fotozábery, ktoré intuitívne vycítia ako zlé, nevhodné – zábery silno zašumené, zábery na ktorých sa sakrálny objekt nenachádza alebo zábery na ktorých je sakrálna stavba slabo výrazná (zaliata slnečným svetlom, v tme a pod.). Učiteľ vhodným výberom prvotných fotografií môže žiakov nasmerovať správnym smerom – poukázať na nesprávne a správne fotozábery.

- Pod vedením učiteľa sa žiaci môžu pokúsiť vytvoriť model na základe nedostatočne malého počtu fotozáberov – výsledkom bude deformovaný, zašumený a nekompletný model (chýba strecha, zadná stena a pod.), alebo

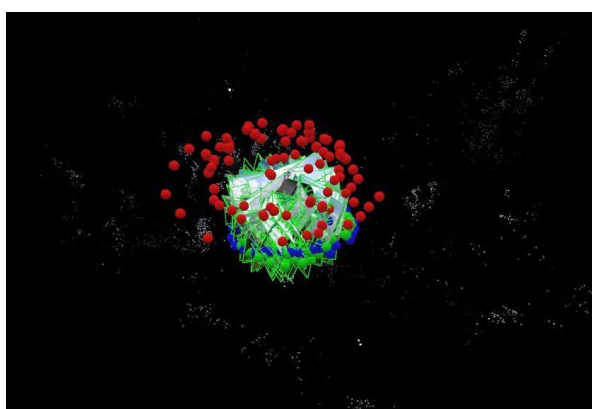
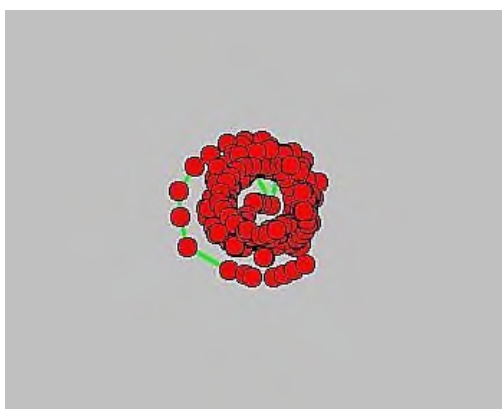
- na základe nevyfiltrovaných fotografií, kedy výsledkom bude zašumený model (šum v tomto prípade znamená napr. nejasný kríž na cípe rotundy, ale tento ríž je neoddeliteľne spojený s akýmsi oblakom ktorý je súčasťou modelu).

Následne môžeme pristúpiť k tvorbe modelu. Použité softvérové pozadie je taktiež neodmysliteľnou súčasťou tvorby. V našom prípade sme použili profesionálny fotogrametrický softvér Pix4D mapper Pro. Podporuje rôzne druhy kamier, či už RGB, multispektrálne alebo termálne. Pix4D mapper Pro umožňuje hlavne desktopové a cloudové spracovanie, generovanie 3D mračien bodov a ich klasifikáciu, Generovanie 3D textúrovaných modelov ale aj 2D merania a 2D vektorové výstupy. [2] Existuje viacero softvérových produktov, ale nie všetky sú Freeware a mnohé sú finančne náročné. Z pomedzi platených produktov ako „výťaz“ nám vyšiel Agisoft, dnes už známi ako Agisoft Metashape. Pix4D mapper Pro má ešte aj ďalšiu, pre nás významnú výhodu – dosahuje veľmi dobré výsledky nielen pri sakrálnych stavbách, ale aj pri múzejných artefaktoch, teda drobných predmetoch. Osobnú skúsenosť máme s fragmentmi náhrobných kameňov a drobných predmetov ako napr. busty a pod.

Na základe zvolených fotografií potom softvér automaticky spracuje letovú dráhu z ktorých boli zábery nasnímané – to vidíme na obrázku č. 2.



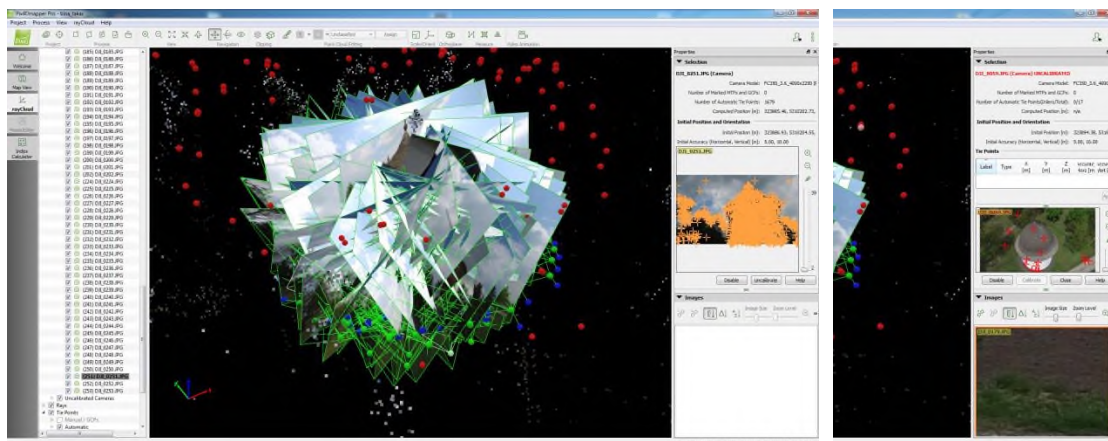
Obrázok 1: Rotunda v Bíni na jednom zo snímkou získaných DJI Phantom



Obrázok 2: Dráha UAV a získané zábery

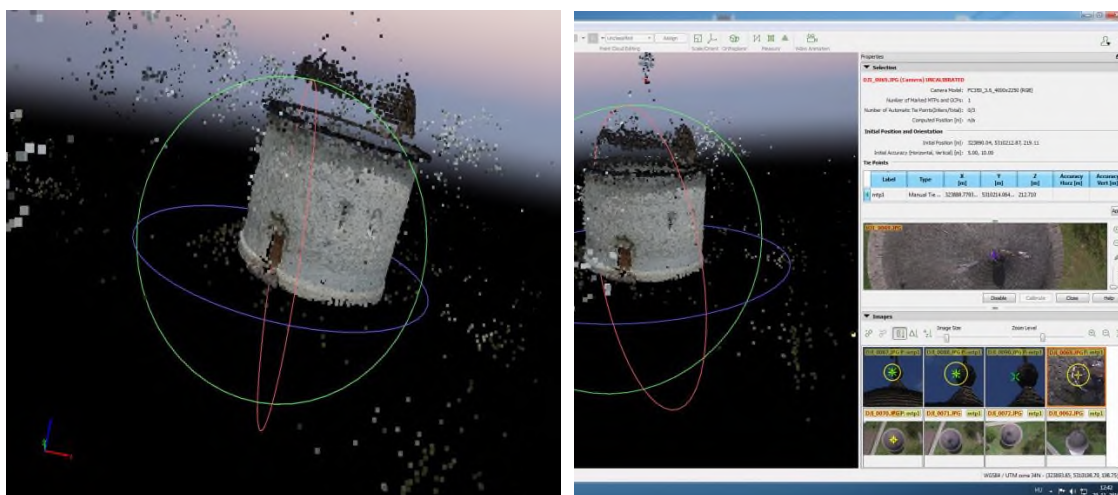
Ako je vidieť, konkrétne fotozábery môžu mať červené, modré a zelené bodové sfarbenie. Zelené body predstavujú reálne použité zábery na tvorbu modelu a často sú prepočítané z pôvodnej pozície – modrá farba. Ilustrujme aspoň dva prípady. Na obrázku č. 3 v ľavo (konkrétne zvolený bod bledozelený) je pohľad na rotundu obsahujúci niekoľko bodov (tmavožltých), ktoré sa objavujú aj na iných záberoch. Softvér tieto body dokázal rozoznať a na viacerých záberoch spojiť. Tým sme využili pohľad na jeden bod modelu z viacerých perspektív, čo už umožňuje aj 3D modelovanie. Červené body predstavujú pohľad na rotundu s veľmi malým alebo žiadnym počtom na iných záberoch

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitná a pedagogická spojitelných bodov. Tieto sa na tvorbe modelu neuplatňujú (konkrétne bledočervený bod na obrázku č. 3 v pravo). Výsledkom ďalšieho spracovania sú tzv. Tie-points – obrázok č. 4. Tie points ešte nepredstavujú mračno bodov. Poskytujú však prvý pohľad na model. Ak je to potrebné, snímky, ktoré boli nespárované – teda červené snímky, môžeme v tomto kroku manuálne pridať.



Obrázok 3: Spracovanie fotozáberov

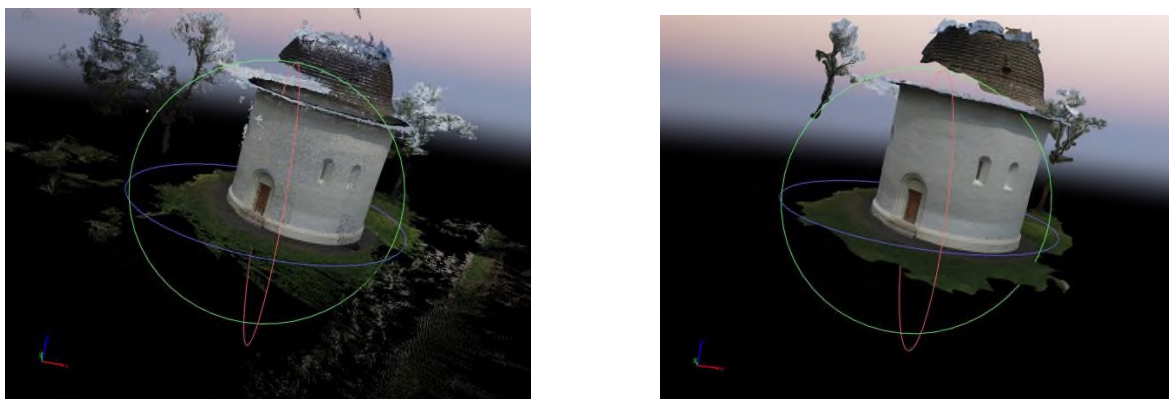
To vykonáme manuálnym označením a pripojením konkrétnych bodov na snímke s bodmi na ostatných snímkach.



Obrázok 4: Manuálne pridanie bodov

3 VÝSLEDKY PRÁCE

Následným spracovaním získame mračno bodov (obrázok č. 5 v ľavo) a vytvorený model (obrázok č. 5 v pravo). Často sa môžeme stretnúť s tým, že vytvorený model nie je hodnoverným obrazom reality ale obsahuje viacero chýb a nepresností. Tieto musíme následným, ale už manuálnym spracovaním odstrániť a opraviť manuálnym pridaním bodov a rezom nevhodných častí modelu. Takýto príklad ilustruje aj už spomínaný obrázok č. 5 v pravo.



Obrázok 5: Mračno bodov (vľavo) a hotový model (vpravo)

Na kvalitu modelu veľmi vplýva počet fotografií z ktorých je model zostavený, kvalita fotografií ale aj prekrytie medzi jednotlivými zábermi ktoré by malo byť čo možno najlepšie. Zo skúseností môžeme tvrdiť, že prekrytie fotografií 2/3 je dostatočné, ale my volíme radšej 3/4. Na nasledovnom obrázku uvádzame už relatívne dostatočne kvalitný model rotundy po manuálnych úpravách.



Obrázok 6: Model rotundy po úpravách

3.1 Možnosti využitia vo výchovno-vzdelávacom procese

Môžeme konštatovať, že dané postupy sú rovnako dobre aplikovateľné na základných, ale hlavne na stredných školách. V súčasnosti už asi ťažko dokážeme vylúčiť prvky IKT z výchovno-vzdelávacieho procesu a každodenného života. To by vôbec nebolo správne, ale práve tieto prvky umožňujú aj ich využitie v oblasti ochrany kultúrneho dedičstva. Nakoľko získanie fotografií zo zeme nie je problém (telefóny, tablety, ...) a ani softvérové vybavenie nie je finančne náročné, teda aj implementácia do vyučovacieho procesu je lacná ale pritom efektívna. Pod efektivitou rozumieme najmä zvýšenie aktivity nielen introvertných žiakov, ktorí sa môžu viac prejavovať, ale hlavne všetkých, ktorí majú pozitívny vzťah k informatike. Tieto postupy taktiež môžu byť vhodné pre žiakov, ktorí majú vizuálno-neverbálny učebný štýl, ale aj ako postupy projektového vyučovania (intuitívnosť, pokus-omyl, ...).

Pri praktickej implementácii do vyučovacieho procesu pravdepodobne bude za potreba aspoň troch hodín informatiky. Na prvej hodine si žiaci predpripravia a upravujú fotografie. Vyberajú zlé a nepoužiteľné fotozábery. Časovo tento krok môže zaberať 20–30 min. Ku koncu hodiny si šikovnejší žiaci môžu vyskúšať aj softvér Pix4D, napr. prípravu rýchleho modelu z niekoľkých málo desiatok záberov, ktorý softvér stihne vygenerovať do konca hodiny. Na druhej hodine dôjde k tvorbe modelu. Pod vedením učiteľa si žiaci môžu vyskúšať tvorbu modelu z kvalitných aj nekvalitných fotozáberov. Model bude nepresný a textúra rozmazaná. Na tretej hodine sa študenti budú venovať úprave modelu.

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitní a pedagogická pomocou *tie points*-ov orežú nepotrebné časti a znova dajú vygenerovať model. Precíznym prístupom vznikne aj precízny model.

Základom sú fotografie, z ktorých sa model bude vytvárať. Stredoškolskí žiaci pod dohľadom dospelého môžu získať fotografie aj z dronov, ktoré sa dajú voľne kúpiť. Tak môže dostať zmysel aj „bezcieľné“ lietanie po sídlisku. Musíme však apelovať na dodržiavanie právnych predpisov.

4 ZÁVER

Tvorba 3D modelov aj vďaka rozšíreniu UAV sa stala veľmi populárnou. Má však aj svoje nezastupiteľné miesto v ochrane kultúrneho dedičstva, najmä v dnešných časoch, kedy mnohé historické, či už sakrálné alebo iné pamiatky chátrajú a nenávratne sa strácajú budúcim generáciám. Práve tvorbou modelov takýchto pamiatok môžeme prispieť k ich uchovaniu ale aj ich výskumu formou, ktorá bez abrazívnych prístupov nie je možná. Môžeme spomenúť napr. meranie hrúbky stien, rozmerov, objemov a pod.

5 PODAKOVANIE

Článok vznikol za pomoci projektu KEGA: 012TTU-4/2018 – Interaktívne animačno-simulačné modely vo vzdelávaní.

6 BIBLIOGRAFICKÉ ODKAZY

- [1] BINA – ROTUNDA. Apsida.sk. 2019. [on-line] (<http://apsida.sk/c/1857/bina-rotunda>).
- [2] FOTOGRAMETRICKÝ SOFTVER PIX4DMAPPER PRO. geopriestor.sk. 2019. [online] (<https://www.geopriestor.sk/produkt/fotogrametricky-softver-pix4dmapper-pro/>).
- [3] FRAŠTIA, M: Fotogrametria v mapovaní, stavebníctve, urbanizme a priemysle. 2019. [online] (https://www.kgk.sk/fileadmin/templates/downloads/Zborn%C3%ADk_refer%C3%A1tov_ku_KS_z_IG/10_Frastia.pdf).
- [4] TRHAN, O – FRAŠTIA, M.: Skúsenosti zo spracovania údajov v aplikáciách Photoscan a UAS Master. In: Technológie priestorového modelovania krajiny a objektov – prístupy a aplikácie. 2014. Bratislava. [on-line] (https://is.stuba.sk/vv/pub_priloha.pl?id=294014).

Algoritmizace a programování ve škole

Algorithm Development and Programming at School

Hravá forma rozvíjania algoritmického myslenia na základnej škole

A Playful Form of Developing Algorithmic Thinking in Primary School

Krisztina Czakoová
Katedra matematiky a informatiky
Univerzita J. Selyeho
Bratislavská cesta 3322
945 01 Komárno
Slovenská republika
czakoovak@uj.sk

Veronika Stoffová
Katedra matematiky a informatiky
Trnavská univerzita v Trnave
Priemyselná 4
918 45 Trnava
Slovenská republika
nikastoffova@seznam.cz

ABSTRAKT

Algoritmické myslenie a jeho rozvoj by sa nemali obmedzovať iba na programovanie – na tvorbu postupu riešenia zvolených úloh a jeho zápis, ale hlavne na riešenie problémov každodenného života. Na rozvoj algoritmického myslenia u žiakov ZŠ je účinné používať metodiku založenú na riešení problémov hravým spôsobom. Žiakov treba usmerňovať tak, aby sa stali kreatívnymi riešiteľmi problémov, experimentátormi a tvorcami alternatívnych riešení. Nato, aby mohli mladší žiaci správne porozumieť spôsobu, ako sa dopracovať k riešeniu pomocou jasne definovaných krokov (preddefinovaných funkcií), dobre poslúžia didaktické hry. Pri hraní hier žiak dostáva okamžitú spätnú väzbu, a tak sám môže korigovať svoje kroky, ktoré vedú ku správne riešeniu. Po zvládnutí správneho algoritmu môže interaktívne naprogramovať aj postup riešenia, s možnosťou vizualizácie sledu krokov, pomocou základných riadiacich štruktúr, ktoré má k dispozícii. Cieľom je, aby si žiaci osvojili nové poznatky na báze konštruktivismu a učili sa robením. Hlavným poslaním príspevku je prezentovanie počítačovej hry, ktorá bola inšpirovaná dobre známou programovateľnou hračkou (robotickou včielkou) Bee-Bot. Pri tvorbe a implementácii hry sme sa snažili dodržiavať hlavné didaktické zásady ako sú: názornosť, primeranosť, vedeckosť, aktívny a individuálny prístup.

ABSTRACT

Algorithmic thinking and its development should not be limited only to programming - to the process of solving selected tasks and its notation, but mainly to solving problems of everyday life. It is effective to use a problem-based methodology in a playful way to develop algorithmic thinking of pupils in primary school. Pupils need to be directed to become creative problem solvers, experimenters and creators of alternative solutions. In order for younger pupils to understand correctly, how to get a solution using clearly defined steps (predefined functions), didactic games will serve well. When playing games, the pupil receives immediate feedback, so he/she can correct his/her steps to get the right solution. After mastering the correct algorithm can be programmed and interactive process solutions with the possibility of visualizing a sequence of steps, using basic control structures at its disposal. The aim is that pupils acquire new knowledge based on constructivism and learning by doing. The main intention of the paper is to present a computer game that was inspired by the well-known programmable toy Bee-Bot. During the creation of the game we tried to comply with the didactic principles of illustratinality, appropriaty and individual approach.

Kľúčové slová

Algoritmy, programovanie, programovateľné hračky, edukačné počítačové hry, učenie sa robením.

Keywords

Algorithms, programming, programmable toys, educational computer games, learning by doing.

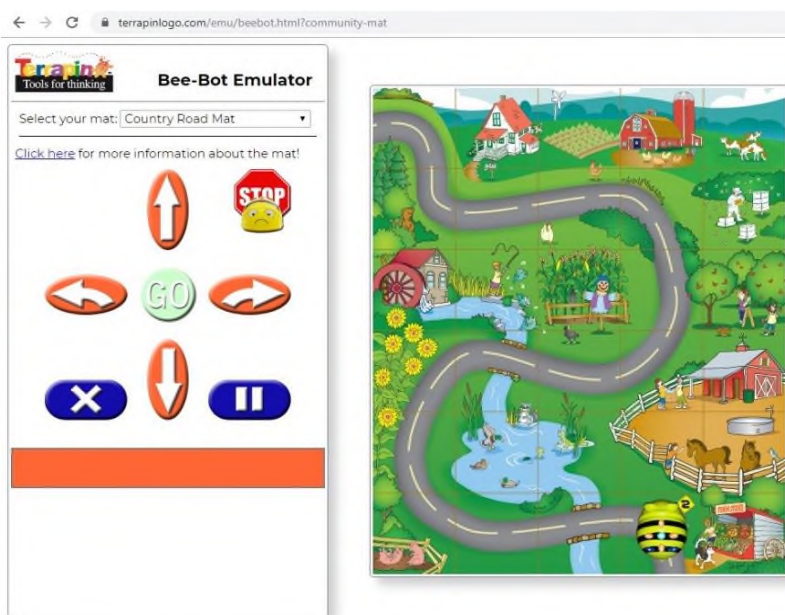
1 ÚVOD

Hra má vo vývoji dieťaťa veľký význam. Cez hru dieťa objavuje svet – svoje bezprostredné okolie, predmety a osoby v ňom. Najprv ich len pozoruje, potom sa ich dotýka a rôznym spôsobom prejavuje vzťah k nim. Hra sprevádza človeka celý život. Služi na rozvoj motorických schopností, ku ktorým sa pridruží aj rozvoj kognitívnych schopností.

2 PROGRAMOVATEĽNÉ HRAČKY

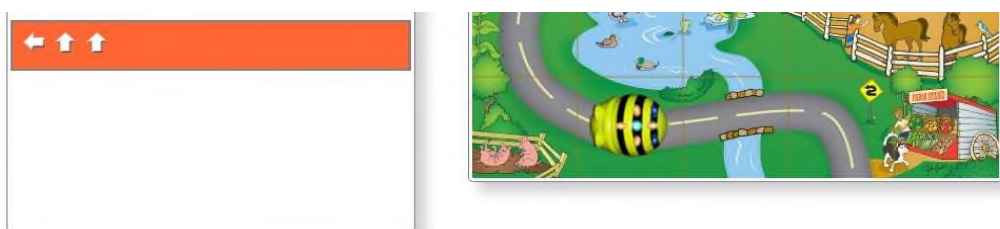
Hračkársky priemysel a samotný trh ponúka nekonečné množstvo hračiek. Hračky možno triediť podľa rôznych kritérií. Napríklad podľa vekovej kategórie, podľa poslanca, aj podľa toho, aké vlastnosti a schopnosti rozvíja a podobne.

My Vám predstavíme len tri, ktoré často sú distribuované ako interaktívny balík. Táto kombinácia zábavných robotov je vhodná pre rôzne vekové kategórie, s rôznymi schopnosťami a skúsenosťami. Pomocou nich sa žiaci učia symbolický zápis, akési kódovanie hravým spôsobom, pričom si rozvíjajú aj logické a programátorské myslenie. Umožňujú vytváranie aj dlhších a komplexnejších programov, vytváranie a integrovanie procedúr a senzorov do programu, ako aj kreslenie tvarov a návrhov. Disponujú širokou škálou vstupov a výstupov pre program, ktorý po napísaní sa spustením okamžite vykoná.



Obrázok 1: Bee-Bot Emulátor (Začiatočná pozícia včielky: Štart)

Pri tvorbe vlastnej aplikácie na tvorbu a rozvíjanie algoritmického myslenia a základných programátorských zručností detí – žiakov základných škôl nás predovšetkým inšpirovala robotická včielka, najmä jej zábavné programovacie rozhranie – voľne a on-line dostupný Emulátor. Vykonanie jednoduchého programu je založené na zadávaní sekvencie základných inštrukcií, pomocou tlačidlových príkazov [5]. (Pozri obrázky 1 a 2).



Obrázok 2: Bee-Bot Emulátor (Pozícia včielky po vykonaní krátkeho programu)

2.1 Bee-Bot

Pomocou tohto jednoduchého interaktívneho nástroja môžu žiaci hravým spôsobom získavať nové vedomosti, v rôznych oblastiach. Je vhodná na identifikáciu písmen a čísiel, na precvičovanie orientácie v rovine a pod. Vďaka tvorbe sekvencie „tlačidlových“ príkazov na riadenie „včielky“ sa u detí rozvíja logické myslenie, orientácia v priestore a schopnosť plánovania hravou formou.



Obrázok 3: Programovateľná hračka Bee-Bot

Na ovládanie robotickej včielky vieme použiť priame inštrukcie, ktoré sú reprezentované riadiacimi tlačidlami. (Pozri obrázok 3). Bee-Bot má ľahko ovládateľné používateľské rozhranie. Do jej pamäte možno uložiť až 40 príkazov (inštrukcií) na vykonanie. Pozná nasledujúce príkazy: pohybovať sa vpred/vzad o danú jednotku (cca. 15 cm), otočiť sa o 90 stupňov doľava/doprava, čakať danú časovú jednotku (cca. 1 s). Pokyny zadané pomocou ovládacích tlačidiel je možné vykonať pomocou tlačidla *Go*, ktorý slúži na spustenie vykonania sekvencie príkazov. V prípade nového algoritmu sa na vymazanie príkazov z pamäte dá použiť tlačidlo *Clear*. Ku hre sa dajú vytvoriť/navrhnúť hracie dosky, na základe štvorcovej mriežky (15cm x 15cm). Na ploche zo štvorcov sa môžu včielky pohybovať a vykonávať pokyny hráča/programátora.

Žiaci môžu porozprávať svoje vlastné príbehy o včielke, čím sa rozvíja ich kreativita a fantázia. Úspech pri ovládaní včielky má aj motivačný účinok na riešenie ďalších zaujímavých úloh, čo zároveň pomáha ďalej rozvíjať a upevniť základy logického a algoritmického myslenia.

2.2 Pro-Bot

Pro-Bot v tvare autíčka (na obrázku 4) je veľký brat robotickej včielky Bee-Bot, ktorý vie kresliť čiary po podložke, prostredníctvom pera, uchyteného do zabudovaného držiaka, a má aj LCD displej. Okrem toho, že sa vie pohybovať dopredu a dozadu v programovateľných vzdialenostiach a otáčať sa vpravo a vľavo v programovateľných uhloch, umožňuje zložitejšie programovanie ako včielka Bee-Bot.



Obrázok 4: Programovateľné autíčko Pro-Bot

Pro-Bot má zabudované programovateľné dotykové senzory v nárazníkoch, a tiež senzor svetla a senzor zvuku. Môže byť pripojený k počítaču cez USB, s možnosťou vytvárania programov na počítači, pomocou samostatne dodávaného softvéru *Probotix*, a teda môže byť ovládaný pomocou programovania, s využitím procedúr a cyklov. Je však použiteľný a programovateľný aj bez počítača.

Zápis a editovanie programu možno vykonať aj pomocou LCD displeja. Balík obsahuje aj simulačný program, ktorý možno použiť na plánovanie trasy, alebo na vytvorenie obrázka, ktorý má byť vykreslený.

2.3 InO-Bot

InO-Bot (na obrázku 5) bol špeciálne navrhnutý tak, aby pokrýval čo najviac programovacích lekcií, a tým prispel k rozvíjaniu programovacích schopností žiakov. InO-Bot sa ovláda pomocou populárneho programovacieho jazyka Scratch alebo iOS aplikáciou (ktoré sú dostupné zadarmo). Žiaci môžu navrhovať a vytvárať zábavné, zaujímavé a nápadité aktivity.

Vďaka širokej škále vstupov a výstupov programu, žiaci dokážu vytvoriť sofistikované a inteligentné



Obrázok 5: InO-Bot – programovateľný podlahový Bluetooth robot

programy. InO-Bot sa pripája k počítaču pomocou Bluetooth rozhrania, ktoré umožňuje obojsmernú bezdrôtovú komunikáciu. Vďaka tomu sú viditeľné výstupy v reálnom čase. Má štyri senzory priblíženia v rohoch, senzor merania vzdialenosti, senzory na sledovanie nakreslenej čiary. Ak vložíme do držiaka pero, dokáže svoju cestu aj nakresliť. Priehľadné púzdro chráni nielen komponenty, ktoré sú zabudované vo vnútri, ale umožní sledovať procesy, ktoré sa vo vnútri robota odohrávajú.

3 HRAVÁ FORMA ROZVÍJANIA ALGORITMICKÉHO MYSLENIA

U žiakov nižších ročníkov základných škôl sú vyššie opísané programovateľné hračky vhodnými nástrojmi na výučbu základov programovania, informatiky a niektorých tematických celkov z matematiky. Použitie rôznych hracích podložiek, máp či obrázkov umožňuje deťom venovať sa rôznym aktivitám. Existuje nespočetne veľa možností, ako využiť schopnosť robotickej včielky premiestniť sa z jedného bodu do druhého, pomocou sekvencie príkazov. Práve táto skutočnosť dáva priestor na objavovanie a skúmanie problému (na analýzu problému podľa [3]), a hľadanie možného správneho riešenia interaktívnou implementáciou možného riešenia. Žiakom bude zrejmé, akým spôsobom ovplyvňuje výsledok (výstup) sled príkazov. Budú schopní sa rozhodovať o výbere príkazov, prípadne o zmene príkazu/-ov, za účelom dosiahnutia nových cieľov [1].

U žiakov vyšších ročníkov základných škôl práca s robotickou včielkou sa môže využiť na potvrdenie fungovania riadiacich štruktúr v riešení problému. Prehľbia si poznatky o tom, ako funguje cyklus, vetvenie či použitie premenných v riešení problému. Práve táto možnosť nás inšpirovala na vytvorenie vlastnej počítačovej hry s edukačným účelom podľa [2], pre žiakov základnej školy na podporu rozvíjania algoritmického myslenia a uplatnenia základných riadiacich štruktúr v programovaní.

Aplikácia sa zrodila ako súčasť diplomovej práce študenta študijného odboru Učiteľstvo akademických predmetov, aprobácie Anglický jazyk a literatúra – Informatika, pod odborným vedením jednej z autoriek článku [4].

4 DIDAKTICKÁ POČÍTAČOVÁ HRA NA ROZVÍJANIE ALGORITMICKÉHO MYSLENIA

Nižšie prezentovaná hra bola navrhnutá tak, aby jeho aktívnym používaním žiaci pochopili fungovanie základných riadiacich štruktúr, a to lineárne sekvencie príkazov, vetvenie (rozhodovanie sa medzi viacerými možnosťami) a cyklus (s daným počtom opakovaní). Počítačová hra má tri úrovne náročnosti. Gradované úlohy nasledujú vždy po sebe. Po vyriešení úlohy nižšej úrovne nasleduje úloha náročnejšia. Grafické interaktívne rozhranie umožňuje výber ponúkaných príkazov a ich radenie do sekvencie, následnú implementáciu, realizáciu a vyhodnotenie zostaveného programu [2, 4].

Ikony ovládacích prvkov na výber príkazov (povelov pre pirátske dievča alebo pirátskeho chlapca) sa nachádzajú v ľavom dolnom rohu obrazovky (pozri obrázok 6). Pripomínajú ovládacie tlačidlá programovateľných hier (smer chodu: dopredu, dozadu; otočenie sa o 90 stupňov smerom: doľava, doprava; Vymazanie inštrukcie (*Törles*) – slúži na opravu kroku/-ov; *Start* – vykoná spustenie programu). Nakoľko ide o strategickú hru (treba nazbierať čo najviac bodov/zlata počas celej hry), voľba vymazania inštrukcie a jej následná oprava spôsobuje stratu určitého počtu bodov. Preto je dôležité, aby hráč sa snažil čo najpresnejšie zadať príkazy na efektívne prejdienie trasy a nájdenie pokladu.

Program, vyjadrený postupom značiek zvolených tlačidiel, sa zobrazuje v pravom hornom rohu hracej plochy. Je tu uvedené aj grafické vyhodnotenie správnosti riešenia. Po spustení programu sa zadané inštrukcie postupne vykonávajú a správne zvolené inštrukcie sa prefarbia na zelenú (pozri obrázok 6). Nesprávne zvolené inštrukcie sa ukážu červenou farbou, ktoré je treba na postup v hre na ďalšiu (vyššiu) úroveň opraviť.



Obrázok 6: Počítačová hra – 1. úroveň (sekvencia príkazov)

Pri zobrazení vykonaných krokov v pravom hornom rohu vidieť, že každá riadiaca štruktúra má iné farebné vyobrazenie, čím sa lepšie znázorňuje aj štruktúra programu. Inštrukcie zapísané pomocou daného počtu opakovaní, ovládané posuvníkom (pozri obrázok 8, s nastaveným počtom opakovaní pre následne zadaný príkaz: 2x) sa zobrazujú v zápise programu modrou farbou, ostatné (po jednom zvolené, teda zadané sekvenčne) sivou. Zelená farba znázorňuje už vykonanú časť programu po stlačení tlačidla *Start* – teda stav po vykonaní určitého počtu krokov na základe zapísaných inštrukcií

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitná a pedagogická
(teda, prefarbia sa tie, ktoré už boli vykonané a paralelne aj znázornené animáciou – pohybom hráča/piráta). (Pozri obrázok 7.)



Obrázok 7: Počítačová hra – 2. úroveň (opakovanie príkazov – cyklus s daným počtom opakovaní)

Na obrázku 8 je cesta labyrintom rozšírená o prekážky. Takto sa hráč musí rozhodovať pri voľbe správnej cesty k pokladu, a to s najmenším rizikom stretnúť sa s pirátom. Pri rozhodovaní hráč má možnosť si zvoliť kratšiu cestu, kde mu cestu skrížia piráti, za cenu bodovej pokuty. Pri voľbe dlhšej cesty však dokáže obísť väčšinu pirátov, a bude bojovať iba s jedným, pričom mu zostane viac bodov (zlata). Hra sa končí, keď hráč získa aj hlavný poklad. Po dosiahnutí cieľa sa hráč dostane na stránku vyhodnotenia výkonu, kde ho čaká menný zoznam hráčov s najlepším dosiahnutým výkonom. Tu sa zaznamená aj jeho výkon.



Slovenský význam textu na ovládacom paneli:

Ismétlődés – opakovanie (s posuvníkom – na zadanie počtu)

Előre lép – krok vpred (aktuálne zvolená inštrukcia pre opakovanie)

Törles – vymazanie

Start – štart (spustenie programu na vykonanie)

Obrázok 8: Počítačová hra – 3. úroveň hry (s legendou ovládania hry)

Cieľom a poslaním hry je posilnenie rozvoja logického a algoritmického myslenia a schopnosti riešiť problémy u žiakov základných škôl. Medzi našimi cieľmi bolo aj to, aby sme doplnili hru o také elementy a funkcie, ktorými robotická včielka Bee-Bot ani jeho on-line emulátor nedisponujú, napríklad o možnosť plynulého sledovania behu programu (farebným vyobrazením vykonaných ako aj nesprávne zvolených inštrukcií, sprevádzané náležitým zvukovým efektom). Zvýšili sme aj počet vykonateľných inštrukcií (zo 40 na 100). Rozšírili sme okruh inštrukcií o možnosť ich opakovania (pridali sme cyklus s daným počtom opakovaní, ktorý sa ovláda pomocou posuvníka), ako aj o možnosť vymazania (opravu) zle zadaného príkazu pred spustením vykonania programu. Nie je nutné vymazať celú pamäť a začať programovanie od začiatku, ako je to u programovateľnej včielky Bee-Bot. Je možné vymazať a opraviť posledné príkazy. Hra je personalizovaná pre hráčku ako aj hráča. Na základe vopred zadaného mena hráča je hráč oslovený a inštruovaný o cieľoch a úlohách hry adresne podľa mena.

Hra má viac správnych riešení, avšak má jedno optimálne riešenie, ktoré nastane vtedy, ak hráč nazbiera najviac bodov (zlata). To sa dosiahne tým, že sa vždy zvolia z ponuky efektívne riadiace konštrukcie. Pozitívne sa hodnotí napríklad uplatnenie cyklu namiesto sekvencie rovnakých príkazov (zapísaných po sebe).

Hra bola implementovaná ako Windows Form aplikácia, vo vývojárskom programovacom prostredí Visual Studio 2015 v jazyku C#. Bohaté grafické elementy boli vytvorené pomocou grafického programu Paint.net.

5 ZÁVER

Učenie je činnosť, v ktorej edukant zámerne získava určité vedomosti a schopnosti. Základným cieľom alternatívnych vzdelávacích stratégií je, aby žiak nebol iba pasívnym prijímateľom poznatkov, ale aby sa aktívne zapojil do poznávacieho procesu. Osvojenie si nových poznatkov môže byť aj zábavou, môže sa stať príjemným zážitkom, a z toho vyplývajú aj vnútorným presvedčením, ku ktorému sa žiak dopracuje hravou formou. Takýto spôsob učenia sa je založený na vlastných skúsenostiach, podporovaný metódami zvedavosťou a objavovaním riadeného učenia sa [1].

Naším zámerom bolo vytvoriť takú interaktívnu aplikáciu, ktorá umožňuje vyučovanie programovania na základnej škole na základe vlastných skúseností a objavovania, a to všetko spontánnou činnosťou, hravou a zábavnou formou.

6 POĎAKOVANIE

Článok vznikol s podporou projektu KEGA 012TTU-4/2018 Interaktívne animačno-simulačné modely vo vzdelávaní.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] CZAKÓOVÁ, Krisztina. Élmény alapú programozás oktatás. In. Bukor, J. – Gubo, Š. (eds.): *Zborník medzinárodnej vedeckej konferencie Univerzity J. Selyeho – 2017 „Hodnota, kvalita a konkurencieschopnosť – výzvy 21. storočia“*. Sekcia informatických vied a IKT. Proceeding of electronic version of reviewed contributions (CD-ROM). Komárno: Univerzita J. Selyeho, 2017. s. 27–33. ISBN 978-80-8122-221-4.
- [2] CZAKÓOVÁ, Krisztina. Interaktív alkalmazások és oktatóprogramok fejlesztése. In. Juhász, Gy. – Horváth, K. – Strédl, T. – Árki, Z. (eds.): *Zborník z medzinárodnej vedeckej konferencie Univerzity J. Selyeho – 2014 „Vzdelávanie a veda na začiatku XXI. storočia“*: Sekcie pedagogických vied. Proceeding of electronic version of reviewed contributions (CD-ROM). Komárno: Univerzita J. Selyeho, 2014. s. 462–466. ISBN 978-80-8122-103-3.
- [3] STOFFOVÁ, Veronika – CZAKÓOVÁ, Krisztina. A playful form of teaching and learning using micro-world-based applications In. *Proceedings of the 15th International Scientific Conference: “eLearning and Software for Education: New technologies and redesigning*

DIDINFO 2020, Technická univerzita v Liberci, Fakulta přírodovědně-humanitní a pedagogická *learning spaces*. Volume 1, DOI: 10.12753/2066-026X-19-014, 2019/1, p. 110-115. Bucharest: "CAROL I" National Defence University Publishing House, 2019. ISSN 2066-026X, ISSN-L 2066-026X, ISSN CD 2343 – 7669. (WoS).

- [4] SZALAI, Dávid. *Rozvíjanie algoritmického myslenia a riešenia problému pomocou interaktívnych hier na ZŠ*. Diplomová práca, Komárno: Univerzita J. Selyeho, Pedagogická fakulta. Vedúca práce: PaedDr. Krisztina Czakoová, PhD. 2017, 62 p.
- [5] TERRAPINLOGO. *Tools for thinking: Bee-Bot Emulator* [online]. [vid. 5. 12. 2019]. Dostupné na: <https://www.terrapinlogo.com/emu/beebot.html?community-mat>.

Ako na grafy vo vyučovaní informatiky na druhom stupni základných škôl

How to Include Graphs in Informatics Education in Lower Secondary Schools

Lucia Budinská
KDMFI, FMFI UK
Mlynská dolina 1
842 48 Bratislava
Slovensko

Lucia.Budinska@fmph.uniba.sk

Karolína Miková
KDMFI, FMFI UK
Mlynská dolina 1
842 48 Bratislava
Slovensko

Karolina.mikova@fmph.uniba.sk

ABSTRAKT

Grafy ako dátové štruktúry by mali byť súčasťou vyučovania na základných školách. Keďže sa úlohy s grafmi často vyskytujú v súťaži iBobor, rozhodli sme sa na základe výsledkov predchádzajúceho výskumu vytvoriť niekoľko pracovných listov. Tie majú žiakov druhého stupňa ZŠ zoznámiť s rôznymi typmi grafov, s ktorými by sa mohli stretnúť aj v reálnom živote (napríklad rodostromy, mapy tratí MHD a diagramy). V článku analyzujeme výsledky pilotného testovania troch pracovných listov so žiakmi prímý a sekundy osemročného gymnázia, pričom sa zameriavame na časté miskoncepce, problematické časti úloh, ale aj na vplyv motivácie úlohy na riešenia žiakov a ich záujem.

ABSTRACT

Graphs as data structures should be a part of informatics lesson in the lower secondary education. While graph tasks are frequently used in Slovak Bebras competition (called iBobor), we have decided to create some worksheets using results from previous research. In these worksheets, pupils get to know the different types of graphs which they could find in the real-life situations (e.g. family trees, transportation lines map, finite state automata). In this paper, we analyse results of pilot testing of three worksheets with pupils attending first and second grade of secondary grammar school. We talk about common misconception, problematic parts of tasks and task motivation effect on pupils' solution and their interest.

Kľúčová slová

Pracovné listy, grafové úlohy, motivácia, súťaž iBobor, 2. stupeň ZŠ.

Keywords

Worksheets, graph tasks, motivation, bebras contest, lower secondary education.

1 ÚVOD

Informatická súťaž iBobor [1] neslúži iba ako jednorazová akcia pre žiakov na spopularizovanie informatiky a stretnutie sa s rôznorodými informatickými úlohami. Práve kvôli veľkému množstvu každoročne použitých úloh a rastúcemu počtu zapojených škôl a žiakov vie súťaž pomôcť ako učiteľom, tak aj výskumníkom. Výskumníci môžu skúmať žiacke riešenia, zisťovať náročnosť úloh v závislosti od ročníkov, alebo úspešnosť úlohy pre nejaké skupiny žiakov. Učitelia zas môžu využívať úlohy z minulých ročníkov súťaže na vyučovaní, či už ako zbierku úloh na zoznámenie žiakov s určitou problematikou, alebo na ozvláštnenie hodiny. Využitie úloh na hodine s presným cieľom však od učiteľov vyžaduje ďalšiu prácu. Na Slovensku síce existuje systém Bobrovo [2], ktorý

učiteľom umožňuje vyhľadávať staré úlohy podľa rôznych kritérií a vytvárať z nich vlastný test pre svojich žiakov, ktorý riešia priamo v systéme, takýto prístup ale ponúka iba jeden pohľad na každú z úloh a nereflektuje zistenia zo súťaže. Aké? Napríklad, že úloha bola ťažšia, ako zadávatelia predpokladali, že zadanie úlohy nebolo žiakom dostatočne jasné, ale aj to, že niektoré úlohy boli zaujímavejšie pre chlapcov a iné pre dievčatá. Rozhodli sme sa preto skúsiť spojiť výsledky z výskumov s potrebami učiteľov pri vyučovaní informatiky.

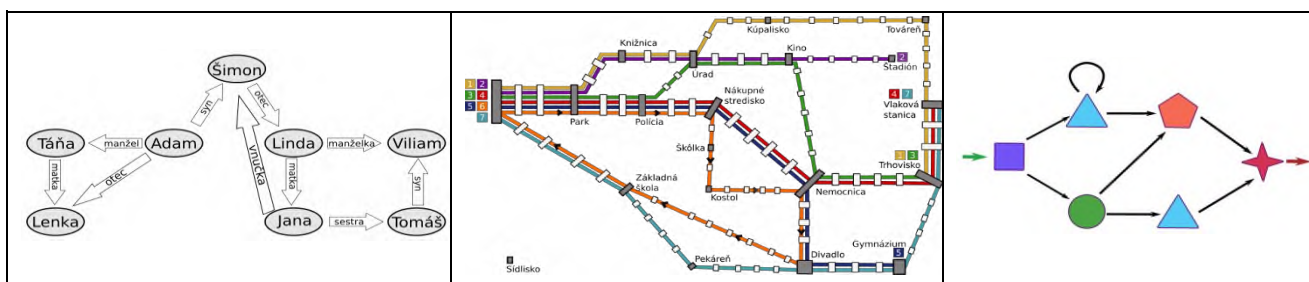
V predchádzajúcom výskume [3] sme skúmali skupinu úloh zo súťaže iBobor, ktoré boli zamerané na grafové dátové štruktúry a grafové algoritmy. Tieto úlohy sme kategorizovali podľa rôznych kritérií ako je téma, motivácia, dĺžka zadania, distraktory, spôsob riešenia úlohy atď. a objavili sme štatisticky významné rozdiely medzi úspešnosťou riešení u chlapcov a u dievčat. Zaujímalo nás, aké sú príčiny týchto rozdielov a či nám vedia povedať o úprave grafových úloh, aby boli zaujímavé a vhodne formulované pre všetkých žiakov.

Rozhodli sme sa preto nájsť vhodný spôsob na rozšírenie týchto úloh, aby mohli pre nás slúžiť ako nástroj na podrobnejšie skúmanie tejto témy, a zároveň aj ako nástroj na využitie úloh zo súťaže v škole. Keďže súťaž Bebras je medzinárodná, pozreli sme sa na prístupy využitia úloh v iných krajinách. V Litve napríklad vytvorili zo súťažných úloh hracie karty [4]. Každá obsahovala zadanie jednej úlohy a krátku informáciu o tom, v čom je úloha informatická. Žiaci riešili úlohy na kartičkách samostatne alebo v malých skupinách a pri riešení si mohli pomáhať aj vytváraním fyzických pomôcok – či už vystrihovaním z papiera, alebo použitím kociek na simulovanie objektov z úlohy. Pilotná štúdia z Litvy ukázala, že na prvom stupni základných škôl „sa učiteľom aj žiakom páčilo riešenie úloh a diskusie o rôznych témach spojených s informatickými konceptmi“. Úlohy motivovali aj učiteľov, ktorí nemali informatické vzdelanie, aby sa venovali informatike na vyučovaní a mnohí z nich začali vytvárať ďalšie variácie úloh – najmä ako fyzicky manipulovateľné objekty.

V Taliansku zas motivovali žiakov šiesteho ročníka k skúmaniu úloh zo súťaže tak, že v skupinkách vytvárali plagáty a hry s témou úlohy, ktorú mali v súťaži vyriešenú nesprávne [5]. Svoje výrobky následne prezentovali žiakom tretieho ročníka, čiže museli úlohu zjednodušiť, resp. prispôbiť mladším žiakom. Ukázalo sa, že takáto práca pomáha obom skupinám žiakov. Šiestaci skúmajú informatický koncept z rôznych uhlov a snažia sa vytvárať rôzne pomôcky (interaktívne) a vysvetlenia, čím sa viac učia. A pre tretiakov je nápomocný práve jazyk iba o čosi starších žiakov.

2 PRACOVNÉ LISTY

Keďže sme chceli vytvorené materiály používať aj ako výskumný nástroj, zamerali sme sa na grafové úlohy zo súťaže a rozhodli sa vytvoriť z nich pracovné listy. Ich vývoj prebiehal vo viacerých fázach. Najprv sme identifikovali úlohy, ktoré boli vhodné pre nižšie ročníky druhého stupňa základnej školy a zároveň spĺňali požiadavky inovovaného Štátneho vzdelávacieho programu (ŠVP) z informatiky [6]. Pre každú z úloh sme zisťovali jej úspešnosť v súťaži a snažili sa identifikovať problematické časti – či už konkrétnu časť zadania, alebo grafické zobrazenie dátovej štruktúry. Na to sme využili kvalitatívny výskum, opísaný v [7], pri ktorom sme našli problematické oblasti každej úlohy a navrhli postupnosť úloh, berúc do úvahy aj ich kognitívnu náročnosť. Vznikli tak tri pracovné listy. Celý postup ich vytvárania je popísaný v článku [8]. Tieto pracovné listy sme sa rozhodli otestovať, aby sme overili, či sú úlohy vhodné na použitie v škole, či sa žiaci pri ich riešení učia a či existujú rozdiely v riešení medzi chlapcami a dievčatami.



Obrázok 1: Grafické znázornenie typov grafov použitých v úlohách. **Vľavo:** Rodostromy. **V strede:** Linky MHD. **Vpravo:** Náramkový automat.

2.1 Prvý pracovný list

Prvý pracovný list má názov *Rodostromy* a venuje sa orientovaným grafom, v ktorých sú zaznamenávané informácie o vzťahoch v rodine, vid' Obrázok 1 vľavo. Okrem informatiky tak prepája aj učivo občianskej náuky a slovenského jazyka, čím bola téma žiakom blízka. Pracovný list obsahuje 11 postupne gradujúcich úloh, pričom posledná je nepovinná a je určená žiakom, ktorí považovali úlohy za zaujímavé a sú kreatívni. Žiaci v úlohách najprv len čítajú rodinné vzťahy priamo znázornené v grafe, potom aj zo zložitejších zápisov, v ktorých je treba informácie odvodzovať. V ďalších úlohách dopĺňajú do grafu vzťahy aj osoby, alebo vyberajú grafy, ktoré spĺňajú zadané podmienky, alebo podľa zadania musia graf nakresliť. V siedmej a ôsmej úlohe riešia rovnaké otázky (len sú zmenené mená a poradie otázok), ale v jednej úlohe je graf nakreslený a v druhej iba slovne opísaný. V závere sa ich pýtame, ktorá úloha sa im riešila ľahšie a prečo. V poslednej povinnej úlohe majú žiaci nakresliť rodostrom pre vlastnú alebo vymyslenú rodinu a napísať k nemu krátky opis. Nepovinná úloha je zameraná na vlastný spôsob zápisu či zakreslenia rodinných vzťahov, keďže pri riešení úloh v pracovnom liste sa stretli s jedným možným zakreslením.

2.2 Druhý pracovný list

V druhom pracovnom liste s názvom *Linky MHD* žiaci pracujú s mapou liniek MHD. V mape sú zakreslené čísla liniek, zastávky, pričom niektoré z nich majú aj názvy, vid' Obrázok 1 v strede. Táto úloha je priamo spojená s reálnym životom, keďže s podobnou mapou sa žiaci majú možnosť stretnúť, či už priamo v meste, kde bývajú, alebo pri návšteve iných väčších (aj zahraničných) miest. Pracovný list má sedem úloh. Prvá úloha je zameraná na zopakovanie názvov zo zadania úlohy (zastávka, jednosmerná linka) a overenie ich porozumenia. V druhej úlohe žiaci hľadajú konkrétne linky podľa rôznych podmienok (cieľová zastávka, počet zastávok na trase...), alebo dopĺňajú údaje do krátkych príbehov (číslo linky, názov zastávky, počet zastávok...). V ďalšej úlohe navrhujú trasu z miesta A na miesto B podľa rôznych kritérií a v nasledujúcej úlohe navrhujú prestupovanie z linky na linku podľa zložitejších príbehov. V poslednej úlohe navrhujú vlastnú linku a hodnotia, čo je pre nich v bežnom živote lepšie – či prestupovať a byť niekde rýchlejšie, alebo radšej ísť iba jednou linkou, hoci dlhšie.

2.3 Tretí pracovný list

V poslednom pracovnom liste nazvanom *Náramkový automat*, sa žiaci stretli s pravidlami pre tvorbu náramkov zapísaných pomocou geometrických tvarov a šípok, pozri Obrázok 1 vpravo. Z informatického pohľadu ide o zjednodušené zobrazenie konečného stavového automatu. V tomto pracovnom liste sa už pridáva, okrem čítania grafovej štruktúry, aj algoritimizácia a to v podobe čítania pravidiel a overovania správnosti. Tento pracovný list obsahuje osem úloh. Prvá je zameraná na ozrejmienie spôsobu, akým automat pracuje – žiaci majú do diagramu nakresliť „cestu“, akou automat vytvára daný náramok. V ďalšej úlohe dopĺňajú chýbajúce koráliky do náramkov a v ďalšej vyberajú tie, ktoré automat mohol vytvoriť. Neskôr pre viaceré náramky určujú, ktorý z dvoch automatov ich vytvoril a následne majú vytvárať vlastné náramky pre iný automat. Sem sme pridali

aj úlohu, v ktorej sa majú zamyslieť nad spôsobom tvorby náramkov, aké budú najkratšie náramky a aké budú najdlhšie (zamerané na pochopenie slučky). V predposlednej úlohe dopĺňajú do automatu koráliky, podľa vytvorených náramkov. Posledná úloha je zameraná na vlastnú tvorbu, kde žiaci môžu vytvoriť ľubovoľný automat a niekoľko jeho „výrobkov“.

V pracovných listoch Linky MHD a Náramkový automat sme na koniec pridali aj šesť otázok na hodnotenie pracovného listu. Pýtali sme sa v nich, či žiaci niekedy riešili podobné úlohy, či sa stretli s podobnými grafmi, ktorá úloha bola pre nich najľahšia resp. najťažšia a ktorá úloha sa im páčila najviac a ktorá najmenej. A na záver mali na päťbodovej škále určiť náročnosti úloh v pracovnom liste. Na poslednej overovacej hodine sme žiakom dali dotazník, kde sme sa pýtali na všetky tri pracovné listy. Keďže niektorí z nich sa nezúčastnili všetkých troch overovaní, najprv sme sa pýtali, ktorých testovaní sa zúčastnili, a potom na ich názor na to, ktorý pracovný list bol najľahší resp. najťažší, a ktorý sa im najviac resp. najmenej páčil.

3 METÓDY VÝSKUMU

Výskum prebiehal na osemročnom gymnáziu v Bratislave, v triede prímý (6. ročník ZŠ, 11–12 roční žiaci) a sekundý (7. ročník ZŠ, 12–13 roční žiaci). Každá trieda mala dve 45 minútové hodiny informatiky za sebou. Počas prvej hodiny riešili pracovný list, počas druhej hodiny sa so žiakmi jednotlivo rozprávali výskumníčky a zvyšní žiaci zatiaľ riešili iné úlohy podľa zadania učiteľa. Výskumu sa zúčastnilo 12 žiakov prímý (4 chlapci a 8 dievčat) a 9 žiakov sekundý (6 chlapci a 3 dievčatá). Na prvej výskumnej hodine niektorí žiaci chýbali, preto sa počet riešení prvého a zvyšných vyriešených pracovných listov líši.

Pri prvom pracovnom liste (Rodostromy) riešili pracovný list všetci žiaci individuálne a následne, po vyriešení úloh, sa s nimi individuálne rozprávala jedna z výskumníčok. Pri zvyšných dvoch pracovných listoch bol vždy vybraný jeden alebo dvaja žiaci, ktorí počas riešenia komentovali svoj postup a tak boli nahrávaní na audiozáznam. Výskumníčka sa pýtala iba doplňujúce otázky pre potreby audionahrávky. Takéto riešenie prebiehalo v inej miestnosti, a následne, keď žiak a aj zvyšok skupiny pracovné listy doriešili, výskumníci opäť viedli individuálne rozhovory so žiakmi o výsledkoch úloh ako pri prvom pracovnom liste.

Pri výskume sme tak zozbierali rôzne typy dát, hlavne kvalitatívne [9]. Jednými sú zápisky výskumníkov, žiacke artefakty, čiže riešenia pracovných listov, ďalšími sú komentované riešenia týchto pracovných listov, kde je zaznamenané, čo a ako žiaci robili a čomu ako (ne)porozumeli. Potom sú to pološtruktúrované rozhovory so žiakmi po riešení pracovných listov, a ešte aj spätná väzba k úlohám (v prvom pracovnom liste je súčasťou rozhovorov, vo zvyšných dvoch ako súčasť pracovného listu).

Pre potreby kvalitatívneho analyzovania dát sme najprv audiozáznamy prepísali do textovej podoby, na ktorú sme aplikovali otvorené kódovanie [10]. Zameriavali sme sa na identifikovanie miskoncepcií a chýb, ktoré by nám pomohli určiť náročnosť jednotlivých úloh, resp. vhodnú primeranosť úloh pre skúmané ročníky. Táto primeranosť by mala zohľadňovať kognitívne schopnosti žiakov daného veku, požiadavky inovovaného ŠVP, motiváciu a motivujúcu náročnosť. Preto sme si pre náš výskum položili nasledujúce výskumné otázky: Aké sú najčastejšie chyby pri riešení grafových úloh? Aká je hranica medzi motivujúcou a demotivujúcou náročnosťou úloh? Aký je rozdiel pri riešení overovaných grafových úloh medzi žiakmi prímý a sekundý? Existujú rozdiely medzi riešeniami chlapcov a dievčat? Ak áno, aké sú a čím sú spôsobené?

4 VÝSLEDKY

4.1 Prvý pracovný list

Prvým pracovným listom, ktorý sme testovali so žiakmi, bol pracovný list *Rodostromy*. V úvode hodiny si žiaci spolu s učiteľom zopakovali základné rodinné vzťahy, ktoré zostali počas riešenia zapísané na tabuli, a aj to, ako vieme z nekompletných dát niečo usudzovať. Problémom bola, ako

sme aj predpokladali, orientácia šípok reprezentujúca vzťahy v grafe. Pre viacerých žiakov nebolo náročné čítať údaje z obrázka, ale skôr čítanie s porozumením a teda získavanie informácií z textu.

Ako najnáročnejšie sa ukázali úlohy 6 a 8. V úlohe 6 mali žiaci zakresliť rodostrom podľa troch viet, ktoré o rodine vedeli. Problematická bola najmä veta „Krištofova mama sa volá Patrícia.“, ktorú zle zakreslilo 6 žiakov, z toho 4 z prímy. Boli aj žiaci, ktorí tento vzťah priamo nezakreslili, lebo mali napríklad zakresleného Krištofovho otca a k nemu iba pridali šípku manžel k Patrícii. S celkovým smerovaním šípok mali problémy viac žiaci prímy, v sekunde sa vyskytovali takéto chyby minimálne.

V úlohe 8, v ktorej nebol rodostrom zakreslený, ale opísaný slovne, mali žiaci najväčšie problémy s určením vzťahu Lujzy a Emílie z vety: „Lujza je matkou Dana a dcérou Emílie.“. Šesť žiakov napísalo, že Lujza je Emíliina matka. Táto chyba sa rovnako často vyskytla v prime aj sekunde a rovnako tak medzi chlapcami aj dievčatami.

Všetci žiaci uviedli, že sa im ľahšie riešila úloha 7, teda úloha, kde bol grafická vizualizácia grafovej štruktúry. V úlohe osem si iba pár žiakov kreslilo, alebo čiastočne kreslilo rodostrom. Boli to najmä primáni. Väčšina žiakov si „v hlave predstavovala strom“ a „zistovala, čo bolo treba“, čiže si vytvárali nejakú mentálnu schému [11]. To sa aj napriek jednoduchosti rodinných vzťahov v úlohe ukázalo ako problematické.

4.2 Druhý pracovný list

Pri overovaní druhého pracovného listu – *Linky MHD*, začala hodina zobrazením použitej mapy na projektore. Žiaci mali hovoriť svoje nápady, čo to môže byť a čo znamenajú jednotlivé časti mapy. V jednej triede sa žiaci zhodli, že na mape sú električky (čo súvisí s označovaním električiek v Bratislave jednocifernými číslami), tmavé obdĺžniky sú významné zastávky a zvyšné iba málo významné zastávky. V druhej triede považovali bledé zastávky za zastávky na znamenie. Svoje názory vzťahovali na realitu, ale treba podotknúť, že bolo viacero žiakov, ktorí podobnú mapu nevideli a aj takých, čo MHD z rôznych dôvodov nevyužívajú.

Úlohy v tomto pracovnom liste považovali žiaci prímy aj sekundy za zaujímavé z viacerých dôvodov, ako napr.: že mohli hľadať rôzne riešenia, vytvárať vlastné trate, hľadať trasy a prestupy, či dopĺňať príbehy. Ako problematické sa ukázalo (hlavne pri žiakoch prímy), že v niektorých úlohách bolo viacero správnych odpovedí. Nepísali, alebo nehládali ich všetky, až pri rozhovore zisťovali, že existujú aj ďalšie možnosti. Napriek úvodu, niektorým žiakom nebolo jasné, čo znamenajú označenia na mape. Napríklad označenie konečnej zastávky, alebo že linka chodí z konečnej zastávky späť po tej istej trase. Keďže všetky tieto informácie boli potrebné pri riešení úloh a často aj vyplývali zo zadania, stávalo sa, že žiaci niektoré pojmy a pravidlá pochopili až počas riešenia úloh. Väčšinou sa ale nevrátili k predchádzajúcim úlohám a neopravili si ich.

Pri žiakoch sekundy bolo viac viditeľné prepojenie s reálnym svetom – napríklad namiesto električky odporučil žiak peší presun, lebo podľa mapy vyzerali dve zastávky ako veľmi blízke, či žiačka zdôvodnila výber prestupnej zastávky pre cestovanie učiteľky so žiakmi „lebo na tejto zastávke bude veľa ľudí a môžu sa tam stratiť“. Taktiež brali do úvahy aj to, že prestup je lepší na prvých zastávkach inej linky.

Viac ako polovica žiakov pri navrhovaní vlastnej linky z novovybudovaného sídliska navrhla jednu linku, ktorá obslúžila všetky zadané zastávky, pričom vo veľkej miere v nej využívali aj iné už existujúce zastávky. Toto bolo priamočiare riešenie, ale zaujímalo nás, či si žiaci uvedomia, že stačí spraviť linku spojenú s nejakou zastávkou, na ktorej sa dá prestúpiť na iné linky, ktoré prídu k zadaným zastávkam. Takýto prístup obsahovala iba necelá tretina všetkých riešení. V zvyšných riešeniach žiaci navrhli viac liniek, alebo linka nešla zo sídliska.

Žiacke ohodnotenie náročnosti úloh na päť bodovej škále bolo pre prímu v priemere na úrovni 3 (normálne) a pre sekundu na úrovni 1,94 (ľahšie). Z pohľadu dievčat boli úlohy normálne (2,9) a z pohľadu chlapcov skôr ľahšie (2,25). U chlapcov bol však väčší rozptyl medzi odpoveďami – pre niektorých boli úlohy veľmi ľahké, pre ďalších ťažšie. So žiackym hodnotením v niektorých

prípadoch nekorešpondovali ich výsledky – niektorí žiaci, ktorí úlohy hodnotili ako ľahké, mali v úlohách chyby, nerozumeli zadaniam, či neriešili presne to, čo mali.

4.3 Tretí pracovný list

Pri treťom overovaní s pracovným listom *Náramkový automat*, boli žiaci oslovení jeho farebnosťou. Najväčší problém žiakom spôsobovala slučka (znak znázorňujúci opakovanie daného koráliku), ktorej viacerí neporozumeli hneď, ale až po preriešení viacerých úloh v pracovnom liste. Vznikalo tu najviac miskoncepcií v rovnakej miere medzi primármi aj sekundármi.

Ako najproblematickejšia sa preto ukázala úloha 2d, v ktorej mali žiaci identifikovať a nakresliť útvar, ktorý sa zopakoval v náramku trikrát. Niektorí žiaci miesto neho dopĺňali tvar, ktorý v náramku nebol ale v automate áno, alebo nechali úlohu nevyriešenú. Väčšinou platilo, že žiaci, ktorí mali toto riešenie chybné, mali potom problémy aj s úlohou šest', v ktorej mali napísať, ktorý korálik sa bude opakovať v náramku a koľkokrát. Boli žiaci, ktorí počas riešenia prišli na to, ako slučka funguje, ale nevrátili sa a neskontrolovali si predchádzajúce odpovede. Viac žiakov s touto chybou bolo v sekunde a v oboch prípadoch to boli najmä chlapci.

Na druhej strane boli žiaci, ktorí prišli na to, že slučka sa môže opakovať viackrát (ostali však pri nízkej konštante, napr. trikrát). Ich odpoveď však nebola konzistentná, lebo v úlohe 6 odpovedali, že to môže byť maximálne dvakrát. Celkovo, najčastejšou odpoveďou bolo, že korálik na slučke sa môže zopakovať iba dvakrát – čo vydedukovali z ukážkového automatu na začiatku pracovného listu. U chlapca zo sekundy sa objavila aj možnosť, že sa môže opakovať „nekonečne veľa“, no musí to byť „minimálne dvakrát“. Zo zaujímavých prepojení na reálny život sa vyskytli odpovede: „toľko, koľko máme tohto typu korálikov“, či: „Ja som dala dvakrát... viac sa mi nechce vykresľovať a bol by veľký na detskú ruku.“

Aj v tomto pracovnom liste sa ukázalo, že žiaci nie sú zvyknutí na riešenia úloh, ktoré môžu mať viacero správnych odpovedí. Celkovo, jedna tretina všetkých žiakov (t. j. sedem) pri riešení úlohy 4b zistila, že náramok mohli vytvoriť oba automaty. Z nich boli dvaja chlapci (jeden z primy, jeden zo sekundy). Počas rozhovoru pri otázke, či tento náramok nemohol vytvoriť druhý automat, niektorí súhlasili a vysvetlili, že hľadali iba jednu odpoveď a teda, ak zistili, že jeden automat je správny, tak už na druhý nepozerali. Toto bolo časté najmä u chlapcov, dievčatá boli viac náchylné testovať oba automaty (čo potvrdzuje aj výsledok nášho predošlého výskumu, že dievčatá častejšie overujú všetky možnosti, ak ich je málo). Niektorí žiaci boli zmätení z viacerých možností a nevedeli, kde nastala chyba. Zaujímavé je, že pri úlohe 2c, kde tiež boli dve možné správne riešenia, iba jedna žiačka zo sekundy napísala do pracovného listu obe. Niektorí zo zvyšných žiakov druhú možnosť objavili až pri rozhovore a asi polovica si myslela, že tam má chybu.

Žiaci ohodnotili náročnosť pracovného listu v príme na 2,5 (medzi ľahkým a normálnym) a v sekunde na 2,8 (viac normálne). Rovnaké čísla vyšli pre dievčatá (2,5) a chlapcov (2,8). Zaujímavé je, že úlohu 8, v ktorej mali žiaci tvoriť vlastné automaty, spomenuli vo svojich odpovediach všetci žiaci. Páčila sa najväčšiemu počtu žiakov – 48 % (približne rovnako zastúpení chlapci aj dievčatá). Vo všetkých prípadoch ale vtedy žiaci úlohu 8 neoznačili ani ako najľahšiu, ani najťažšiu. Ako najľahšiu ju označilo 20 % žiakov a ako najťažšiu 33 % žiakov.

4.4 Celkové zhodnotenie pracovných listov

Pri prvom pracovnom liste, Rodostromy, sa nám potvrdili výsledky z predchádzajúceho výskumu [7] a to, že žiaci majú problémy so zakresľovaním údajov do štruktúr, ak nie sú vo vhodnej textovej forme. Toto môže byť spôsobené aj pravidlami v grafe, t. j. orientáciou šípok s rodinnými vzťahmi. Preto máme v pláne otestovať, či má orientácia šípok vplyv na riešenie úloh. Celkovo ale žiaci nemali v pracovných listoch veľa chýb. Zaujal najmä dievčatá, čo môžeme vidieť napríklad pri riešení úlohy, v ktorej mali zakresliť vlastnú rodinu – chlapci najčastejšie nakreslili iba minimálny potrebný počet ľudí a vzťahov, dievčatá zvykli kresliť celé rodiny.

Druhý pracovný list, Linky MHD, viac oslovil chlapcov. Zo všetkých troch pracovných listov tu bolo najviac prítomné hľadanie stratégie a optimálneho riešenia, v čom chlapci dosahujú lepšie výsledky na základe našich výskumov zo súťaže iBobor [3]. Napriek tomu dievčatá mali rovnako dobré výsledky ako chlapci. Rozdiely sme zaznamenali skôr pri žiakoch, ktorí nepoužívajú MHD. Pre nich boli niektoré úlohy ťažšie uchopiteľné, keďže im chýbalo prepojenie na reálny zážitok. Uvedomujeme si, že toto môže byť problémom v malých obciach a mestách, kde sa žiaci nestretávajú s rôznymi linkami či prestupovaním medzi nimi. Myslíme si ale, že vedieť čítať takéto diagramy, a vedieť plánovať cestu, je užitočné aj pre takéto deti, keďže sa v budúcnosti môžu s podobnými situáciami stretnúť.

Posledný pracovný list, Náramkový automat, zasahoval aj do oblasti Algoritmizácia a riešenie problémov zo ŠVP [6], či už z pohľadu analýzy problému alebo aj interpretácií riešenia (príkazov, ktoré boli zakreslené v automate). Tento pracovný list sa viac páčil dievčatám, ktoré oslovila najmä jeho farebnosť a väčšie množstvo obrázkov v porovnaní s predchádzajúcimi pracovnými listami. Problémy v pracovnom liste spôsobovala hlavne slučka a jej pochopenie, a to rovnako v prvej aj v druhej sekunde. Keďže sme nechceli ovplyvňovať výsledky, nezasahovali sme do žiackych riešení, ale pri použití pracovného listu v škole, odporúčame riešiť so žiakmi napríklad prvú úlohu skupinovo a diskutovať o fungovaní jednotlivých častí automatu. Uvažujeme aj nad úpravami prvých dvoch úloh, aby ukazovali širšiu škálu možností pre automat.

4.5 Záverečný dotazník

Zo záverečného dotazníka, ktorý vyplnilo 9 žiakov prvej (3 chlapci a 6 dievčat) a 9 žiakov druhej (6 chlapcov a 3 dievčatá), nám vyplynulo nasledovné. V prvej bol podľa žiakov najťažší pracovný list Linky MHD a ako najľahší hodnotili pracovný list Náramkový automat. V druhej ako najľahší pracovný list ohodnotili Linky MHD a pri najťažšom sa nezhodli. V druhej sa im najviac páčili Linky MHD a najmenej Náramkový automat. Tieto výsledky korelujú s pohlavím žiakov.

Keď sa pozrieme iba na odpovede chlapcov, zistíme, že pracovný list Linky MHD bol pre nich najľahší a zároveň sa im najviac páčil. Naopak, pracovný list Náramkový automat sa im nepáčil a ohodnotili ho ako najťažší. Z hodnotenia dievčat vyplývajú opačné výsledky. Pracovný list Náramkový automat bol pre nich najľahší a najviac sa im páčil. Pracovný list Linky MHD ohodnotili ako najťažší a páčil sa im najmenej. Dôvod, prečo pracovný list Rodostromy, bol v dotazníku zastúpený veľmi málo, môžu súvisieť aj s tým, že medzi testovaním tohto pracovného listu a dotazníka prešli tri týždne, čiže si ho žiaci nemuseli až tak dobre pamätať.

5 ZÁVER A DISKUSIA

Náš výskum sa venoval testovaniu troch pracovných listov so žiakmi prvej a druhej osemročného gymnázia. Všetky tri pracovné listy sú spojené s dátovými štruktúrami, konkrétne grafmi, a rozvíjajú základné operácie – orientácia v štruktúre, organizácia údajov do štruktúr a interpretácia údajov zo štruktúr, ktoré sú definované aj v inovovanom Štátnom vzdelávacom programe [6]. Pri riešení úloh na rôznych úrovniach kognitívnej náročnosti sa tak žiaci stretli s rôznymi pohľadmi na konkrétnu grafovú štruktúru a mohli si tak spraviť lepší mentálny model [11].

V našom výskume sme si kladli viaceré otázky ako napr.: „Aké sú najčastejšie chyby pri riešení grafových úloh?“ Na základe pozorovaného si myslíme, že je to hľadanie, resp. nehľadanie viacerých možných odpovedí, ktoré sú pri grafových štruktúrach časté. Taktiež žiaci často robia chyby v úlohách, v ktorých treba použiť viaceré typy reprezentácie údajov. Čiže v úlohách, v ktorých je potrebné prepájať informácie z textu s informáciami z obrázka. Aj naše výsledky ukazujú, že žiakom robí problémy čítanie s porozumením hoci aj stručného textu.

Ďalšia výskumná otázka, ktorú sme si kladli, bola: „Aká je hranica medzi motivujúcou a demotivujúcou náročnosťou úloh?“ Ako veľmi nápomocné vidíme prepojenie motivácie s reálnym životom, teda skúsenosťami, ktoré žiaci majú z bežného života. Takéto zručnosti vedú výrazne napomôcť k správne vyriešeniu úlohy. Napr. žiaci, ktorí cestujú bežne s MHD a riešili úlohy so

zástavkami a prestupovaním, dosahovali lepšie výsledky, ako žiaci, ktorí necestujú bežne s MHD. Motivácia, ktorá žiakov demotivuje je viac závislá od pohlavia. Farebné vizualizácie grafových štruktúr tak napr. viac pomohli k obľúbenosti úloh s náramkami u dievčat a tak úlohy pokladali za ľahké a riešili ju s väčším záujmom a nasadením. U chlapcov naopak táto úloha bola označená ako najnáročnejšia a zároveň sa im najmenej páčila. Takéto zistenia však súvisia aj s ďalšou výskumnou otázkou a to: „Existujú rozdiely medzi riešeniami chlapcov a dievčat? Ak áno, aké sú a čím sú spôsobené?“.

Čo sa týka otázky rozdielnosti medzi primánmi a sekundánmi, najviac boli viditeľné pri riešení pracovného listu Rodinné vzťahy, v ktorom mali primáni výrazne väčšie problémy. Sekundánom vo všeobecnosti stačilo menej času na riešenie pracovných listov, mali menej otázok k zadaniam a taktiež viac prepájali pracovné listy s reálnym svetom. Keďže ale boli obe skupiny pomerne malé a líšili sa v pomere chlapcov a dievčat, nevieme zhodnotiť ktoré problémy priamo súviseli s ročníkom a ktoré zas s pohlavím žiakov. Chystáme sa tieto rozdiely viac analyzovať v ďalšom výskume.

Aby sme lepšie porozumeli postupom, ako žiaci riešia úlohy, zvolili sme počas výskumu samostatnú prácu žiakov. Avšak na základe rozhovorov so žiakmi o ich riešeníach, pri ktorých prišli na to, čo jednotlivé veci znamenajú a taktiež boli schopní hľadať aj ďalšie riešenia a zdôvodňovať ich, si myslíme, že vhodnou formou, ako pracovné listy¹ používať v triede, je práca vo dvojiciach alebo skupinkách. Žiaci tak môžu diskutovať o riešení, vzájomne si vysvetľovať zadania, porovnávať svoje návrhy a objavovať rôzne spôsoby riešenia. Okrem informatických kompetencií tak môžu získavať ďalšie ako sú práca v skupine, vyjadrovanie svojich názorov a argumentácia, ktoré sú nemenej dôležité.

6 POĎAKOVANIE

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

7 BIBLIOGRAFICKÉ ODKAZY

- [1] iBobor. Dostupné na: www.ibobor.sk.
- [2] TOMCSÁNYIOVÁ, Monika, et al. Využitie úloh súťaže iBobor na hodinách informatiky. In: *Konferencia DidInfo 2012*. Banská Bystrica, Univerzita Mateja Bela. 2012.
- [3] BUDINSKÁ, Lucia, a Karolína MAYEROVÁ. Graph Tasks in Bebras Contest: What does it have to do with gender? In: *Proceedings of the 6th Computer Science Education Research Conference*, s. 83–90. ACM, 2017.
- [4] DAGIENE. Valentina, FUTSCHEK, Gerald a Gabriele STUPURIENE. Creativity in Solving Short Tasks for Learning Computational Thinking. In: *Constructivist Foundations* 14(3). s. 382–396. 2019.
- [5] BELLETTINI, Carlo, et al. Situated Learning with Bebras Tasklets. In: *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2019*. s. 225–239. Springer, Cham, 2019.
- [6] ŠPU: 2015, *Inovovaný štátny vzdelávací program: Informatika. Vzdelávacia oblasť: Matematika a práca s informáciami. Príloha ISCED 2*. Bratislava: Štátny Pedagogický Ústav. [vid. 2019-12-27]. Dostupné tiež na: http://www.statpedu.sk/sites/default/files/dokumenty/inovovany-statny-vzdelavaci-program/informatika_nsv_2014.pdf.
- [7] BUDINSKÁ, Lucia a Karolína MAYEROVÁ. Ako žiaci piateho a šiesteho ročníka základnej školy riešili vybrané grafové úlohy. In: *Medzinárodná konferencia o vyučovaní informatiky DidInfo 2019*, s. 35-40. Banská Bystrica, Univerzita Mateja Bela. 2019.

¹ dostupné na <http://edu.fmph.uniba.sk/~budinska/metodiky.php>

- [8] BUDINSKÁ, Lucia a Karolína MAYEROVÁ. From Bebras Tasks to Lesson Plans – Graph Data Structures. In *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2019*. s. 256–269. Springer, Cham, 2019.
- [9] ŠVAŘÍČEK, R. et al. *Kvalitativní výzkum v pedagogických vědách*. Praha: Portál, 2007. ISBN 978-80-7367-313-0.
- [10] CRESWELL, J. *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research*. 3 vyd. New Jersey: Pearson Education, 2008. ISBN 978-0-13-207208-0.
- [11] PAPERT, S. *The Eight Big Ideas of the Constructionist Learning Laboratory*. Unpublished internal document. South Portland, Maine, 1999.

Prezentovanie binárneho vyhľadávania žiakom základných škôl prostredníctvom interaktívnej aplikácie

The Presentation of the Binary Search Algorithm to Middle Schoolers Using the Interactive Application

Iveta Demková
UK FMFI KDMFI
Mlynská dolina
842 48 Bratislava
Slovensko
ivetkacs@gmail.com

Michal Anderle
UK FMFI KI
Mlynská dolina
842 48 Bratislava
Slovensko
anderle@dcs.fmph.uniba.sk

ABSTRAKT

Binárne vyhľadávanie je jedným z najintuitívnejších algoritmických konceptov. Naskytá sa preto otázka, do akej miery ho dokážu žiaci základných škôl samostatne objaviť a následne sformalizovať. Na základe úlohy v základníckej súťaži PRASK sme vytvorili metodiku jednej vyučovacej hodiny, ktorá prezentuje žiakom koncept binárneho vyhľadávania prostredníctvom interaktívnej aplikácie. Túto metodu sme následne otestovali na niekoľkých skupinách žiakov základných škôl navštevujúcich programátorské krúžky v Centre voľného času v Šali. V tomto príspevku prinášame popis použitej metodiky a interaktívneho prostredia, ako aj analýzu dát získaných testovaním v CVČ.

ABSTRACT

Binary search is one of the most intuitive algorithmic concepts. Therefore, the question of to what extent can primary school pupils discover it and then formalize it arises.

Based on the task in the primary school competition PRASK, we created a lesson plan, which presents the concept of binary searching through the interactive application. We then tested this lesson plan on several groups of elementary school pupils attending the programming clubs at the Leisure Activities Center (CVČ) in Šaľa. In this paper, we present a description of the used lesson plan and interactive environment, as well as the analysis of the data obtained from the testing in CVČ.

Kľúčová slova

Binárne vyhľadávanie, základná škola, súťaž PRASK, metodika

Keywords

Binary search, elementary school, PRASK competition, lesson plan

1 ÚVOD

Programovanie nie je a nemalo by byť jediným spôsobom rozvoja algoritmického myslenia žiakov. Ak totiž žiaci neporozumeli danému procesu ani na intuitívnej úrovni, je pre nich náročné ho zapísať vo formálnom programovacom jazyku [1]. Aj preto v posledných rokoch získavajú na popularite tzv. „unplugged aktivity“, ktoré predstavujú žiakom teoretické koncepty informatiky pomocou rôznych aktivít, ktoré nevyžadujú zapojenie počítača [2].

Ďalšou z možností, pri rozvoji informatického myslenia, je využitie algoritmických súťaží [3]. Práve tie majú na Slovensku a Česku pomerne dlhú tradíciu. Už viac ako 30 rokov existuje Olympiáda v informatike ale tiež Korešpondenčný seminár z programovania, či súťaž iBobor. A hoci sú tieto súťaže určené najmä pre talentovaných žiakov, ktorí majú záujem o informatiku, bolo by zaujímavé sa pozrieť na to, či by sa úlohy z takýchto súťaží nedali využiť aj v bežnej triede.

V tomto článku sa zameriame na algoritmickú súťaž PRASK, ktorá je určená pre žiakov druhého stupňa základnej školy. Cieľom tejto súťaže je rozvíjať algoritmické a programátorské schopnosti súťažiacich bez ohľadu na ich predchádzajúce skúsenosti. Počas roka sú zverejnené štyri sady piatich príkladov, na ktorých riešenie majú žiaci vždy aspoň mesiac času. Úlohy sa líšia, niektoré vyžadujú odovzdanie odladeného programu, iné slovný popis riešenia či prácu s interaktívnym prostredím. Hlavnou črtou týchto úloh je, že žiakom prezentujú pomerne komplexné algoritmické koncepty a dávajú im príležitosť k samostatnému skúmaniu pomocou postupne sa gradujúcich podúloh [4].

Úlohy z tejto súťaže boli použité aj v iných výskumoch [5, 6], dve z nich boli zaradené do vyučovania na strednej škole a získané výsledky sa porovnávali s výsledkami súťažiacich. Ukázalo sa, že prezentované koncepty boli žiakom pomerne prirodzené a vedeli sa v nich orientovať, problém im však spôsobovalo formálne vyjadrovanie.

V našom výskume sme sa rozhodli pozrieť na jeden zo základných algoritmických konceptov, binárne vyhľadávanie, a preskúmať, ako k nemu budú pristupovať aj mladší žiaci. V tomto príspevku predstavíme metodiku vyučovacej hodiny, ktorú sme navrhli na základe úlohy zo súťaže PRASK, a ktorá využívala interaktívne webové prostredie a uvidíme výsledky jej testovania.

2 TVORBA METODICKÉHO MATERIÁLU A INTERAKTÍVNEJ APLIKÁCIE

Binárne vyhľadávanie je jedným zo základných konceptov informatiky používaný na hľadanie špecifickej hodnoty v monotónnej postupnosti. Tento princíp však intuitívne využíva aj veľké množstvo ľudí v bežnom živote a to nielen pri hre „Hádaj na čo myslím!“. Vďaka tomu by mohol byť vhodným informatickým konceptom aj pre žiakov základnej školy.

Úloha o binárnom vyhľadávaní bola použitá v 2. kole 1. ročníka súťaže PRASK pod názvom „Prefíkaný kocúr“ [7]. V tejto úlohe mali riešitelia nájsť zadanú hodnotu v skrytej postupnosti čísel prostredníctvom otázok na konkrétne miesta postupnosti. K dispozícii mali interaktívne prostredie, ktorému mohli tieto otázky klást, a ktoré na ne odpovedalo. Úloha sa skladala z troch podúloh, ktoré sa líšili tvarom postupnosti čísel. V našom výskume nás však zaujíma iba prvá podúloha, v ktorej bola postupnosť 1000 rastúcich čísel.

Samozrejme, interaktívne prostredie z tejto úlohy sa nedalo priamo použiť na náš výskum. Napriek tomu sa nám zaradenie podobného interaktívneho prostredia zdalo vhodné a to z viacerých dôvodov. Chceli sme zachovať podobnosť s úlohou „Prefíkaný kocúr“, zvýšiť záujem a zapojenie žiakov a takisto zozbierať relevantné dáta na ďalšiu analýzu. Rozhodli sme sa preto implementovať vlastnú verziu tejto aplikácie.

Naša implementácia mala formu jednoduchšej webovej aplikácie s databázou¹ do ktorej sa prihlasovali anonymnými, vopred pripravenými, účtami. Ukážka sa nachádza na Obrázku 1. Táto aplikácia obsahovala tri rôzne podúlohy. V prvej mali žiaci nájsť zadané číslo v rastúcej postupnosti 100 čísel. Pýtať sa mohli na ľubovoľnú pozíciu a odpoveďou bolo, či sa hľadané číslo nachádza v postupnosti skôr alebo neskôr. V druhej obsahovala táto rastúca postupnosť 1000 čísel a odpoveďou na otázku bola hodnota čísla na danej pozícii. Tretia podúloha obsahovala postupnosť, ktorá najprv rástla až po hľadané číslo, a potom zase klesala. Binárne vyhľadávať bolo teda potrebné robiť na derivácii tejto postupnosti. Táto úloha však bola pridaná len ako aktivita pre najšikovnejších žiakov a ďalej sa ňou zaoberať nebudeme.

Pri každej podúlohe mohli žiaci zadávať svoje otázky, až kým nenašli hľadané číslo. Zároveň však mohli kedykoľvek začať odznova na novej postupnosti čísel. Aplikácia im počítala počet otázok, ktoré spravili a pridelať im body na základe toho, ako sa im darilo. Navyše, aby sme predišli tomu, že sa niekomu podarí uhádnuť správnu pozíciu na prvý pokus, postupnosť čísel sa generovala počas hádania a to tak, že každá odpoveď vylúčila čo najmenej možností tak, aby ostala konzistentná

¹ Interaktívne prostredie Stratený kocúrik: <http://kocurkovo.geo-inf.sk/> {prihlasovacie kontá na vyskúšanie: didinfo1 až didinfo10, heslo: didinfo2020} (webová stránka nemá HTTPS, tento problém môžete v prehliadači ignorovať)

s predchádzajúcimi odpoveďami. Vďaka tomu sme žiakom dopredu vedeli povedať, na koľko najmenej otázok môžu uhádnuť pozíciu hľadaného čísla.

Aby bola úloha pre žiakov prítiažlivá a prístupná, celé zadanie bolo prezentované v príbehu o „Stratenom kocúrikovi“, ktorý má na obojku číslo 10 000. Postupnosť čísel sa zmenila na postupnosť domov, v ktorých žili ostatní kocúry, a v ktorých sa mohli pýtať na číslo na obojku v danom dome, pričom vedeli, že čísla na obojku pozdĺž ulice iba rastú. Zaobalenie zadania úlohy do príbehu je dôležité, keďže vytvorenie vhodnej metafory môže ovplyvniť to, či žiaci úlohu porozumejú a dokonca, či ju správne vyriešia [8].



Obrázok 1: Ukážka prostredia

Pre potreby nášho výskumu sme pripravili metodiku jednej vyučovacej hodiny². K tejto aktivite bol potrebný počítač a účet v aplikácii pre každého žiaka. Priebeh hodiny bol nasledovný. Učiteľ predstavil žiakom aktivitu, ktorú budú na dnešnej hodine robiť a porozprával im príbeh zadania. Následne sa všetci žiaci usadili za počítače a začali samostatne riešiť vopred pripravené podúlohy, pričom mali začať najľahšou (100 domov) a pokračovať k ťažším. Webové rozhranie obsahovalo aj slovný popis jednotlivých úloh a takisto počet otázok, na ktoré je potrebné nájsť hľadané číslo, aby bola úloha akceptovaná ako vyriešená.

Žiaci napredovali podúlohami samostatne, pričom sa očakávalo, že iba zlomok z nich sa dostane k tretej podúlohe a väčšina sa podarí vyriešiť prvú a začať druhú podúlohu. Keď väčšina žiakov vyriešila prvú podúlohu, prebehla spolu s učiteľom diskusia, ktorej cieľom bolo upevniť a formalizovať novonadobudnuté vedomosti.

Uvedená metodika bola testovaná na informatických krúžkoch v Centre voľného času v Šali, ktoré navštevujú žiaci základných škôl. Toto prostredie nám poskytovalo priestor na experimentovanie, keďže zaradenie takejto aktivity do krúžku je jednoduchšie a takisto sme neboli až tak limitovaný časom a v prípade potreby sa aktivita mohla mierne predĺžiť. Vedúcou týchto krúžkov je autorka tohto článku, Iveta Demková, ktorá túto aktivitu aj viedla.

Spomenuté krúžky boli rôzneho zamerania – krúžok digitálnych technológií, robotický krúžok a krúžok programovania – a navštevovalo ich dokopy 32 žiakov z rôznych ročníkov základnej školy. Prehľad toho, koľko žiakov bolo v ktorom ročníku si môžete pozrieť v Tabuľke 1. Treba navyše podotknúť, že väčšina žiakov navštevovala uvedené krúžky aspoň druhý rok.

Tabuľka 1: Výskumná skupina

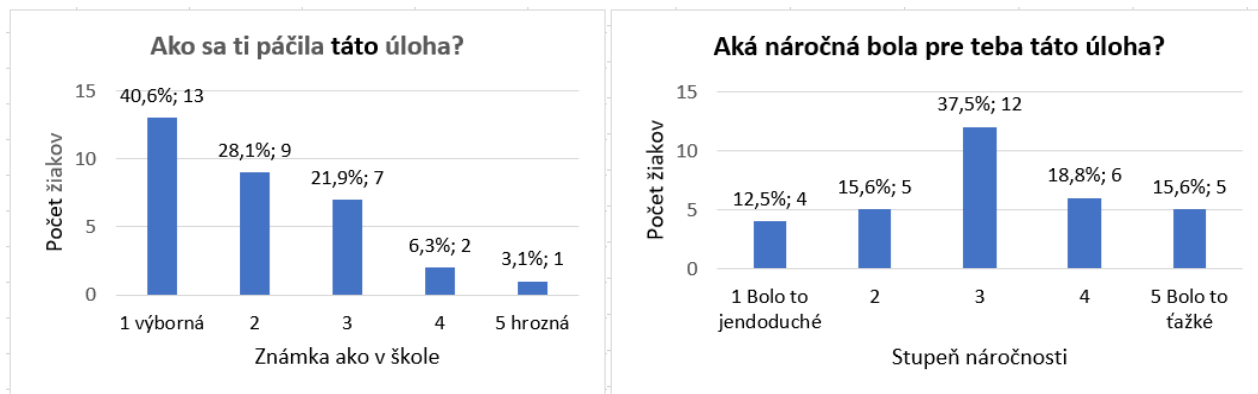
ročník	3.	4.	5.	6.	7.	8.	9.	spolu
chlapec	3	3	10	2	3	0	8	29
dievča	1	0	0	1	0	0	1	3

² Metodika: https://drive.google.com/file/d/1uDMI-7SebH0SqzeyhC5_Dz4phsVE0pbv/view?usp=sharing

3 ANALÝZA ZÍSKANÝCH DÁT

Žiaci po skončení aktivity vyplnili dotazník. Dotazník vyplnilo všetkých 32 skúmaných žiakov. Pýtali sme sa žiakov, ako sa im úloha páčila (Obrázok 2). Úloha sa veľmi páčila 40,6 % žiakov, ktorí ju označili ako výbornú. Iba 9,4 % žiakov sa úloha nepáčila (čísla na stupnici 4 a 5).

Zároveň nás zaujímalo, aká náročná bola pre žiaka táto úloha (Obrázok 3). Ako jednoduchú ju označilo 12,5 % žiakov, pre 15,6 % žiakov bola ťažká. Strednú hodnotu označilo 37,5 % žiakov.



Obrázok 2, Obrázok 3: Otázky z dotazníka

Zaujímalo nás, čo sa žiakom na úlohe páčilo. Deväť žiakov sa nevedelo vyjadriť. Ďalším ôsmim žiakom sa páčili gifové animácie s mačkami, ktoré sa žiakom ukazovali po zadaní otázky. Piatim žiakom sa páčilo využívanie logiky a premýšľanie nad riešením. Zvyšní žiaci uviedli, že sa im páčilo, že úloha bola ťažká, bolo to niečo nové a zaujímavé.

Chceli sme vedieť, aj to, čo sa žiakom nepáčilo a mali by sme do budúcnosti zmeniť. Osemnástim žiakom neprekážalo nič. Zvyšní žiaci navrhli, aby bola v zadaní ukážka alebo bolo viac detailov k úlohe. Jeden zo žiakov by privítal zníženie počtu domov na 50, ďalší by zase privítal, keby nemusel ručne zapisovať číslo a niekoľkí žiaci by uvítali, keby bola úloha ľahšia.

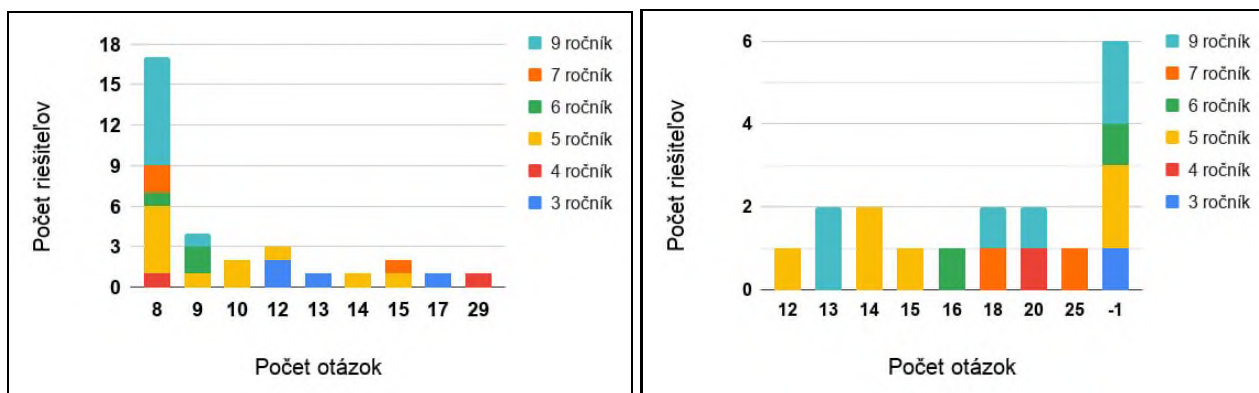
V budúcnosti by žiaci mohli riešiť viac podobných úloh, či už ako riešitelia súťaže PRASK, alebo prostredníctvom podobných prostredí. Pýtali sme sa preto žiakov, či by uvítali viac takýchto úloh. Až 52 % žiakov by uvítalo viac podobných úloh, 30 % žiakov je to jedno a iba 18 % žiakov nemá záujem o podobné úlohy.

Vďaka webovej aplikácii sme mali pomerne podrobné záznamy o tom, ako sa žiakom pri riešení darilo, koľko otázok potrebovali na riešenie, či ako rýchlo ich zadávali. Na tieto údaje sme sa teda bližšie pozreli. Pri prvej podúlohe so 100 domami sme zohľadňovali riešenia 32 žiakov, pri druhej podúlohe s 1000 domami iba riešenia 18 žiakov, ktorí sa dostali k riešeniu tejto podúlohy. Na Obrázku 4 môžete vidieť, prehľad počtu pokusov, ktoré žiaci použili v ich najlepšom pokuse, v ktorom sa im podarilo nájsť hľadané číslo. Podotýkame, že vzorové riešenie v prvej podúlohe vyžaduje 7 otázok, v druhej 10 otázok.

Prvú podúlohu (Obrázok 4) sa podarilo vyriešiť všetkým, aj keď nie optimálnym spôsobom. Napriek tomu väčšina výrazne zredukovala počet potrebných otázok, čím aspoň čiastočne uplatnili princíp binárneho vyhľadávania. Druhú podúlohu (Obrázok 5) sa podarilo vyriešiť iba 12 žiakom, zvyšní sa k nej buď nedostali, alebo nestihli nájsť hľadané číslo. Na obrázku 4 je uvedený počet otázok v najlepšom riešení prvej podúlohy (100 domov) pre každého žiaka. Na obrázku 5 je uvedený počet otázok v najlepšom riešení druhej podúlohy (1000 domov) pre každého žiaka. Hodnota – 1 označuje žiakov, ktorí podúlohu riešili, ale hľadané číslo nenašli.

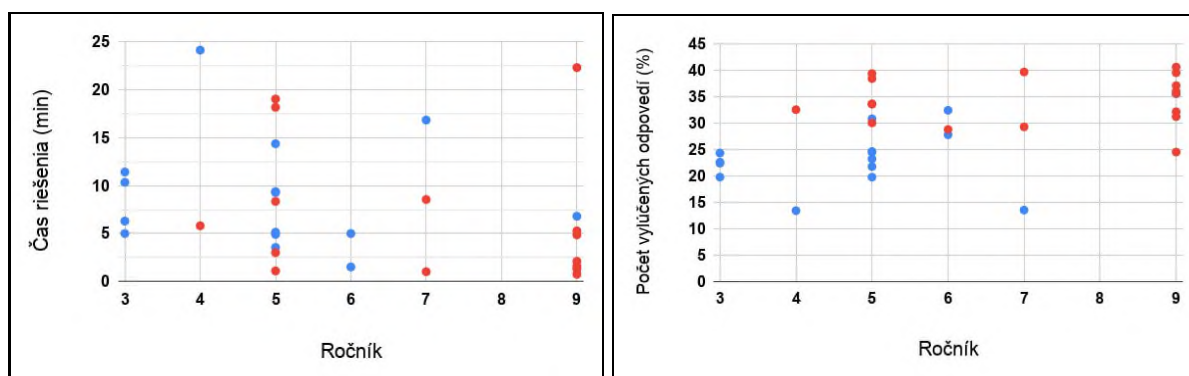
Treba si všimnúť, že hoci optimálne riešenie pre 100 domov je 7, najlepšie žiacke riešenie potrebovalo 8 otázok. Keďže číslo 100 nie je mocninou dvojky, žiak musí vznikajúce intervaly rozdeľovať veľmi pozorne, najmä ak majú nepárnu veľkosť. Ak sa čo len raz pomýli, algoritmus ho

prinúti položiť 8 otázok. Toto riešenie sme sa preto rozhodli pokladať za vzorové. Problémom navyše bolo, že zadanie špecifikovalo, koľko hádaní je potrebné spraviť na získanie „plného počtu bodov“. Tieto počty sme žiakom zverejnili aby sme ich motivovali a zároveň im ukázali, k akým hodnotám majú smerovať. Pomerne bežné teda bolo, že žiaci strávili dlhší čas vylepšovaním riešenia s 8 otázkami, namiesto toho, aby radšej riešili ďalšiu podúlohu.



Ako ďalšie sme sa pozreli na to, ako rýchlo sa žiaci dostali k ich najlepšiemu výsledku. V tomto prípade sme sa pozreli na čas od prvej otázky po poslednú otázku ich prvého najlepšieho riešenia. Na Obrázku 6 si môžete pozrieť tieto časy zoskupené podľa ročníka, ktorý žiaci navštevovali. Priemerný čas riešenia bol zhruba 9 minút, čo dáva odhad toho, koľko času je potrebné si vyhradiť pre takúto aktivitu.

Hľadali sme však aj hodnotu, podľa ktorej by sme vedeli posúdiť, nakoľko sa žiakom pri riešení darilo a ako dobre postupovali pri riešení. Princípom binárneho vyhľadávania je vylučovanie čo najväčšieho množstva zlých možností pomocou správne kladených otázok, pričom samozrejme optimálna otázka vylúči polovicu možných odpovedí. Pre každú žiacku otázku sme teda vedeli vypočítať, aké percento možných odpovedí daná otázka vylúčila. Následne sme sa pozreli na priemer týchto hodnôt pre všetky žiacke otázky. Tieto hodnoty sú uvedené v Obrázku 7 zoskupené podľa ročníkov. V priemere vylúčila jedna otázka zhruba 28 % odpovedí. Pomerne zaujímavé však je aj to, že 10 % otázok bolo zbytočných, teda nevylúčili žiadnu možnosť.

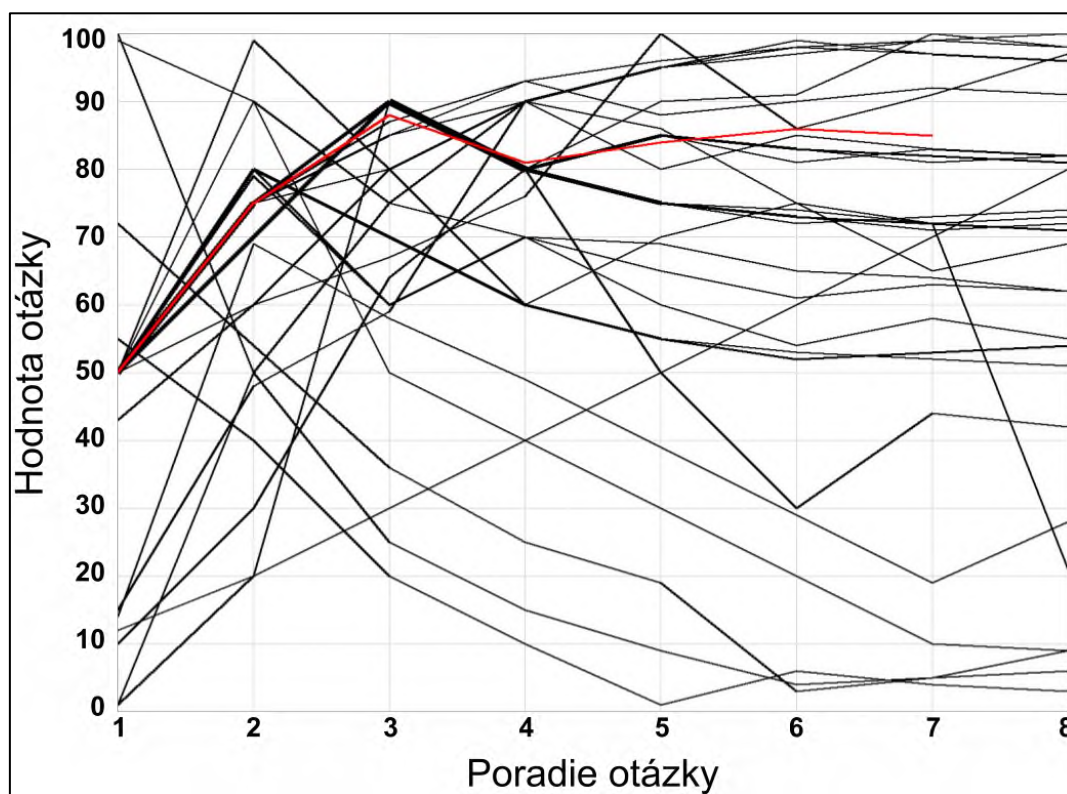


Už počas testovania sme si však všimli, že žiaci skúšali, čo im predložený program dovolí, preto časť týchto otázok pochádzala z takéhoto experimentovania. Aj pri pohľade do dát sa potvrdilo, že žiaci sa pýtali naozaj rôznorodé otázky.

Okrem tohto sme sa pozreli aj na druhú podúlohu, kde mali žiaci nájsť hľadané číslo v postupnosti 1000 domov. Najväčším problémom bolo, že žiaci nemali na riešenie dostatok času, častokrát to bolo iba pár minút. Dvanásť žiakov však našlo hľadané číslo, lepšie výsledky dosiahli žiaci, ktorí mali na riešenie viac času a mohli svoj postup opakovať.

Pomerne výrazný bol nárast otázok, ktoré nevyklúčovali žiadnu z možností, v priemere kládli žiaci 4krát viac takýchto otázok. Treba si však uvedomiť, že samotná úloha bola obtiažnejšia nielen väčšou dĺžkou postupnosti, ale aj odpoveďou, ktorú dostávali. Kým v prvej podúlohe sa priamo dozvedeli, či je hľadaná pozícia väčšia alebo menšia, v druhej podúlohe sa dozvedeli iba hodnotu čísla na tejto pozícii. Museli toto číslo teda porovnať s číslom hľadaným (čo bolo pre jednoduchosť vždy 10 000) a zároveň toto porovnanie správne vyhodnotiť. Táto pridaná zložitosť teda určite prispela k zvýšenému počtu zbytočných otázok. Navyše, z rozhovorov počas testovania vyplynulo, že žiaci očakávali v postupnosti čísel nejakú pravidelnosť a uvedené čísla boli pre nich máťúce. Časť žiakov dokonca prekážala veľkosť čísel, s ktorými museli pracovať.

Zaujímalo nás aj to, ako vyzerali samotné otázky, ktoré žiaci kládli, napríklad prečo potrebovali až 8 otázok. V prvej podúlohe sme teda zobrali najlepšie riešenie každého žiaka a všetky sme vložili do Obrázku 8. Na obrázku je znázornených prvých 8 otázok najlepších riešení každého žiaka a nadväznosť medzi týmito otázkami. Hrubšie čiary znázorňujú, že dané poradie otázok zvolilo viacero žiakov. Červená čiara reprezentuje vzorové riešenie. Vidíme, že takmer všetci žiaci začali správne otázkou na pozíciu 50, iba časť z nich však pokračovala na otázku 75 a nikto nezadal správnu možnosť 88 ako tretiu otázku. Aj z ostatných prechodov je zrejmé, že žiaci postupovali skôr odhadom ako po presných poloviciach, napríklad pomerne časté je zlepšovanie po 10.



Obrázok 8: Najlepšie riešenia žiakov z prvej podúlohy

Okrem tohto všetkého, sme do pôvodnej metodiky zaradili aj časť, v ktorej sme žiakov vyzvali, aby napísali na papier postup, ktorým úlohu riešili. Táto aktivita sa však stretla s nesúhlasom žiakov, ktorí po skončení úlohy nemali náladu alebo ochotu svoj postup aj spisovať. Túto časť teda nahradila diskusia s učiteľom. Iba piati z 32 žiakov dokázali opísať postup riešenia, ktorý reprezentuje binárne

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitná a pedagogická
vyhľadávanie. Väčšina žiakov buď iba spomenula prvý krok, kde zadali číslo 50, poprípade povedali, že tipovali.

4 DISKUSIA

Cieľom analýzy v predchádzajúcej časti bolo najmä získať základný pohľad o tom, ako žiaci zadanú úlohu riešili, koľko im to trvalo a ako úspešní boli. Nepozerali sme sa pri nej napríklad na to, ako vyzerali jednotlivé pokusy, čo viedlo žiakov k tomu, aby začali riešiť danú podúlohu odznova, poprípade sme sa nepozerali na to, či sa žiaci naprieč podúlohami zlepšujú.

Časť z toho bola spôsobená tým, že dát sme nemali dostatok, napríklad sme vopred nevedeli správne odhadnúť čas a druhú podúlohu riešila iba časť žiakov. Takisto to však nebolo hlavným cieľom našej práce, ale určite by sa v tom dalo pokračovať. Tiež poznamenávame, že vzorka žiakov bola pomerne malá, a preto sa zo získaných výsledkov nedajú odvádzať všeobecne platné závery.

Napriek tomu sa v riešeníach ukázali nejaké trendy, ktoré stoja za zmienku. Žiakom sme prezentovali problém, ktorého riešenie vyžadovalo zapojenie netriviálne konceptu binárneho vyhľadávania. Zdalo sa však, že aspoň na intuitívnej úrovni bolo žiakom jasné čo majú robiť a všetkým sa podarilo dopracovať k pomerne dobrému riešeniu. Dvoma najväčšími skupinami žiakov boli piatáci a deviataci. Vidíme, že deviatikom sa očakávateľne, darilo o niečo lepšie, naznačuje to rýchlosť riešenia aj úspešnosť ich otázok. Už piatáci si však vedeli s úlohou v rámci jednej hodiny poradiť.

Najväčšou výzvou pre žiakov však bolo zovšeobecnenie a formalizovanie ich riešenia. Ukázalo sa, že žiaci na niečo také nie sú zvyknutí a odmietajú to. Toto potvrdzujú aj dva iné výskumy s úlohami zo súťaže PRASK [5, 6], ktoré ukazujú, že formalizácia postupov je problematická aj na strednej škole. Nazdávame sa však, že niečo také sa od žiakov nezvykne vyžadovať, a preto v tom ani nie sú dobrí. V spomínaných článkoch bolo vidieť, že žiaci základnej aj strednej školy, ktorí sa stretávali s úlohami, ktoré od nich vyžadovali formalizáciu riešení, v tom boli výrazne lepší.

Inovovaný štátny vzdelávací program [9] hovorí, že táto schopnosť by mala byť na hodinách informatiky rozvíjaná. Výkonový štandard v časti „Algoritmické riešenie problémov“ udáva, že žiaci by mali byť schopní vyjadriť, krokovat' a realizovať návod, postup či algoritmus. Toto sa však nemusí týkať výlučne programov v programovacích jazykoch, teoretické úlohy podobné tej, ktorú sme prezentovali v tomto článku, predstavujú výbornú možnosť na to, aby si žiaci formálny zápis osvojili najprv vo svojom prirodzenom jazyku a až potom sa presunuli k programovacím jazykom.

5 ZÁVER

Výskum prezentovaný v tomto článku voľne nadväzuje na predchádzajúce výskumy, ktoré sa snažia skúmať, nakoľko sa dajú využiť teoretické úlohy zo súťaže PRASK vo výučbe informatiky. V tomto prípade sme sa pozreli na úlohu z tejto súťaže, ktorá sa týkala konceptu binárneho vyhľadávania, na ktorej základe sme vytvorili interaktívne webové prostredie a metodiku jednej vyučovacej hodiny. Počas novembra 2019 sme túto metodiku otestovali na žiakoch základných škôl navštevujúcich informatické krúžky v CVC Šaľa a získané dáta sme podrobili kvalitatívnej analýze.

Vo výsledkoch vidíme, že žiakom prišiel takýto typ úlohy zaujímavý a subjektívne ju považovali za primerane náročnú. Väčšine z nich, dokonca aj žiakom piateho ročníka základnej školy, sa podarilo v priebehu pár minút správne vyriešiť prvú podúlohu a postupom podobným binárnemu vyhľadávaniu našli v postupnosti dĺžky 100 hľadané číslo. Pri riešení síce postupovali správne, ale najmä intuitívne a mali veľký problém svoje riešenie zovšeobecniť a správne formalizovať.

A práve v tom vidíme najväčší prínos zaradenia podobných teoretických úloh do vyučovania. Pri správnom prevedení dokážu takéto úlohy žiakov zaujať, oprieť sa o ich intuíciu a popri tom im prezentovať zaujímavú a netriviálnu časť teoretickej informatiky a rozvíjať ich vyjadrovacie schopnosti, ktoré sú tak potrebné pre programovanie, ale aj naprieč ich celým štúdiom.

Samozrejme, tento výskum bol testovaný iba na malej a špecifickej vzorke, poskytuje však dobré východisko pre ďalšie práce. Bolo by zaujímavé overiť, či sa takáto hodina dá použiť aj na klasickej vyučovacej hodine a nakoľko dokážu aj mladší žiaci pochopiť všeobecný zápis riešenia.

6 BIBLIOGRAFICKÉ ODKAZY

- [1] LU, J., FLETCHER, G. "Thinking about computational thinking." ACM SIGCSE Bulletin 41, 2009, p. 260– 264.
- [2] CSUnplugged [online] [vid. 1.1.2019]. Computer Science Without a Computer, Dostupné na: www.csunplugged.org.
- [3] DAGIENE, V., FUTSCHEK, G. Introducing informatics concepts through a contest. In: IFIP Working Conference: New Developments in ICT and Education. Universite de Picardie Jules Verne, Amiens, 2010.
- [4] ANDERLE, M. PRASK – an Algorithmic Competition for Middle Schoolers in Slovakia, Olympiads in Informatics, 2018, Vol. 12, pp. 147–157.
- [5] ANDERLE, M. Transformation of tasks from Competition to High School Lessons – Binary Search Trees, ICERI, 2018 Proceedings, pp. 6549–6557.
- [6] ANDERLE, M. Porovnanie slovných riešení teoretickej úlohy žiakov strednej školy a riešiteľov súťaže PRASK. Zborník konferencie Didinfo, Banská Bystrica, 2019. ISBN 978-80-557-1533-9, ISSN 2454-051X. Dostupné na: https://io.fpv.umb.sk/didinfo/Zbornik_Didinfo_2019.pdf.
- [7] Prefíkaný kocúr [online] [vid. 1. 1. 2019]. Dostupné na: <https://prask.ksp.sk/ulohy/zadania/1013/>.
- [8] FORIŠEK, M. a STEINOVÁ, M. Explaining algorithms using metaphors. Springer, 2013
- [9] INOVOVANÝ ŠVP PRE 2.STUPEŇ ZŠ. [online] [vid. 1. 11. 2019]. Dostupné na: http://www.statpedu.sk/files/articles/dokumenty/inovovany-statny-vzdelavaci-program/informatika_nsv_2014.pdf.

Simulačné modely vo vyučovaní programovania

Simulation Models in Programming Teaching

Veronika Stoffová
Katedra matematiky a informatiky
Trnavská univerzita v Trnave
Priemyselná 4
918 45 Trnava
Slovenská republika
nikastoffova@seznam.cz

Ladislav Végh
Katedra matematiky a informatiky
Univerzita J. Selyeho
Bratislavská cesta 3322
945 01 Komárno
Slovenská republika
veghl@uj.sk

ABSTRAKT

Simulačné modely zohrávajú významnú úlohu vo vyučovaní. Podporujú aktívne a hĺbkové učenie sa hlavne v exaktných vedách, kde predmetom vyučovania sú matematickým modelom vyjadriteľné dynamické javy a procesy. Účinnosť a didaktický prínos simulačných modelov zabezpečuje algoritmom riadená vizualizácia výsledkov experimentov vykonaných na modeli. Na základe dobre naplánovaných simulačných experimentov žiaci, príp. študenti získavajú nové poznatky na základe vlastného bádania, pozorovania a objavovania.

ABSTRACT

Simulation models play an important role in teaching and learning. They support active and deep-learning, especially in the exact sciences, if the subject of teaching are dynamic phenomena or processes, which can be substituted with an exact mathematical model, function, or expression. The efficiency of simulation models is provided by algorithm-controlled visualization as results of the experiments. Based on well-planned simulation experiments, students gain new knowledge based on their own observation, research and discovery of facts.

Kľúčové slová

Programovanie, algoritmy, modelovanie, simulácia, simulačné experimenty, vizualizácia, animácia.

Keywords

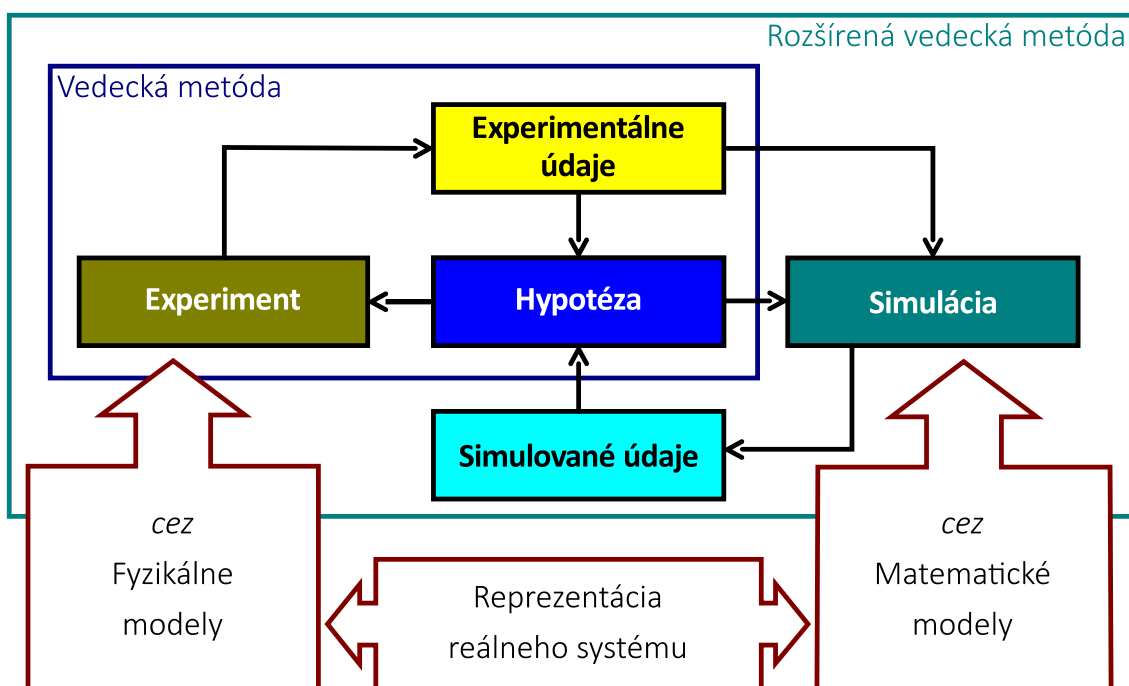
Programming, algorithms, modelling, simulation, simulation experiments, visualization, animation.

1 ÚVOD

Modelovanie a simulácia je známa, uznávaná, dobre rozpracovaná a často používaná výskumná metóda v inžinierskej praxi. Slúži na skúmanie systémov, na objavovanie ich vlastností a zvláštnosti správania sa bez ohľadu na to či bol systém definovaný alebo zavedený na skutočnom alebo na fiktívnom objekte. Simulačné modely sú najčastejšie implementované ako počítačové modely, ktoré umožňujú realizovať rôzne simulačné experimenty. Realizácia simulačných experimentov na reálnom objekte môže byť náročná, drahá, deštruktívna, môže priniesť fatálne výsledky, ba môže dokonca zapríčiniť katastrofu. Simulačným experimentom skúmame reakcie systému na rôzne podnety – na rôzne vstupné údaje a pri nastavení rôznych hodnôt parametrov samotného systému. Často práve optimálne nastavenie parametrov systému je cieľom modelovania. Toto nastavenie parametrov systému možno realizovať ešte pred výrobou prvého prototypu, či vyhotovením fyzikálneho modelu, makety a pod.

Najnáročnejšou fázou modelovania je identifikácia systému spolu s tvorbou matematického modelu, preverenie relevantnosti a validity systému a určenie podmienok a intervalu platnosti modelu. Cieľom modelovania je získanie nových poznatkov o modelovanom systéme a tým aj o objekte skúmania, na

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitní a pedagogická ktorom bol systém zavedený. Nasledujúci obrázok znázorňuje ako modelovanie rozširuje a obohacuje paletu vedecko-výskumných metód.



Obrázok 1: Simulácia ako výskumná metóda.

2 VYUČOVANIE POMOCOU VIZUALIZOVANÝCH SIMULAČNÝCH MODELOV

Vyučovanie pomocou vizualizovaných simulačných modelov je nielen zaujímavejší, zábavnejší a atraktívnejší spôsob získavania vedomostí, ale zvyšuje aj efektívnosť a kvalitu procesu učenia sa. Animačno-simulačné modely zohrávajú významnú úlohu vo vyučovaní takých tematických celkov, ktoré si vyžadujú porozumenie základných princípov fungovania študovaných dynamických javov a procesov, umožňujú pochopiť ich vlastnosti a kauzálne súvislosti medzi nimi. Simulačné experimenty môžu byť veľmi efektívnym nástrojom pri aktívnom učení sa s uplatnením konštruktivismu, s využívaním zvedavostou riadeného učenia sa a bádateľsky orientovaného vyučovania. Používanie simulačných modelov vo vyučovaní vyžaduje od učiteľa dôkladnú a intenzívnu prípravu na vyučovanie. V článku je ukázané ako didaktické simulačné modely vo vyučovaní využívať. Hlavná pozornosť je venovaná využívaniu interaktívnych vizualizovaných simulačných modelov vo vyučovaní algoritmov, algoritmizácie a programovania.

3 POHĽAD DO HISTÓRIE MODELOVANIA A SIMULÁCIE

Modelovanie a simulácia systémov je dobre známa a dôsledne rozpracovaná výskumná metóda v inžinierskych odboroch. Ako samostatná vedná disciplína má svoj predmet skúmania a metódy skúmania. Do študijných programov inžinierskych štúdií sa začala zavádzať a udomáčať koncom sedemdesiatych rokov minulého storočia. Najprv sa stala predmetom štúdia v študijnom odbore výpočtová technika na VUT v Brne. Podľa ich vzoru postupne sa táto disciplína vyučovala aj na ostatných technických univerzitách v bývalom Československu. Do študijného programu v odbore Technická kybernetika na Elektrotechnickej fakulte SVŠT (teraz STU) bola zavedená v akademickom roku 1975/76. Trojica mladých odborných asistentov dostala za úlohu nielen vypracovať študijné osnovy predmetu, ale aj napísať skriptá, aby sa mali študenti z čoho učiť a o čo sa oprieť v príprave na skúšku (Šafařík – Štofová – Cvik, 1981). Predmet mal silne aplikačný charakter a mal za úlohu spájať teóriu s výskumnou a výrobnou praxou. Skupina odborníkov, v ktorej

dominovali pedagogickí a výskumní pracovníci z technických univerzít, ČSAV a SAV postupne objavovala výskumný potenciál počítačových simulačných modelov. K tejto komunite odborníkov sa pridávali odborníci aj z oblasti ekonómie a ďalších vedných odborov. Založili konferenciu Modelovanie a simulácia systémov (MOSIS) ako pokračovateľa, prípadne ako sesterské podujatie seminárov MEDA. V Československom, neskôr medzinárodnom organizačnom výbore dominovali odborníci z elektrotechnických fakúlt VÚT v Brne, pražského ČVUT a z Ostravskej technickej univerzity. Slovensko reprezentovali odborníci z Elektrotechnickej fakulty SVŠT a tiež z košickej a žilinskej technickej univerzity. Táto konferencia, ktorá ešte stále funguje, bola a je dobrou platformou na výmenu skúseností, na prezentovanie nových trendov a výskumných metód nielen v oblasti simulácie systémov, ale aj v oblasti emulácie, využívania trenažérov, umelej inteligencie, neurónových sietí a celého radu moderných vedných disciplín. To dokazuje nielen interdisciplinárny charakter modelovania a simulácie, ale aj to, že táto vedná disciplína spája výsledky výskumu a vývoja z mnohých oblastí vedy. Veľká pozornosť bola venovaná pojmoslovnému systému a jeho štandardizácii. Vyčlenila sa silná skupina odborníkov hlavne z vysokých škôl nielen z bývalého Československa ale aj z okolitých štátov na čele s Ing. Milanom Kotvom. Táto odborná skupina sa pravidelne stretávala (minimálne v rámci konferencií MOSIS) a mala na starosti (zobrala si dobrovoľne na starosť) štandardizáciu obsahu jednotlivých pojmov, ktoré patria do základného terminologického systému modelovania a simulácie systémov. Táto skupina bola rozšírená o pozvaných odborníkov aj zo susedných štátov. Prísne sa diferencovali pojmy modelovanie a simulácia, modelovanie systémov a simulácia systémov, model a simulačný model, počítačový model a počítačový simulačný model), (Kotva, 1986), (Stoffová a kol., 2001). Štandardizoval sa postup tvorby simulačných modelov (Stoffová, 2004). Organizačný výbor konferencie zaviazal účastníkov konferencie, ako aj autorov konferenčných príspevkov, aby rešpektovali prijatý pojmoslovný systém a dodržiavali prijaté pravidlá. Nedodržanie týchto pravidiel, mohlo byť príčinou neprijatia príspevku na publikovanie, prípadne programový výbor vyzval autora, aby článok podľa prijatých zásad prepracoval alebo upravil (Kotva, 1986).

4 MODELOVANIE A SIMULÁCIA V UČITEĽSKEJ PRÍPRAVE NA SLOVENSKU

Modelovanie a simulácia (MaS) ako predmet výučby sa dostal do učiteľského študijného programu „Výpočtová technika“ (teraz „Informatika“) prvýkrát na samostatnej Pedagogickej fakulte v Nitre, keď hlavná autorka tohoto článku bola poverená dekanom fakulty vypracovať podkladové materiály na otvorenie nového učiteľského programu. Takýto učiteľský program v tom čase už bol otvorený na Univerzite Mateja Bela v Banskej Bystrici, tento predmet ale neobsahoval.

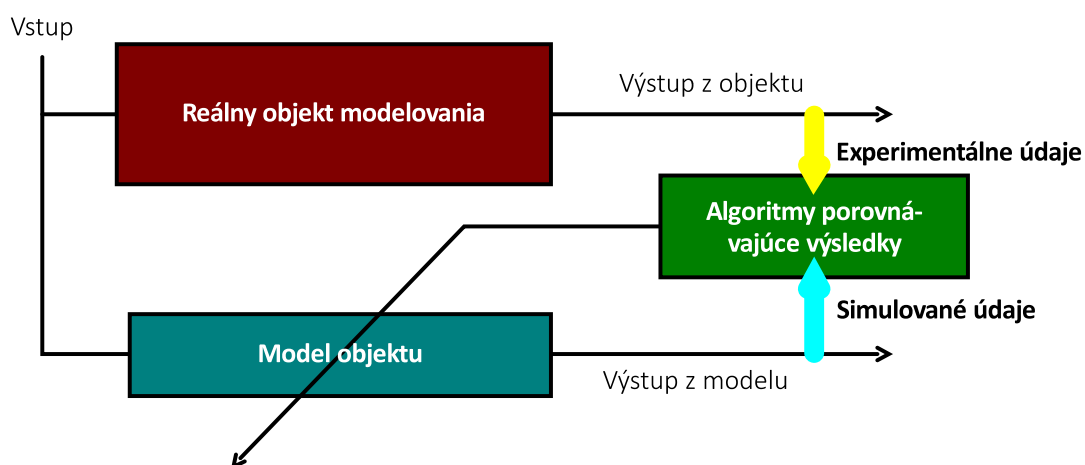
Už do prvej verzie študijného programu učiteľstva informatiky na Pedagogickej fakulte v Nitre sa dostal aj predmet Modelovanie a simulácia. Dôvodov na zaradenie tohto predmetu do učiteľského programu bolo niekoľko. Hlavným dôvodom bola, adaptácia tohto predmetu z inžinierskeho študijného programu do učiteľského a objavenie didaktického potenciálu počítačového modelovania a simulácie vo vzdelávaní na získavanie nových poznatkov (Stoffová, 1979). Motiváciou boli aj výsledky výskumu v oblasti tvorby a počítačovej implementácie edukačných expertných systémov a tvorby simulačných modelov objektov vyučovania na vzdelávacie účely. Inšpiráciou bola aj v tom čase čerstvá publikácia „Modelování a modely ve vyučování fyzice“ (Lepil – Vachek, 1986). Predmet MaS bol zaradený do 4. ročníka štúdia, kde v zimnom semestri sa vyučovalo modelovanie diskretných systémov a v letnom semestri modelovanie spojitých systémov. Predmet v porovnaní s rovnomenným predmetom inžinierskych študijných programov sa viac orientoval na optimálne riadenie vyučovacieho/vzdelávacieho procesu a tiež na objekty a javy jednotlivých predmetov vyučovania s cieľom spojenia vzdelávania s praxou. Pri vysvetľovaní a aplikovaní jednotlivých metód MaS vždy sa našla vhodná aplikácia do vzdelávania, príp. v oblasti vzdelávania. V rámci predmetu študenti vypracovávali semestrálny projekt, v ktorom implementovali vizualizované simulačné modely na didaktické účely. Témy, z ktorých si mohli vybrať boli orientované na predmety

informatiky ako aj na predmety druhého aprobačného predmetu. Druhým aprobačným predmetom prvej skupiny študentov učiteľstva informatiky bola matematika. Veľmi skoro ako druhý aprobačný predmet pribudla fyzika, neskôr postupne všetky ostatné aprobačné predmety Fakulty prírodných vied UKF v Nitre. Po niekoľkých rokoch existencie učiteľstva informatiky na UKF v Nitre vznikli aj medzifakultné dvojpredmetové aprobácie s Filozofickou fakultou a Pedagogickou fakultou UKF. Populárna bola dvojpredmetová aprobácia: informatika a anglický jazyk (medzi FPV a FF) a informatika a základy techniky (medzi FPV a PF). Ojedinele vznikali aj málopočetné skupiny s tzv. výchovami, napr. informatika s výtvarnou, hudobnou a telesnou výchovou. Práve táto rozmanitosť dvojpredmetových kombinácií umožnila rozšíriť aplikačné pole pre metódy modelovania a simulácie.

Aj v inžinierskych (tak technických ako aj ekonomických) a prírodovedných odboroch sa popri všeobecnej metóde modelovania a simulácie systémov začali objavovať špecializované predmety orientované na aplikovanie modelovania a simulácie vo vedných odboroch ako sú fyzika, chémia, biológia, ekológia, ekonómia, astronómia, meteorológia, architektúra, strojárstvo, stavebníctvo, výrobné procesy, robotika, bionika, biokybernetika, lekárske vedy pod. a tiež v ich pododboroch. Vznikali zaujímavé aplikačné systémy pre užšie, špecifické a problémové oblasti a konkrétne tematické celky. Genetika a pravidlá genetiky, modelovanie molekúl a chemických reakcií, modelovanie v meteorológii, modelovanie senzometrických javov, modelovanie 2D a 3D útvarov, modelovanie technologických procesov, CAD, CAM systémy a pod.

Vznikajú komplexné softvérové nástroje, ktoré umožňujú interaktívny návrh a analýzu systémov na základe ich modelovania. Takýmto popredným vývojárskym prostredím na interaktívnu tvorbu matematického počítačového softvéru je MathWorks. Obsahuje niekoľko relatívne samostatných častí, ktoré sú určené pre inžinierov a vedcov a významne uľahčujú ich prácu. MatLab je navrhnutý pre inžiniersky spôsob myslenia a na zefektívnenie práce vývojára. Či už ide o analýzu údajov, vývoj algoritmov alebo vytváranie modelov systémov. Na návrh systémov a vykonanie simulačných experimentov slúži softvér Simulink. Simulačný model umožňuje preskúmanie systému a implementáciu jeho návrhu. Zohľadniť požiadavky na systém a nájsť jeho optimálne parametre na základe matematického modelu ešte pred vybudovaním reálneho systému, by sa dalo určiť bez napísania programu (napr. C, C++, HDL a pod.) len veľmi ťažko. Práve tento proces zjednodušuje Simulink a podobné systémy, pomocou ktorých implementovať simulačný model pre odborníka nie je zvláštnym problémom.

Vymenovať odbory, oblasti vedy a výskumu a konkrétne problémy, kde sa modelovanie a simulácia systémov môže uplatniť nielen ako výskumná metóda ale aj ako optimalizačná metóda nie je možné pre obmedzený rozsah článku. Možno ale všeobecne konštatovať, že využívanie modelovania a simulácie je veľmi široké a oblasti aplikácií sú neobmedzené. V súčasnosti je ťažko nájsť vednú oblasť, kde by sa nemohli metódy modelovania a simulácie uplatniť.



Obrázok 2: Všeobecná schéma tvorby modelu objektu (z archívu autorov).

Najnáročnejšou fázou modelovania je identifikácia systému spolu s tvorbou matematického modelu, preverenie relevantnosti a validity modelu a určenie podmienok a intervalu jeho platnosti. Cieľom modelovania je získanie nových poznatkov o modelovanom systéme a tým aj o objekte skúmania na ktorom bol systém zavedený.

Na obrázku 2 je schematicky znázornený proces preverovania validity a relevantnosti modelu na základe porovnania výstupov z modelu a z reálneho objektu na rovnaký vstupný signál (na rovnaké vstupné údaje). Na základe rozdielu výstupných hodnôt sa vykonávajú zmeny v modeli, tak v matematickom ako aj implementačnom. Zároveň sa určujú podmienky platnosti a interval platnosti modelu pre možnosť náhrady reálneho objektu jeho modelom na rôzne účely.

5 MODELOVANIE A SIMULÁCIA VO VZDELÁVANÍ

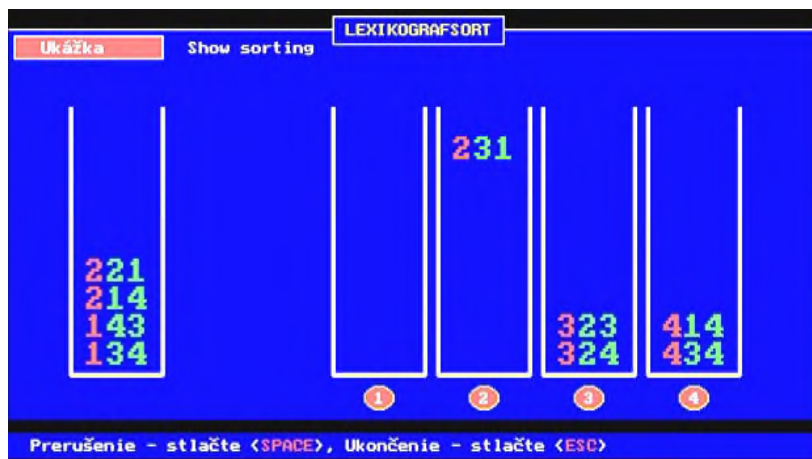
Dôvodom a didaktickým cieľom využívania simulačných modelov v edukácii je získanie nových poznatkov o skúmanom objekte (o predmete vyučovania). V tomto prípade získané poznatky sú len relatívne nové (aj keď sa nevylučuje ani získanie absolútne nových poznatkov o predmete poznávania). Prvé animačno-simulačné modely algoritmov boli naprogramované v rámci študentskej vedeckej a odbornej činnosti, ktoré implementovali študenti z prvej skupiny učiteľstva akademických predmetov Výpočtová technika a Matematika v prostredí programovacieho jazyka Pascal s použitím grafickej knižnice Unit Graph. Katedra matematiky, na ktorej bol zahájený učiteľský študijný program výpočtová technika, bola v tom čase vybavená 8-bitovými počítačmi PMD 85, kde ako zobrazovacie zariadenie sa používal obyčajný prenosný televízny prijímač. V tom čase na katedre bol jediný počítač značky Novotec s farebnou obrazovkou. Autor súťažnej práce ŠVOUČ s názvom Triedenia a triediace algoritmy, Erik Urland, študent prvého ročníka aprobácie matematika a informatika, ktorý sa v sekcii Matematika s didaktikou v rámci fakultného kola umiestnil na 3. mieste, získal právo zúčastniť sa celoštátneho kola. V celoštátnom kole súťaže o najlepšiu študentskú vedeckú prácu v odbore: Výpočtová technika vo vyučovaní predmetov prezentovaná súťažná práca získala 1. miesto. Celoštátne kolo ŠVOUČ v šk. r. 1998/99 sa konalo v Prahe. Tento úspech nás nielen potešil, ale aj povzbudil v našej ďalšej tvorivej výskumnej činnosti.

6 DIDAKTICKÉ VIZUALIZOVANÉ SIMULAČNÉ MODELY TRIEDIACICH ALGORITMOV

V ďalšom predstavíme dve aplikácie na podporu vyučovania algoritmizácie a programovania. Obe aplikácie sú orientované na triediace algoritmy. Už prvé verzie týchto vizualizovaných simulačných modelov mali veľkú didaktickú hodnotu. Náborne demonštrovali ako jednotlivé triediace algoritmy fungujú a aké majú vlastnosti. Aj použité farby skrývali určité informácie, ktoré vysvetlíme pri opise jednotlivých modelov triediacich algoritmov. V prvej sérii triediacich algoritmov boli ako samostatné aplikácie implementované: jednoduché triedenie, triedenie výberom (MinSort a MaxSort), bublinkové triedenie, ShellSort, Insertsort (vo verzii lineárne aj binárne hľadanie hľadanie miesta pre testovaný prvok), MergeSort, triedenie výpočtom adresy, lexikografické triedenie a QuickSort (v dvoch grafických vyhotoveniach: ako triedenie paličiek a ako triedenie bodov v rovine podľa horizontálnej súradnice). Tieto aplikácie boli vytvorené na pochopenie ako jednotlivé triediace algoritmy fungujú. Používateľ na základe vlastného pozorovania priebehu simulačných experimentov mal pochopiť základné princípy jednotlivých triediacich algoritmov (Stoffová, 1991).

Na obrázku 3 je odpamätaná obrazovka z poslednej fázy lexikografického triedenia čísel. Triedené čísla sú trojčiferné, ktoré boli vygenerované náhodne a uložené do zásobníka na ľavej strane obrazovky. Na triedenie (na konečné usporiadanie) čísel v tejto implementácii sa používajú pomocné zásobníky. Počet pomocných zásobníkov je daný počtom rôznych čísiel, z ktorých sú vytvorené čísla. Teoreticky pre dekadickú číselnú sústavu by to bolo 10 zásobníkov. Pre väčšiu prehľadnosť na tvorbu a triedenie trojčiferných čísel boli použité len 4 číselné znaky, a to 1, 2, 3 a 4. Teda pri triedení stačí mať k dispozícii 4 pomocné zásobníky. Zásobníky fungujú ako štandardizovaná údajová štruktúra LIFO. Teda čísla sa ukladajú do zásobníka jedno na druhé a vyberajú sa v opačnom poradí.

Stručný opis lexikografického triedenia: Vygenerované náhodné čísla sa uložia jedno po druhom do hlavného zásobníka. Z hlavného zásobníka sa postupne vyberajú čísla a ukladajú sa do pomocných zásobníkov najprv podľa číslice v najnižšom ráde. (Teda sa vytvárajú skupiny čísel v pomocných zásobníkoch podľa číslice v najnižšom ráde.) Po vyprázdnení hlavného zásobníka sa čísla postupne vracajú do neho z pomocných zásobníkov tak, že najprv sa vyprázdni zásobník označený najnižším číslom. Potom sa postupne vyprázdňujú zásobníky označené najbližším vyšším číslom (podľa pravidiel fungovania zásobníka), kým pomocné zásobníky sa nevyprázdnia úplne (teda nevyprázdni sa pomocný zásobník s najvyšším číslom). Tento postup opakujeme pre rád desiatok, stoviek ... kým nedostaneme usporiadanú postupnosť čísel v hlavnom zásobníku.



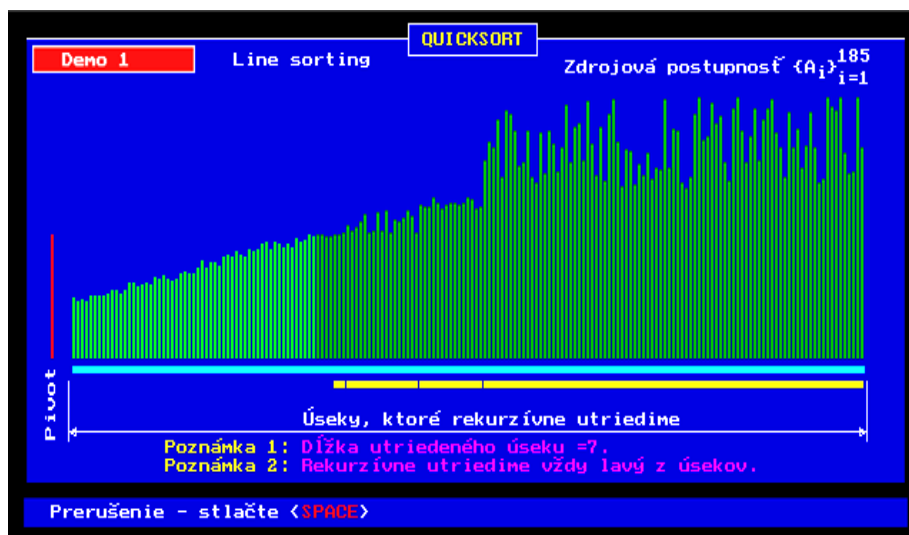
Obrázok 3: Animačno-simulačný model lexikografického triediaceho algoritmu.

Význam farieb a informácií na obrazovke: Na obrazovke hore nájdeme názov triediaceho algoritmu. Na začiatku triedenia všetky čísla v hlavnom zásobníku majú žltú farbu, čo znamená, že sú v náhodnom poradí – teda neusporiadané. Keď sa dokončí triedenie podľa príslušného rádu, farba číslice v danom ráde (v hlavnom zásobníku) sa zmení na zelenú. Tlačidlo „SPACE“ slúži na zastavenie/prerušenie vykonávania algoritmu a animácie. Po stlačení tlačidla „ESC“ vykonávanie algoritmu ako aj animácia sa ukončia.

Na obrázku 3 je zachytená posledná fáza lexikografického triedenia, keď do hlavného zásobníka sa vracajú čísla z pomocných zásobníkov, kde boli uložené podľa číslice v ráde stoviek.

Ďalšie aplikácie, ktoré chceme ukázať, demonštrujú triediaci algoritmus QuickSort. Tento algoritmus sme si zvolili preto, lebo bol implementovaný v dvoch verziách. Aplikácia pod menom QuickSort I (Line sorting) demonštruje ako funguje QuickSort na triedenie stĺpikov náhodnej výšky a náhodného usporiadania, a QuickSort II (Pixel sorting) demonštruje tento triediaci algoritmus na triedenie bodov v rovine podľa vertikálnej súradnice.

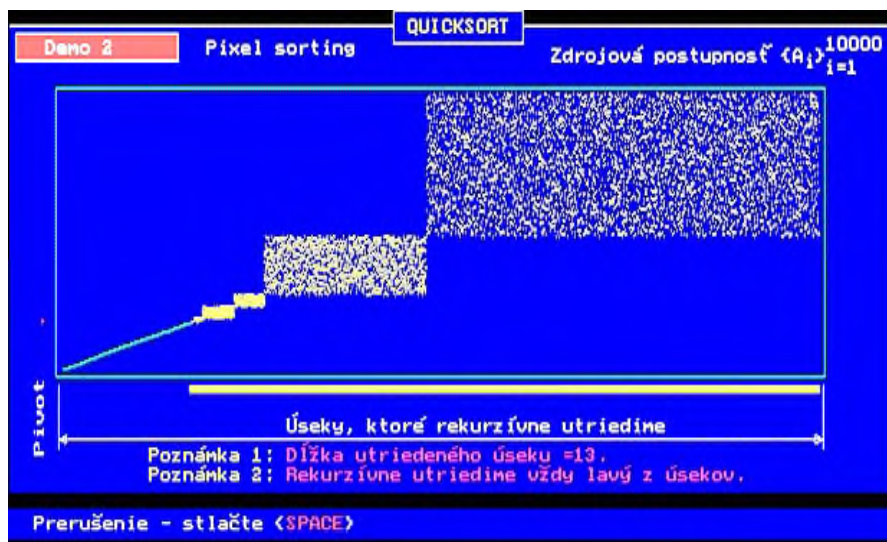
Stručný opis triediaceho algoritmu QuickSort I: Náhodne vygenerovaná sekvencia (185) náhodných čísel reprezentovaných pomocou bledozelených horizontálnych stĺpikov sa rozdelí na dve časti podľa jedného stĺpika tzv. pivota tak, že všetky od neho nižšie stĺpiky budú pred pivotom a všetky väčšie za ním. Tento stav usporiadania sa dosiahne tak, že vzájomne sa vymieňajú od pivota väčšie prvky pred, za prvky menšie od pivota, ktoré sa nachádzajú za pivotom. Prvky pred pivotom sa testujú od začiatku smerom k pozícii pivota a prvky za pivotom kontrolujú od konca smerom k pozícii pivota, kým sa nenájde vhodný prvok na výmenu v jednotlivých úsekoch. V tomto postupe sa pokračuje, kým nebudú všetky prvky menšie od pivota pred pivotom a všetky väčšie za pivotom. Tento postup sa opakuje na menšie (polovičné) úseky kým úseky sa nezredukujú na jedno-, príp. dvojprvkové. Prvky v jednoprvkových úsekoch (medzi pivotmi) sú už na správnom mieste. Prvky v dvojprvkových úsekoch sa porovnávajú a ak nie sú v správnom poradí tak sa vzájomne vymenia.



Obrázok 4: Animačno-simulačný model usporiadania stĺpikov podľa triediaceho algoritmu QuickSort

Význam farieb a informácií na obrazovke: Na obrazovke hore nájdeme názov triediaceho algoritmu. Náhodné čísla, ktoré sa majú usporiadať, sú prezentované zelenými stĺpikmi. Pivot (porovnávací prvok) je vyznačený červenou farbou na ľavej strane okna. Ostatné doplňujúce informácie sa nachádzajú vo forme poznámok. Tlačidlo „SPACE“ slúži na zastavenie vykonávania algoritmu, aby učiteľ mohol doplniť opis toho, čo sa deje na obrazovke. Prípadne, aby vysvetlil jednotlivé kroky algoritmu a situáciu, ktorá je znázornená na obrazovke.

Na pochopenie, ako QuickSort funguje, animácia algoritmu pomocou paličiek úplne postačí. Pre objavenie jeho vlastností má vyššiu názornosť reprezentácia hodnôt pomocou bodov v rovine (2D), kde triedenie bodov prebieha na základe hodnoty vertikálnej súradnice bodu. Po usporiadaní bodov vznikne bledozelená čiara.



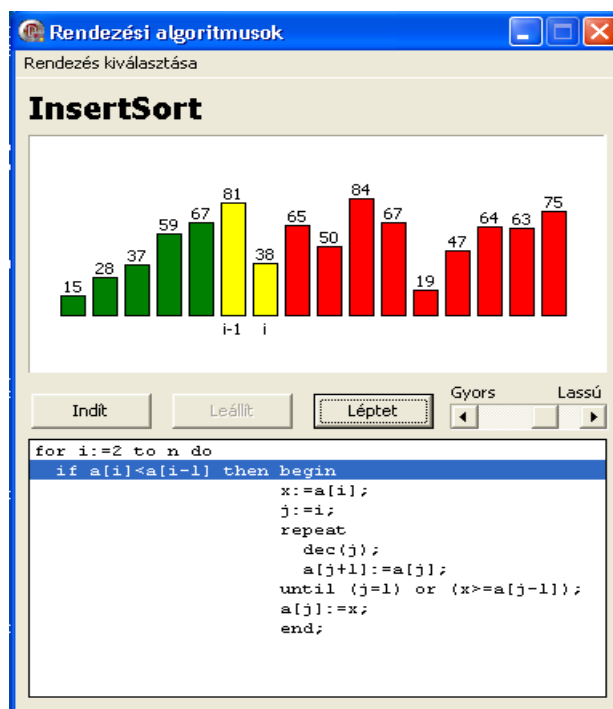
Obrázok 5: Animačno-simulačný model usporiadania bodov podľa triediaceho algoritmu QuickSort

Stručný opis triediaceho algoritmu QuickSort II: Náhodne vygenerovaná množina bodov vo zvolenom počte (10 000) náhodných čísel sa zobrazí podľa x-ovej a y-ovej súradnice na ploche obdĺžnika. Súradnice i -tého bodu x_i, y_i sú náhodné čísla, $x_i \in (x_{min}, x_{max})$ a $y_i \in (y_{min}, y_{max})$ reprezentovaných žltým bodom na ploche obdĺžnika s rozmermi $(x_{max} - x_{min}) \times (y_{max} - y_{min})$ sa rozdelí na dve časti podľa x-ovej súradnice bodov porovnaním x-ovej súradnice tzv. pivota tak, že všetky body, ktoré majú y-ovú súradnicu menšiu ako je y-ová súradnica pivota sa vzájomnou výmenou

(preusporiadaním) umiestnia na začiatok sekvencie bodov. Tak sa rozdelí plocha pôvodného obdĺžnika na 4 obdĺžniky, ktoré sú ohraničené obvodovými čiarami pôvodného obdĺžnika a vodorovnou a zvislou čiarou, ktorá prechádza bodom reprezentujúcim pívota. Tento rekurzívny postup sa opakuje dovtedy, kým sa neusporiadajú všetky body podľa y-ovej súradnice do rastúcej postupnosti.

Význam farieb a informácií na obrazovke: Význam farieb a informácií na obrazovke je rovnaký ako v predchádzajúcom prípade. Jasne vidieť, že vlastnosti QuickSortu sú zvýraznené lepšie (teda sú čitateľnejšie) pri triedení bodov ako pri prezentovaní náhodných čísel pomocou výšok stĺpikov.

Postupným vylepšovaním prvej série animačno-simulačných modelov algoritmov sme chceli zvýšiť ich didaktickú hodnotu, demonštračnú silu, názornosť, interaktivitu, podporu aktívneho zapojenia sa edukanta do vzdelávacieho procesu a získavanie hlbokých poznatkov o tom ako algoritmy fungujú, aká je ich zložitosť a kde sa dajú efektívne využívať. Po nástupe nadstavbového objektovo orientovaného programovacieho prostredia Delphi bol implementovaný ďalší súbor animačno-simulačných modelov s využitím našich predchádzajúcich skúseností z ich vývoja a uplatnenia vo vzdelávacom procese. Vizualizácia simulačných experimentov bola rozšírená o okienko s programom v metajazyku, ktoré bolo prepojené algoritmom riadenou animáciou tak, že sa vždy zvýrazňovala tá časť kódu, ktorá sa práve vykonávala. Tiež bolo možné z animácie rozpoznať, v ktorej fáze (v ktorom kroku) realizovaného algoritmu sa nachádzame. Používala sa nielen jednotná koncepcia pri implementácii a grafickom znázornení, ale jednotne (v rovnakom význame) sa používali aj farby a navigačné a ovládacie prvky. Opísané skutočnosti znázorňuje obrázok 6, ktorý ukazuje stav v určitej fáze triedenia vkladáním.



Obrázok 6: Ukážka animačno-simulačného modelu triediaceho algoritmu InsertSort

Stručný opis obrázku 6: V strede obrazovky je názov triediaceho algoritmu. Okno pod názvom názorne zobrazuje stav triedenia. Zelenú farbu majú stĺpiky, ktoré sú už usporiadané. Žltú farbu majú stĺpiky, ktoré sa porovnávajú a červenou farbou sú znázornené stĺpiky, ktoré nie sú ešte usporiadané. Pod stĺpikmi vidieť nastavenie hodnoty riadiacej premennej cyklu i . Označenie riadiacich tlačidiel zľava doprava. Štart, Zastaviť, Krokovat' a nastavenie rýchlosti animácie (Rýchlo – Pomaly). V dolnom okienku vidíme celý kód/program, ktorý vyjadruje algoritmus triedenia vkladáním. Riadkový kurzor modrej farby ukazuje ktorý príkaz sa práve realizuje – teda umožňuje ľahšie a jednoduchšie čítanie algoritmu.

Animačno-simulačné modely (ASM) algoritmov a tiež na podporu vyučovania programovania sa systematicky vyvíjali a testovali a postupne sa objavovalo ich didaktický potenciál a motivačná sila. Na základe výsledkov pedagogického výskumu a našich skúseností z využívania animačno-simulačných modelov sa vypracovali nielen zásady ich tvorby, ale aj ich využívania na vzdelávacie účely a na hĺbkové učenie sa, na tzv. „**deep learning**“. (Stoffová, 2004), (Végh 2017), (Udvaros, 2017). Vývojom nových programovacích jazykov, prostredí, nástrojov ako aj počítačovej grafiky, implementácia, vizualizácia, interaktivita ASM sa dostala na vyššiu úroveň (Stoffová, 1997a, 1997b, 1997c), (Stoffová, 1998). V súčasnosti na tvorbu takýchto softvérových aplikácií a na ich implementáciu a vizualizáciu sa využívajú moderné nástroje ako sú: Flash, Java, HTML5 a ďalšie podporné prostredia, mikrosvety (Czakóová – Stoffová, 2012), (Czakóová, 2017). knižnice objektov, predprogramovaných stavebných elementov a pod. Vytvorila sa webová stránka <http://anim.ide.sk>, kde sú prezentované voľne dostupné animačno-simulačné modely na podporu vzdelávania.

6 DIDAKTICKÉ ZÁSADY A RADY NA POUŽÍVANIE ANIMAČNO-SIMULAČNÝCH MODELOV VO VYUČOVANÍ ALGORITMOV

Na efektívne využívanie animačno-simulačných modelov je nutné:

- jednoznačne stanoviť didaktický cieľ ich využívania;
- starostlivo pripraviť simulačné experimenty na dosiahnutie didaktického cieľa;
- explicitne stanoviť úlohy/otázky na riešenie/na zodpovedanie;
- rozvíjať u používateľov schopnosť stanoviť výskumné hypotézy;
- rozvíjať schopnosť pomocou simulácie potvrdiť alebo vyvrátiť stanovenú hypotézu;
- rozvíjať pozorovací talent, kreativitu, kritické myslenie, predvídavosť a zvedavosť;
- a nie v poslednom rade správne interpretovať výsledky simulačných experimentov (Stoffová – Végh, 2010, 2013).

7 ZÁVER

Simulačné modely sú vhodným nástrojom prezentovania učebného materiálu. Sú koncentrovanou formou informácií a poznatkov o objekte poznania a skúmania a o jeho správaní sa. Vizualizované simulačné experimenty umožňujú získať nové poznatky o modelovanom objekte na základe vlastného pozorovania aktívnym zapojením sa edukantov do poznávacieho procesu (Stoffová, 2010). Simulačné modely zohrávajú významnú úlohu vo vzdelávaní hlavne pri tzv. deep learningu.

V dotazníkovom prieskume (Végh, 2017) viac ako 80 % študentov odpovedalo, že uprednostňuje animácie a prácu s animačno-simulačnými modelmi pred ostatnými formami a metódami učenia.

Interaktívne animácie, ktoré je možné zabudovať do edukačnej webovej stránky alebo inej vzdelávacej aplikácie (na adrese <http://anim.ide.sk>), sú voľne prístupné, môžu obohatiť a oživiť vyučovanie a majú veľkú motivujúcu silu. Tieto interaktívne animácie môžu byť užitočným nástrojom na výučbu algoritmov, údajových štruktúr (aj dynamických) a programovania.

Výsledky pedagogických experimentov prezentovaných v práci (Végh, 2017) súvisiacich s používaním interaktívnych animácií ukazujú, že animácie a vizualizácie zohrávajú dôležitú úlohu aj vo výučbe informatiky. S animáciou algoritmov môžu začínajúci programátori ľahšie a rýchlejšie porozumieť pojmom z oblasti algoritmizácie a programovania, pochopiť rôzne algoritmické procesy a rozdiely medzi nimi, a to zábavným a hravým spôsobom.

Je však dôležité, aby pri plánovaní, navrhovaní a vývoji nového animačno-simulačného modelu boli rešpektované princípy a návrhy tvorby multimedialných uvedených v našich publikáciách. aby boli správne zvolené grafické znázornenia a aby bola do animácie zabudovaná primeraná interaktivita. Starostlivo navrhnuté, implementované a testované interaktívne animácie môžu byť užitočným vzdelávacím nástrojom pre učiteľov aj študentov.

8 POĎAKOVANIE

Článok vznikol s podporou projektu KEGA 012TTU-4/2018 Interaktívne animačno-simulačné modely vo vzdelávaní.

9 BIBLIOGRAFICKÉ ODKAZY

- [1] CZAKÓOVÁ, Krisztina. Élmény alapú programozás oktatás. In: Bukor, J. – Gubo, Š. (eds.): *Zborník medzinárodnej vedeckej konferencie Univerzity J. Selyeho – 2017 „Hodnota, kvalita a konkurencieschopnosť – výzvy 21. storočia“*. Sekcia informatických vied a IKT. Proceeding of electronic version of reviewed contributions (CD-ROM). Komárno: Univerzita J. Selyeho, 2017, s. 27–33. ISBN 978-80-8122-221-4.
- [2] CZAKÓOVÁ, Krisztina – STOFFOVÁ, Veronika: Animačné modely v didaktických aplikáciách vytvorených v LogoMotion. In: Hájková, E. – Vémolová, R. (ed.): *XXX. International Colloquium on the Management of Educational Process*. Proceeding of electronic version of reviewed contributions (CD-ROM). Brno: Univerzita obrany, 2012. s. 66–69. ISBN 978–80-7231-865-0.
- [3] GABALOVÁ, Veronika: Problémové vyučovanie programovania. [Dizertačná práca]. Univerzita Konštantína Filozofa v Nitre. Fakulta prírodných vied. Školiteľ: Prof. Ing. Veronika Stoffová, CSc. Stupeň odbornej kvalifikácie: PhD. Odbor: Teória vyučovania informatiky, Nitra: FPV, 2010. 147 s.
- [4] KOTVA, Milan: Dohoda o chápaní pojmu „simulace systému“. *Automatizace*, roč. 29, 1986, č. 12.
- [5] LEPIL, Oldřich – VACHEK Jaroslav: *Modelování a modely ve vyučování fyzice*. 2. vydání, 1986, SPN - Státní pedagogické nakladatelství, Praha, 208 s.
- [6] STOFFA, Ján – STOFFOVÁ, Veronika: Terminológia – celoživotný údel každého edukátora. In: *Eruditio – Educatio*, roč. 6, 2011, č. 1, s. 3–14. ISSN 1336-8893.
- [7] STOFFA, Ján – STOFFOVÁ, Veronika: *Terminológia informatiky a IKT*. 1. vyd. Trnava: Trnavská Univerzita / VEDA, 2017. 245 s. ISBN 978-80-568-0065-2 (monografia).
- [8] STOFFOVÁ, Veronika: Modelovanie, simulácia, animácia – účinný prostriedok vyučovacieho procesu. In: *Zborník 5. vedeckej konferencie s medzinárodnou účasťou CO-MA-TECH*. Trnava: Materiálovo-technologická fakulta STU, 1997, s. 153–156. ISBN 80-227-0979-4.
- [9] STOFFOVÁ, Veronika: Počítačové modelovanie a simulácia v informačných technológiách. *X. DIDMATECH '97: Zborník*. 1. vyd. Nitra: Pedagogická fakulta Univerzity Konštantína Filozofa, 1997, s. 150–153.
- [10] STOFFOVÁ, Veronika: Využívanie didaktických modelov vo vyučovacom procese. In: *Zborník konferencie DIDINFO '98*. Banská Bystrica: MC MB a FPV UMB, 1998, s. 35–42. ISBN 80-8041-261-8
- [11] STOFFOVÁ, V. et. al.: Informatika, informačné technológie a výpočtová technika: Terminologický a výkladov slovník. 1. vyd. Nitra: Fakulta prírodných vied UKF v Nitre, 2001. 230 s. ISBN 80-8050-450-4.
- [12] STOFFOVÁ, Veronika: *Počítač – univerzálny didaktický prostriedok*. 1. vyd. Nitra: Univerzita Konštantína Filozofa, Fakulta prírodných vied, 2004. 172 s. ISBN 80-8050-450-4.
- [13] STOFFOVÁ, Veronika: Demonštračné animácie pre elektronické vzdelávanie. (Demonstrative animation for electronic education). In: Hájková, E. – Vémolová, R. (eds.): *XXVIII. International Colloquium the Management of Educational Process*. Brno: Univerzita obrany, Fakulta ekonomiky a managementu, 2010, s. 65–66. (príspevok na CD 5 s.) ISBN 978-80-7231-722-6.

- [14] STOFFOVÁ, Veronika: The Importance of Didactic Computer Games in the Acquisition of New Knowledge. In: *The European Proceedings of Social & Behavioural Sciences EpSBS*. pp. 676-688. eISSN 2357-1330 HTTP://DX.DOI.ORG/10.15405/EPBSBS.2016.11.70.
- [15] STOFFOVÁ, Veronika: Conceptual cybernetic model of teaching and learning. In: *Mathematical Modeling*, year 1, 2017, issue 2, p. 80–83. ISSN (WEB) 2603-2929, Print 2535-0986.
- [16] STOFFOVÁ, Veronika: Computer Games as a Tool for Development of Algorithmic Thinking In: *The European Proceedings of Social & Behavioural Sciences EpSBS*. pp. xxx-xxx. <http://dx.doi.org/> Corresponding Author: Selection and peer-review under responsibility of the Organizing Committee of the conference eISSN: 2357-1330.
- [17] STOFFOVÁ, Veronika: – KOSTOLANSKÁ, Jana: AVIS v tvorbe grafických animačných modelov. In: *Zborník medzinárodného workshopu – Pokroky tvorby a využiti simulačných modelů – Krnov*. Ostrava: VŠB – TU, 1997, 5. s.
- [18] STOFFOVÁ, Veronika – STOFFA, Ján: Tvorba elektronických slovníkov a encyklopédií. *Technológia vzdelávania v príprave učiteľov prírodovedných a technických predmetov*. Prešov: Prešovská univerzita, 2011. s. 325–333. ISBN 978-80-555-4.
- [19] STOFFOVÁ, Veronika – STOFFA, Ján: Terminological dictionary as the organic part of electronic textbook. *AGRIAMEDIA 2011*, Eger: EKTf, 2011.
- [20] STOFFOVÁ, Veronika – VÉGH, Ladislav: Szemléltető animációk a programozásban. (Demonštračné animácie v programovaní). In: *INFODIDACT 2010, 3. Informatika Szakmódszertani Konferencia*. Editor Zsakó, Szombathely, 2010. (príspevok na CD 6 s.).
- [21] STOFFOVÁ, Veronika – VÉGH, Ladislav: Grafické modely dynamických údajových štruktúr. In: Havelka, M., Chráska, M., Klement, M., Serafin, Č. (ed.): *Trendy ve vzdělávání 2013*. Olomouc: agentura GEVAK s.r.o., 2013. 309–314. s. ISBN 978-80-86768-52-6 / ISSN 1805-8949.
- [22] STOFFOVÁ, Veronika – VÉGH, Ladislav: Tvorba animačno-simulačných modelov v rôznych prostrediach. Paper presented at the Informatika v škole a v praxi, Ružomberok.
- [23] STOFFOVÁ, Veronika – VÉGH, Ladislav: Prieskum vhodnosti využívania animačno-simulačných modelov algoritmov vo vzdelávaní. In: *New methods and technologies in education and practice: XXVII. DDDMATTECH 2014: International and Scientific Conference*. Editori Veronika Stoffová a Miroslav Chráska. 1. vyd. Olomouc: Agentura Gevak, 2014, s. 36–44. ISBN 978-80-86768-96-0.
- [24] STOFFA Veronika – UDVAROS József: Opportunities of visualization of object-oriented programming for learning. In: *XXVI. DIDMATTECH 2013: New technologies in science and Education*. Red. Veronika Stoffová, 1. vyd. Győr – Komárno: University of West Hungary Press, Győr, University of West Hungary, Győr – J. Selye University, Komárno, 2014, s. 142–146. ISBN 978-963-334-184-1.
- [25] STOFFOVÁ, Veronika – UDVAROS, József: Grafické modely štandardných údajových štruktúr a ich význam vo vyučovaní programovania. In: Havelka, M., Chráska, M., Klement, M., Serafin, Č. (ed.): *Trendy ve vzdělávání 2013*. Olomouc: agentura GEVAK s.r.o., 2013. 323–328. s. ISBN 978-80-86768-52-6 / ISSN 1805-8949.
- [26] ŠTOFOVÁ, Veronika: Simulácia vyučovacieho procesu. In: *Simulace systému a vědeckotechnické výpočty*. Ostrava: Dům techniky ČSVTS, 1988, s. 255–259
- [27] ŠTOFOVÁ, Veronika: Využívanie modelovania a simulácie v počítačom podporovanej výučbe. In: *Zborník medzinárodnej konferencie MEDACTA 91: č. 1*. Nitra: Pedagogická fakulta, 1991, s. 78–87.
- [28] ŠTOFOVÁ, Veronika: Modely triediacich algoritmov. In: *Acta MOSIS č. 47: Zborník: Vybrané problémy simulačných modelů*. Ostrava: Pobočka ČSVTS VÚHŽ, 1991, s. 86–87.

- [29] VÉGH, Ladislav – STOFFOVÁ, Veronika: An interactive animation for learning sorting algorithms: How students reduced the number of comparisons in a sorting algorithm by playing a didactic game. In: *Teaching Mathematics and Computer Science*. Debrecen: Institute of Mathematics – University of Debrecen, 14th volume, issue one, 2016/1, s. 45–62. ISSN 1589-7389.
- [30] VÉGH, Ladislav – STOFFOVÁ, Veronika: Algorithm Animations for Teaching and Learning the Main Ideas of Basic Sortings. In: *Informatics in Education*, Lithuania: Vilnius University. Vol. 16, No. 1, 2017, p. 121–140. ISSN: 1648-5831 (printed), 2335-8971 (online). DOI: 10.15388/infedu.2017.07. Dostupné na: https://www.mii.lt/informatics_in_education/htm/infedu.2017.07.htm. WOS:000399818000007.
- [31] VÉGH, Ladislav: A programozás tanulásának és tanításának támogatása elektronikus tananyagba beépíthető interaktív animációs modellekkel. (Support for learning and teaching programming by incorporating interactive animation models into electronic teaching tools). [Dizertačná práca]. Budapest: Eötvös Loránd Tudományegyetem, Informatika Kar, 2017, 202 s.
- [32] UDVAROS, József: A digitális technológiák szerepe és lehetőségei a programozás tanításában. (Význam a úloha digitálnych technológií vo vyučovaní programovania.) [Dizertačná práca]. Budapest: Eötvös Loránd Tudományegyetem, Informatika Kar, 2017, 104 s. The role and opportunities of digital technology in teaching programming.
- [33] URQUIZA-FUENTES, J., – VELAZQUEZ-ITURBIDE, J. A.: Toward the effective use of educational program animations: The roles of student's engagement and topic complexity. *Computers & Education*, 67, 178–192. doi:10.1016/j.compedu.2013.02.013.

The Role of Sprego Programming in Teaching Informatics

Maria Csernoch
University of Debrecen
Kassai út 26.
Debrecen
+36-52-512-900/75128
csernoch.maria@inf.unideb.hu

ABSTRACT

Sprego (Spreadsheet Lego) programming combines the advantages of the graphical interface of spreadsheet applications, functional languages, concept-based problem-solving approaches, knowledge-transfer, and schema construction. The approach based on this combination of methods has been proved more effective in developing students' computational thinking skill and gaining spreadsheeting and fundamental programming knowledge than the traditional surface approach and the code-centred methods. The question is how Sprego can be smoothly integrated into existing curricula and how students can benefit the most from this novel approach.

Keywords

Sprego, spreadsheeting, programming, computational thinking, problem-solving, knowledge-transfer

1 INTRODUCTION

1.1 Spreadsheeting should take the responsibility?

It is widely accepted that spreadsheeting and spreadsheet-teaching is nothing more than computer skills [1], low level routine tasks [2], and should be banned from education, because they do more harm than good [3]. “Instead of children bored out of their minds being taught how to use Word and Excel by bored teachers...” [3], “serious” computer sciences should be taught [3]. The undesired side effect of these approaches is that spreadsheeting itself is blamed and ignored [4]–[8]. However, it is found that this is only a form of scapegoating, trying to find a shelter and explanation for teachers' lack of ability and skills to teach this topic effectively [4]–[13], and for downgrading spreadsheet programming [15]–[22] and introductory functional programming [23]. In the meantime, proof has been found that it is not spreadsheeting which should be blamed, but the low-mathability methods with which spreadsheeting is conveyed to students and end-users, and put into action [4]–[23].

1.2 Deep and surface computer problem-solving approaches

It has been found that the keywords to good spreadsheet practice [25] and teaching spreadsheeting effectively [12][13] are ‘problem-solving approaches’ [5][9][11][26][27][28] and ‘teaching goals’ [10]. Panko revealed and explained [5], based on the work of Kahneman [29], that spreadsheet errors occur because during problem-solving, slow thinking is applied, instead of reliable fast thinking. However, we have found that Panko's finding only covers one aspect of the problem; namely that fast thinking should be applied. The other aspect, that ‘thinking is bad’ should be reconsidered. The explanation and the solution to this paradox lies in the consideration of the different problem-solving approaches [9][30].

The typology of computer problem-solving approaches registers two hypernym categories: deep and surface approaches, and within these, five further hyponym categories (Figure 1). Among these five hyponym categories the two lowest, the information- and the trial-and-error wizard based, are not problem-solving, but low-mathability approaches [26], primarily focusing on tools, instead of real

problems. What Panko found is that end-users primarily apply Level 1 and Level 2 approaches, which are slow thinking activities leading to erroneous documents and data; namely browsing interfaces and wizards until a questionable output is generated [4][5][6][16].

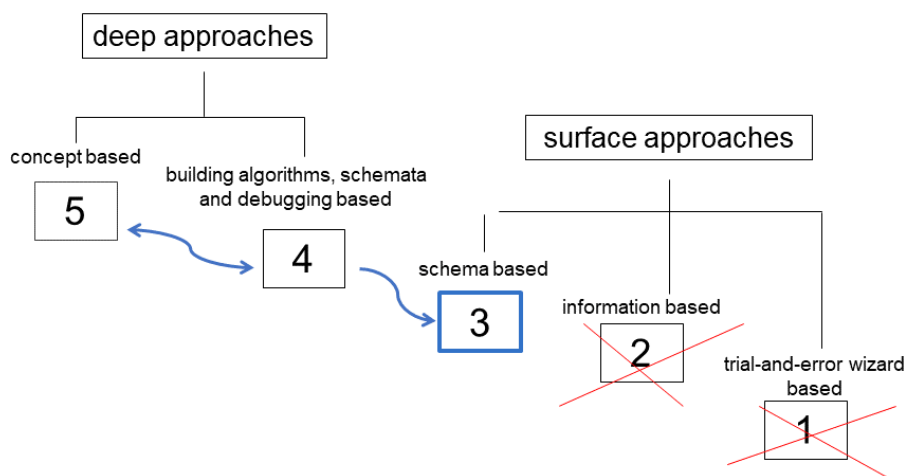


Figure 1: The typology of computer-problem solving approaches

However, thinking itself is not bad, only when it is applied in incorrect and non-effective ways. Considering the two highest hyponym categories, the concept based on Pólya [31] and building algorithms, schemata and debugging based on programming [23], cognitive load [32], and mathability [26] studies, it is clear that slow thinking has its own extremely important role in problem-solving, but should be applied in the right place.

Being aware of all these considerations, there is only one hyponym category which requires and applies fast thinking – the schema based problem-solving approach (Figure 1, Level 3). At this level, soundly built schemata should be brought into problem-solving from long-term memory. With this method problem-solving becomes fast and reliable.

In cases when schemata are not available, the two highest levels should be applied, focusing on solving the actual problem and providing opportunities for practising and building up missing schemata [29][31][32] and concepts [33].

One further aspect of problem-solving is how education and teachers can guide students to work effectively, fast, without errors, and remain open to real-world problems. It is found that developing students' computational thinking skills [34] is a heavy mental load, which teachers should lighten [35]. It has also been proved that at present there is no such thing as a digital native; relying on digital tools is not enough for effective problem-solving [12][13]. Furthermore, teachers should be open to all the aspects of Technological Pedagogical Content Knowledge [36] and the Meaning System Model [37].

2 SPREGO PROGRAMMING

Based on the theoretical background and on experiences of the limited effectiveness of current spreadsheet practices detailed in the Introduction, a deep approach programming method is introduced in spreadsheeting and spreadsheet teaching, entitled Spreadsheet Lego, or Sprego for short [19][20].

2.1 Problem-solving in Sprego

In general, Sprego programming is a concept-based problem-solving approach for data-management (Figure 1, Level 5), based on the methodology of Pólya, published in his well-known work, How To Solve It [31]. In the Sprego context, problem-solving is broken down into four steps adapted from Pólya, with the necessary backtracking, when students run into dead ends.

- First: You have to understand the problem.
- Second: Find the connection between the data and the unknown. You may be obliged to consider auxiliary problems if an immediate connection cannot be found. You should obtain eventually a plan of the solution.
- Third: Carry out your plan.
- Fourth: Examine the solution obtained.

Another feature of Sprego is the application of the coaching method [38][39], where students are guided through the steps of the problem-solving process. The essence of this method is that problems are presented and then discussed in detail, involving as many students as possible, where the primary role of the teacher is to keep track of the steps of problem-solving and clarify both the global concept and the necessary details. In general, “coaching is a guided discovery learning in which students construct their new knowledge from their existing knowledge.” [39].

In close connection with the coaching method three further concepts must be mentioned where the existing knowledge of the students are focused on: TCPK [36], the Meaning System Model [37], and knowledge-transfer. These three concepts are integrated in Sprego.

- Meaningful, real-world problems are presented [36][41][42] to motivate students and make data-management interesting for them.
- Teachers believe in the “Incremental” nature of science (e.g. dynamic contextual knowledge) and have as their goal that students should develop a deep understanding and appreciation of science [37].
- There is an intensive application of knowledge brought into classes from other sources [41][42].

To build up long-lasting and transferable knowledge and to allow fast thinking to work properly, a great emphasis is put on building schemata and recalling them [29][32]. To achieve this goal the Sprego tools are detailed in the following section.

2.2 Sprego tools

In Sprego, the data-management and the programming aspects run parallel. The programming tool is the functional language of spreadsheet programs, while the data-management is based on real data tables [41][42].

2.2.1 The concept of function

One of the reasons that functional languages are more effective as introductory languages than other text-based languages [23] is that the concept of function is brought into classes from mathematics, and calling functions is simple; consequently the focus is on the problem, not on the technical details. In spreadsheet interfaces the tools are more user-friendly than in classical functional languages, a feature which also serves non-professional end-users better. In programming, and consequently in spreadsheeting, n-ary functions are called, with various data types as arguments. This use and practice of n-ary functions can help maths studies by providing examples of a wide range of domains and ranges.

One further aspect of Sprego is the application of multilevel functions. This feature is mostly avoided in spreadsheeting, as it is considered high-level mathematics [24], and therefore not suitable for primary and secondary school students. However, it has been proved that when introducing the concept of multilevel functions with the proper methodology, students are open to this knowledge and are able to digest it without any difficulty [14].

2.2.2 Sprego functions

Spreadsheet programs are considered user-friendly due to the high number of functions included, according to software companies and most textbooks. However, it has been proved that memorizing functions and their arguments does not improve spreadsheet problem-solving [40] and not knowing them leads to erroneous spreadsheet practice [4][5][6][16]. These findings are in complete accordance

with cognitive load theory [32] and the concepts of thinking fast and slow [5][29]. To avoid the failure of user-friendly spreadsheet-teaching, a set of Sprego functions were defined, limited to a dozen general purpose functions (Table 1). The dozen functions are categorized into three subsets to further lighten the cognitive load. The Sprego Text and the Sprego Number categories include functions whose domain is text and number data types, respectively. The Sprego Pro category has four general purpose miscellaneous functions which require a more advanced level of computational thinking skills.

Table 1: Sprego functions

Sprego Text	Sprego Number	Sprego Pro
LEFT()	SUM()	IF()
RIGHT()	MIN()	ISERROR()
LEN()	MAX()	MATCH()
SEARCH()	AVERAGE()	INDEX()

However, we must call attention to the fact that the set of Sprego functions is an open set, meaning that further functions can be added, if the problem so requires, and/or the students' abstraction level increases, and teachers are sure that the dozen Sprego functions are stored in long-term memory.

2.2.3 Array formulas

Arrays and along with them, iteration, are introduced from the very beginning of Sprego programming [17][21][22]. This is another concept whose simplified version arrives from mathematics but plays a crucial role in programming. In spreadsheet interfaces the declaration of arrays is a simple selection, and as such, does not require a deep understanding of memory allocation and the syntactic rules of creating arrays.

By making the selection, students practice and, through practicing, understand that with one array the same operation can be carried out on all the items of the array.

2.2.4 Unplugged tools

Teaching and learning programming, developing students' computational thinking skills and computer problem-solving abilities are extremely demanding on all the participants of the teaching-learning process. To further lighten the cognitive burden, within the framework of Sprego, several unplugged tools were developed and introduced in classes.

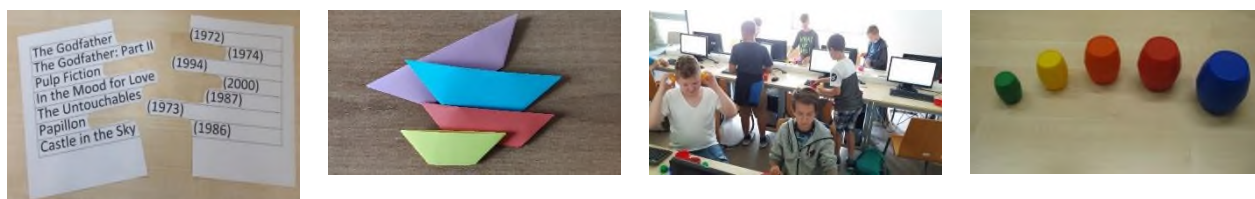


Figure 2: Various unplugged tools used in Sprego classes

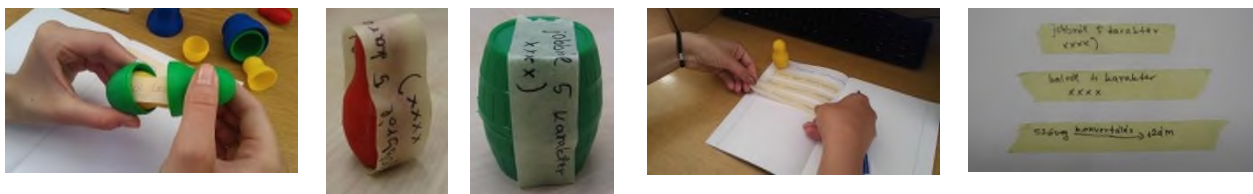


Figure 3: The application of matryoska dolls for understanding multilevel functions and input-output data flow between levels

To make students understand how the output of one function can serve as one of the inputs of the function on the next level, we use matryoska doll avatars, their 3D printed version, or origami tools folded by the students (Figure 2 and Figure 3).

One further unplugged method frequently applied in Sprego classes is role-play. Students carry out the algorithm by themselves following the instructions of the teacher or a selected director. These role-plays are accompanied by further unplugged tools which are usually prepared by the students.

2.2.5 Semi-unplugged tools – webtables

Sprego relies on real-world data and problems – authentic data sources – to carry out reliable data-management. To provide these data sources, webtables – tables or table-like structures embedded in webpages – are converted into datatables and problems are formulated based on these data [41]. The wide range of webtables available on the internet makes data-management classes motivating, students are interested in the content, and find these problems valuable and useful [43].

One further advantage of the use of these authentic sources is the conversion process from webtable to datatable. The webtable→datatable conversion involves various knowledge-transfer items and serves as a practicing platform for various topics within informatics and other school subjects and sciences [42]. During the development of the methodology and its use in classes, we concluded that the conversion process fulfils all the requirements of TCPK and of the teachers' belief in the incremental nature of science. One further 'side-effect' of the webtable→datatable conversion is that the process can serve as webtable semantic validator, where, based on the algorithm of the conversion, we are able to tell whether the webtable is correctly designed and/or formatted [41].

2.2.6 Semi-unplugged tools – applications

For further assistance, 2D and 3D Sprego applications were developed [44][45][46]. On the one hand, these applications function as role-plays with our matryoska doll-avatars, while at the same time they imitate the Evaluate Formula command of spreadsheet applications. The role-play and the evaluation sheets appear side-by-side on a graphical interface, and the animations are played simultaneously in the sheets, relying on the advantages of multisensorial observation (Figure 4).



Figure 4: Samples of the 2D and 3D Sprego applications

3 SUMMARY

The essence of Sprego (Spreadsheet Lego) programming is presented in this paper. It is an approach to teaching spreadsheeting and introductory programming within spreadsheet interfaces. The main idea of Sprego is that programming is taught with minimal tools, letting students focus on the problem, instead of on the details of the interfaces.

Most of the methods are adapted from classical programming-teaching approaches, with a wide extension of unplugged and semi-unplugged tools. The importance of these tools is to make programming more understandable, to lighten the cognitive load in the teaching-learning process, and to make Sprego programming enjoyable.

It has been proved that this algorithm- and programming-oriented approach is more effective than the traditional spreadsheet-teaching methods, provides students with long-lasting knowledge, and serves as an introductory programming language to further studies in computer sciences.

4 REFERENCES

- [1] HROMKOVIČ, J. *Algorithmic Adventures*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. Retrieved July 21, 2016, from <http://link.springer.com/10.1007/978-3-540-85986-4>.
- [2] BELL, T. and NEWTON, H. *Unplugging Computer Science*. In Improving Computer Science Education, 3012 (Eds.) Kadijevich, D. M. Angeli, C. and Schulte, C., Routledge.
- [3] GOVE, M. Michael Gove speech at the BETT Show 2012. Published 13 January 2012. Digital literacy campaign. Retrieved March 3, 2019, from <http://www.theguardian.com/education/2012/jan/11/digital-literacy-michael-gove-speech>.
- [4] PANKO, R. R. *What we know about spreadsheet errors*. Journal of End User Computing, 10, pp. 15–21, 1998.
- [5] PANKO, R. R. *The Cognitive Science of Spreadsheet Errors: Why Thinking is Bad*. Proceedings of the 46th Hawaii International Conference on System Sciences, January 7–10, 2013, Maui, Hawaii.
- [6] GARRETT, N. *Textbooks for Responsible Data Analysis in Excel*. Journal of Education for Business, 0: 1–6, 2015, Taylor & Francis Group, LLC, ISSN: 0883-2323 print / 1940-3356 online, DOI: 10.1080/08832323.2015.1007908, 2015.
- [7] PAPP, P. and CSERNOCH, M. *Spreadsheeting Is Problem-solving?* In Hungarian *A táblázatkezelés is problémamegoldás?* INFODIDACT'2018, 11. Informatika Szakmódszertani Konferencia, Zamárdi, pp.187–202, 2018.
- [8] KO, A. J. ABRAHAM, R. BECKWITH, L. BLACKWELL, A. BURNETT, M. M. ERWIG, M. SCAFFIDI, C. LAWRENCE, J. LIEBERMAN, H. MYERS, B. A. ROSSON, M. B. ROTHERMEL, G. SHAW, M. and WIEDENBECK, S. *The State of the Art in End-User Software Engineering*. ACM Computing Surveys, 43 (3) pp. 1–44, 2011.
- [9] CSERNOCH, M. *Thinking Fast and Slow in Computer Problem Solving*. Journal of Software Engineering and Applications, Vol.10 No. 01, 2017, Article ID:73749, 30 pages 10.4236/jsea.2017.101002.
- [10] CSERNOCH, M. and BIRÓ, P. *Edu-Edition Spreadsheet Competency Framework*. Proceedings of the EuSpRIG 2017 Conference "Spreadsheet Risk Management", Imperial College, London, pp.121–136 ISBN: 978-1-905404-54-4, 2018.
- [11] CSERNOCH, M. and BIRÓ, P. *Teaching methods are erroneous: approaches which lead to erroneous end-user computing*. EuSpRIG2016. London. Retrieved July 21, 2016, from <http://www.eusprig.org/mcsernoch-2016.pdf>.

- [12] CSERNOCH, M. and BIRÓ, P. *Are digital natives spreadsheet natives?* European Spreadsheet Risk Interest Group Conference, EuSpRIG2017, 2017, Imperial College, London.
<https://arxiv.org/abs/1909.00865>, 2019.
- [13] KIRSCHNER, P. A. & DE BRUYCKERE, P. *The myths of the digital native and the multitasker*. Teaching and Teacher Education, 2017, 67 pp.135–142.
- [14] CSAPÓ, G. CSERNOCH, M. and Abari, K. *Sprego: Case Study on the Effectiveness of Teaching Spreadsheet Management with Schema Construction*. Education and Information Technologies, 2019, pp. 1–21. ISSN: 1360-2357 (Print) 1573-7608 (Online).
- [15] ANGELI, C. *Teaching Spreadsheets: A TPCK Perspective*. In Improving Computer Science Education. (Eds.) D. M. Kadijevich, C. Angeli, and C. Schulte. Routledge, 2013.
- [16] EuSpRIG, European Spreadsheet Risk Interest Group *EuSpRIG Horror Stories*.
<http://www.eusprig.org/horror-stories.htm>, 2019.
- [17] SESTOFT, P. *Spreadsheet technology*. Version 0.12 of 2012-01-31. IT University Technical Report ITU-TR-2011-142. IT University of Copenhagen, December 2011.
- [18] JONES, S. P. BLACKWELL, A. and BURNETT, M. *A User-Centred Approach to Functions in Excel*. International Conference on Functional Programming (ICFP'03), Uppsala, 2003.
- [19] CSERNOCH, M. *Programming with Spreadsheet Functions: Sprego*. In Hungarian, *Programozás táblázatkezelő függvényekkel – Sprego*. Műszaki Könyvkiadó, Budapest, 2014.
- [20] CSERNOCH, M. and BIRÓ, P. *Sprego Programming*. Spreadsheets in Education (eJSiE), 8(1). Retrieved July 21, 2016, from <http://epublications.bond.edu.au/ejsie/vol8/iss1/4>, 2015.
- [21] WALKENBACH, J. and WILCOX, C. *Putting basic array formulas to work*. Retrieved June 15, 2019 from <http://office.microsoft.com/en-us/excel-help/putting-basic-array-formulas-to-work-HA001087292.aspx?CTT=5&origin=HA001087290>, 2003.
- [22] WILCOX, C. and WALKENBACH, J. *Introducing array formulas in Excel*. Retrieved June 15, 2019 from <http://office.microsoft.com/en-us/excel-help/introducing-array-formulas-in-excel-HA001087290.aspx>, 2003.
- [23] BOOTH, S. *Learning to program: A phenomenographic perspective*. Gothenburg, Sweden: Acta Universitatis Gothoburgensis, 1992.
- [24] ICAEW *Spreadsheet competency framework*. A structure for classifying spreadsheet ability in finance professionals. Retrieved November 21, 2016 from <http://www.icaew.com/-/media/corporate/files/technical/information-technology/it-faculty/spreadsheet-competency-framework.ashx>, 2016.
- [25] ICAEW THOUGHT LEADERSHIP *Twenty principles for good spreadsheet practice*. Third Edition. Retrieved November 15, 2019 from <https://www.icaew.com/-/media/corporate/files/technical/information-technology/excel-community/20-principles-of-good-spreadsheet-practice-2018.ashx>, 2018.
- [26] BARANYI, P. and GILÁNYI, A. *Mathability: Emulating and Enhancing Human Mathematical Capabilities*. In 2013 4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom2013) pp.555–558, December 2–5, 2013, Budapest, Hungary.
DOI=<https://ieeexplore.ieee.org/document/6719309/>.
- [27] BIRÓ, P. and CSERNOCH, M. *The mathability of computer problem solving approaches*. In 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom2015), pp.111–114. DOI=<http://doi.org/10.1109/CogInfoCom.2015.7390574>.
- [28] BIRÓ, P. and CSERNOCH, M. *The mathability of spreadsheet tools*. In 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom2015), pp.105–110. DOI= <http://doi.org/10.1109/CogInfoCom.2015.7390573>.
- [29] KAHNEMAN, D. *Thinking, Fast and Slow*. New York: Farrar, Straus; Giroux, 2011.

- [30] CSERNOCH, M. and BIRÓ, P. *Computer Problem Solving*. In Hungarian: *Számítógépes problémamegoldás*. TMT, Tudományos és Műszaki Tájékoztatás, Könyvtár- és információtudományi szakfolyóirat, 2015, 62(3), pp.86–94.
- [31] PÓLYA, G. *How To Solve It. A New Aspect of Mathematical Method*. (2nd Edition 1957), Princeton University Press, Princeton, New Jersey, 1954.
- [32] SWELLER, J. AYRES, S. and KALYUGA, S. *Cognitive Load Theory*. Springer New York Dordrecht Heidelberg London, 2011.
- [33] SKEMP, R. *The Psychology of Learning Mathematics*. Lawrence Erlbaum Associates, New Jersey, USA, 1971.
- [34] WING, J. M. *Computational thinking*. Communications of the ACM, 2006, 49(3), 33. DOI=<http://doi.org/10.1145/1118178.1118215>.
- [35] KIRSCHNER, P.A., SWELLER, J., and CLARK, R.E. *Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist Discovery, Problem-Based, Experiential, and Inquiry-Based Teaching*. Educational Psychologist, 2006, 41(2), pp.75–86.
- [36] MISHRA, P. and KOEHLER, M. J. *Technological Pedagogical Content Knowledge: A new framework for teacher knowledge*. Teacher College Record, 2006, 108(6), pp.1017–1054.
- [37] CHEN, J. A. MORRIS, D. B. and Mansour, N. *Science Teachers' Beliefs. Perceptions of Efficacy and the Nature of Scientific Knowledge and Knowing*. In International Handbook of Research on Teachers' Beliefs, (Eds.) Fives, H. & Gill, M. G. Routledge, 2015, pp.370–386.
- [38] BURTON, R. R. and BROWN, J. S. "An investigation of computer coaching for informal learning activities," International journal of man-machine studies, vol. 11, no. 1, pp.5–24, 1979.
- [39] CHMIELEWSKA, K. and MATUSZAK, D. *Mathability and coaching*. 8th IEEE International Conference on Cognitive Infocommunications, (CogInfoCom2017), pp.427–432.
- [40] NAGY, T. *Knowledge-transfer in Data-management*. In Hungarian: *Tantárgyközi kapcsolatok és tudástranszfer elemek az adat-kezelésben*. TDK dolgozat, DEIK, 2019.
- [41] CSERNOCH, M. and DANI, E. *Data-structure validator: an application of the HY-DE model*. In: IEEE 8th International Conference on Cognitive InfoCommunications: (CogInfoCom2017), Debrecen, Hungary, Piscataway (NJ): IEEE Computer Society, pp.197–202. (ISBN:978-1-5386-1264-4).
- [42] CSERNOCH, M. *From webtables to datatables*. EuSpRiG 2019 Conference, London, accepted.
- [43] BIRÓ, P. and CSERNOCH, M. *Semi-Unplugged Tools for Building Algorithms with Sprego*. TOJET: Turkish Online Journal of Educational Technology Spec. Issue for INTE:(2), 2017, pp.946–957.
- [44] CSAPÓ, G. and SEBESTYÉN, K. *Educational Software for the Sprego Method*. The Turkish Online Journal of Educational Technology, INTE 2017 October, pp.986–999. Retrieved November 21, 2019 from http://www.tojet.net/special/2017_10_1.pdf.
- [45] GULÁCSI, Á. and DIENES, N. *3D Software Environment for Educational Sprego Programming*. The Turkish Online Journal of Educational Technology, INTE 2018 November (2), pp.837–844. Retrieved November 21, 2019 from http://www.tojet.net/special/2018_12_3.pdf.
- [46] SEBESTYÉN, K., CSAPÓ, G. and CSERNOCH, M. *Visualising Sprego Inequality Problems With 2D Representations*. The Turkish Online Journal of Educational Technology, INTE 2018 November (2), pp.888–898. Retrieved November 21, 2019 from http://www.tojet.net/special/2018_12_3.pdf.

Code review v škole a na programátorskom krúžku

Code Review at School and at Programming Club

Iveta Demková
Univerzita Komenského v Bratislave
FMFI KDMFI, Mlynská dolina
842 48 Bratislava
Slovensko
ivetskacs@gmail.com

Zuzana Kubincová
Univerzita Komenského v Bratislave
FMFI KDMFI, Mlynská dolina
842 48 Bratislava
Slovensko
kubincova@fmph.uniba.sk

ABSTRAKT

Edukačné code review je aktivita, ktorá nielen pomáha pripraviť budúcich programátorov do praxe, ale aj učí študentov pracovať s kódom iným spôsobom. V školskom prostredí sa s aktivitami zameranými na code review stretávame najmä na vysokej škole. V našom výskume sme sa sústredili na nižšie úrovne vzdelávania a v predchádzajúcich publikáciách sme prezentovali výsledky využitia code review na strednej škole. Experimentálne sme však realizovali code review aj na programátorskom krúžku. Tento príspevok prináša popis realizovaného výskumu, pričom výsledky porovnávame s výsledkami zo strednej školy. Uvádzame aj prehľad prínosov, ale aj problémov, ktoré takéto aktivity môžu priniesť do vyučovania.

ABSTRACT

Educational code review is an activity that not only helps prepare future programmers into practice, but also teaches students to work with code in a different way. In the educational settings, activities focused on code review are mainly encountered at universities. In our research, we focused on lower levels of education and in our previous publications we presented the results of using code review at secondary school. Experimentally, we also tried to use the code review activities on the sessions at the leisure time activities club. This paper provides a description of the research carried out. We compare the outcomes from the club with the results from the secondary school. We also give an overview of the benefits, as well as the problems that such activities can bring to the classroom.

Kľúčové slová

Code review, programovanie, programátorský krúžok, stredná škola

Keywords

Code review, programming, leisure time activities club, high school

1 ÚVOD

Metodík na vyučovanie programovania existuje veľa. My sme sa v našom výskume sústredili na code review a možnosti jeho využitia ako edukačnej aktivity.

Code review je technika, ktorá sa bežne využíva v programátorskej praxi pri vývoji softvéru. Dá sa charakterizovať ako posudzovanie programového kódu [1] zvyčajne kolegami autora programu, tiež programátormi. Cieľom tejto aktivity je nájsť a odstrániť chyby v programe, prípadne program optimalizovať, a tým zlepšiť jeho kvalitu [2]. Ako ukázali viaceré výskumy, code review môže pomôcť odhaliť až 70 % chýb v softvérových projektoch [3, 4]. Preto sa táto technika radí medzi osvedčené postupy využívané profesionálnymi programátormi. Dobrým príkladom softvérového

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitní a pedagogická giganta, u ktorého sa code review stalo nevyhnutnou súčasťou vývoja softvéru už od začiatku histórie spoločnosti je Google [5].

Pre potreby využitia code review vo vyučovaní treba túto techniku upraviť. Vzniklo tak tzv. edukačné code review, ktorým sa vo svojich výskumoch zaoberá v poslednej dobe veľa profesionálov z oblasti vzdelávania. Pri analyzovaní publikácií z tejto oblasti, sme našli niekoľko výskumov aplikujúcich code review do vyučovania programovania, avšak iba na univerzitnej úrovni [6, 7, 2]. Našou cieľovou skupinou sú však žiaci na nižších stupňoch vzdelávania, preto sa vo svojom výskume zameriavame na nich.

2 NÁŠ VÝSKUM

Od školského roka 2017/2018 využívame code review prispôbené školskému prostrediu na strednej škole, ale i na voľnočasovom krúžku, ktorý navštevujú žiaci rôznych ročníkov základnej a strednej školy. Code review vykonávané v programátorskej praxi, je pomerne zložitý proces, ktorý má svoje pravidlá a štandardy. Naším cieľom pri zavádzaní tejto aktivity do vyučovania nebolo vykonávať ho rovnakým spôsobom, ako je bežné pri vývoji softvérových projektov v programátorských tímoch. Snažíme sa aplikovať niektoré jeho časti do vyučovania tak, aby pomohli rozvíjať u žiakov nové zručnosti, ktoré im môžu pomôcť pri programovaní a nadobúdaní nových kompetencií.

Už v predchádzajúcich publikáciách [8, 9] sme analyzovali výsledky z výskumu, ktorý bol realizovaný na bilingválnej päťročnej strednej škole v prvej fáze v šk. r. 2017/2018 v treťom ročníku, neskôr aj v druhej fáze výskumu v šk. r. 2018/2019 s tými istými žiakmi, už seminaristami v štvrtom ročníku.

V treťom ročníku v prvom polroku sa výskumu zúčastnilo 52 žiakov, v druhom polroku 55 žiakov. Výučba prebiehala podľa plánu vypracovaného podľa Inovovaného štátneho vzdelávacieho programu. Žiaci v druhom ročníku postupne prechádzali z programovacieho jazyka Pascal na Python. V treťom ročníku sa dostali po práci s časovačom a tvorbu vlastnej minihry. Vo štvrtom ročníku sa výskumu zúčastnilo 10 žiakov, seminaristov z informatiky. Pokračovali funkciami, poliami, prácou s reťazcami, textovými súborami a rôznymi algoritmickými výpočtami.

Výskum sme realizovali dvoma spôsobmi: prostredníctvom malých recenzií a recenzií projektu. Pri malých recenziách išlo o recenzovanie krátkych programov pripravených učiteľkou. Žiaci dostali na papieri krátky test, ktorý sa vždy skladal z dvoch úloh. V prvej úlohe mali zistiť, čo by daný program vykonal, keby ho spustili na počítači, popísať to a prípadne nakresliť. V druhej úlohe na nich čakal taktiež krátky program, avšak tento program obsahoval niekoľko chýb. V zadaní úlohy mali napísané, čo by tento program vykonal, ak by v ňom dané chyby neboli. Úlohou žiaka bolo chyby nájsť, označiť ich a správne opraviť. Pri recenziách projektov išlo o recenzovanie dlhších programov, ktoré naprogramovali sami žiaci, a ktoré si komentovali navzájom medzi sebou.

Paralelne sme experimentálne realizovali výskum aj na voľnočasovom programátorskom krúžku v Centre voľného času v Šali. Využili sme tu prvý typ recenzovania kódu – malé recenzie.

Účastníkmi programátorského krúžku bolo päť osôb v rôznom veku a s rôznymi programátorskými skúsenosťami. Krúžok navštevovali: štvrták a ôsmak na ZŠ, ktorí mali predtým skúsenosti iba s programovaním v jazyku Scratch; prvák na SŠ, ktorý súčasne programoval v jazyku Python na hodinách informatiky v škole; a dvaja dospelí, ktorí mali len veľmi základné skúsenosti s programovaním v jazyku Basic z dávnejších stredoškolských čias.

Krúžok sa konal pravidelne raz týždenne a trval 90 minút. Jeho cieľom bolo naučiť účastníkov základy programovania v jazyku Python. Postupne sme sa venovali témam: premenné, základné grafické príkazy (knihovnica tkinter), náhodné čísla, for cyklus, funkcie, udalosti na klikanie myšou a udalosti klávesnice, podmienky, textové reťazce, časovač, posúvanie objektov canvasu. Vďaka

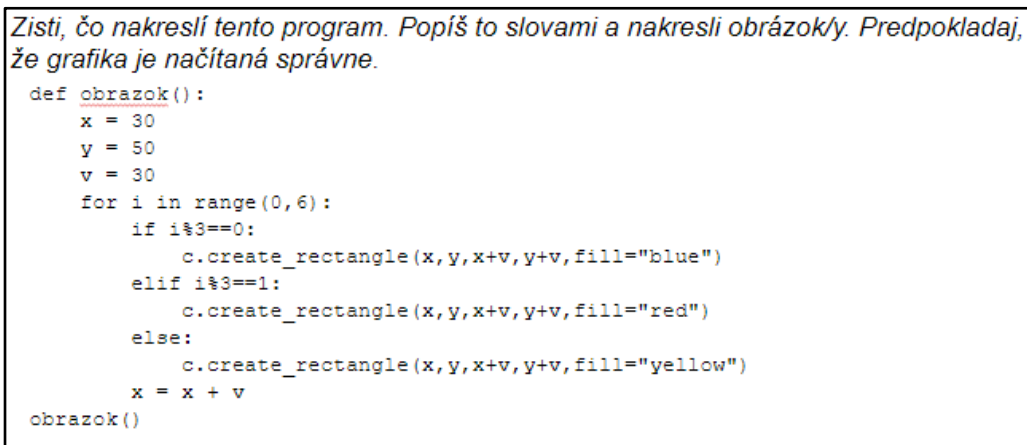
DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitní a pedagogická vyššej časovej dotácii sme na krúžku stihli prejsť aj mnohé ďalšie témy, ktoré sa na strednej škole preberali až na seminári z informatiky.

Ku každej preberanej téme bola vytvorená dvojica úloh a po ukončení témy sme do aktivít krúžku zapojili code review. Účastníci v priebehu 10 minút recenzovali dva zadané programy (5 minút jeden a 5 minút druhý). Rovnako, ako pri malých recenziách na strednej škole, jedna z úloh bola vždy zameraná na zisťovanie, čo daný program robí a druhá na hľadanie chýb. Aj keď bežné vyučovanie na krúžku prebiehalo voľnejšou formou a účastníci mohli spolupracovať a radiť si navzájom, toto neplatilo počas recenzovania. Táto aktivita bola chápaná ako „písomka bez známkovania“ a každý ju teda musel riešiť sám za seba.

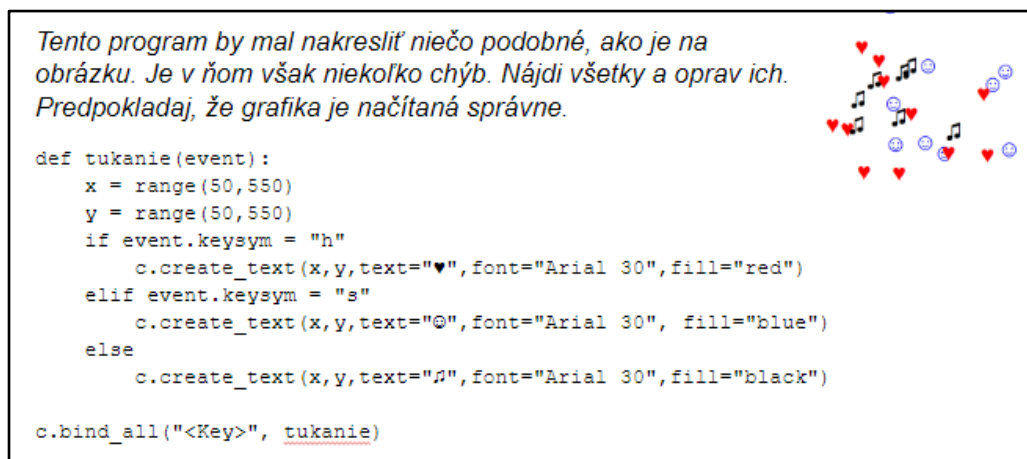
3 VÝSLEDKY Z PROGRAMÁTORSKÉHO KRÚŽKU

Každá malá recenzia sa týkala posledného prebratého učiva. Prvá recenzia obsahovala úlohy zamerané na for cyklus, druhá recenzia bola z funkcií, tretia z podmienok, štvrtá z časovaču a piata z posúvania objektov canvasu.

Pre predstavu, ako takáto malá recenzia vyzerala, prikkladáme ukážku oboch typov úloh na Obr. 1 a Obr. 2.



Obrázok 1: Úloha 1



Obrázok 2: Úloha 2

Za každú recenziiu bolo možné získať maximálne 5 bodov. Za prvú úlohu, v ktorej mali napísať, čo program urobí, mohli získať 2 body, v druhej úlohe, kde mali nájsť a opraviť chyby, mohli získať 3 body.

Výsledky z malých recenzií na programátorskom krúžku, sú uvedené v tab. 1. Väčšinu recenzií vypracovali len štyria účastníci z piatich, keďže jeden z nich sa na viacerých stretnutiach krúžku nezúčastnil.

Tabuľka 1: Výsledky malých recenzií z krúžku

Pohlavie	Označenie	1.recenzia	2.recenzia	3.recenzia	4.recenzia	5.recenzia
muž	štvrták	0	5	4	3	4
muž	dospelý	0	5	5	5	5
muž	ôsmak	0	4	5	4	5
žena	dospelá	0	3	1	5	4
muž	stredoškólák	chýbal	chýbal	4	chýbal	chýbal

Za prvú recenziu nezískal nikto z účastníkov krúžku žiadny bod. Nikto nedokázal zistiť, čo je výsledkom zbehnutia prvého programu, ani nájsť a opraviť chyby v druhom programe. Problém bol pravdepodobne v tom, že išlo o úplne nové typy úloh, s akými sa dovtedy nestretli.

Situácia sa výrazne zmenila pri druhej recenzii, keď dvaja účastníci získali dokonca plný počet bodov a zvyšní dvaja štyri a tri body. Veľmi podobné výsledky dosahovali účastníci aj vo všetkých ostatných recenziách. Najlepšie dopadlo posledné recenzovanie, kde dvaja účastníci získali plný počet bodov a zvyšní dvaja stratili len po jednom bode.

Postoj účastníkov krúžku k tejto aktivite sme zisťovali iba z pozorovania na krúžku a z osobného rozhovoru s nimi. Ich reakcie v rozhovore boli pozitívne. Pri pozorovaní ich práce sme si všimli, že často vypracovali oba príklady za kratší čas, než mali k dispozícii.

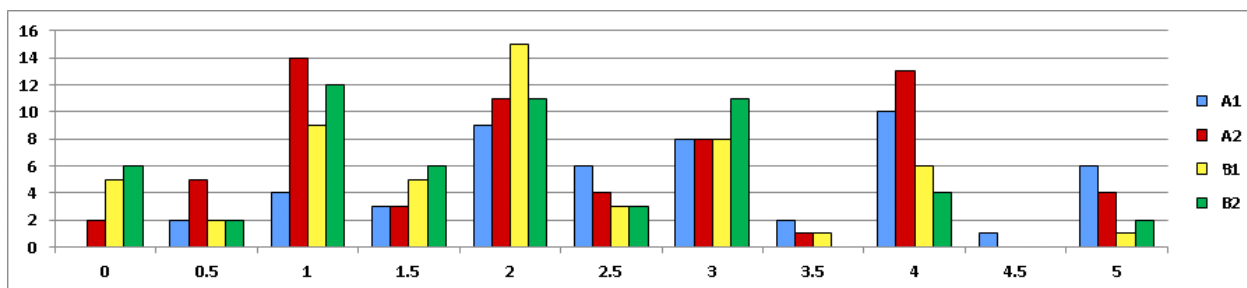
4 VÝSLEDKY V POROVNANÍ SO STREDNOU ŠKOLOU

V našom predchádzajúcom výskume na strednej škole tvorili našu výskumnú vzorku v prvej fáze žiaci dvoch tried, ktorí boli rozdelení do štyroch skupín (A1, A2, B1, B2). Keďže chceme porovnávať výsledky využitia code review na krúžku a na strednej škole, uvedieme tu len výsledky stredoškóľakov z malých recenzií. Tieto prebiehali na SŠ rovnako ako neskôr na krúžku – v piatich kolách. Programy, ktoré žiaci recenzovali, boli tiež rovnaké. Najlepšie výsledky v tejto aktivite dosiahla skupina A1, v ktorej sa nevyskytla ani jedna recenzia s nulovým hodnotením a tri žiačky získali dvakrát po plnom počte bodov. Druhou v poradí bola skupina A2. Jedna žiačka z tejto skupiny získala plný počet bodov z dvoch malých recenzií a dve ďalšie žiačky z jednej recenzie. Iba jednému žiakovi bolo udelené nulové hodnotenie, a to z dvoch recenzií.

Trieda B bola menej úspešná ako trieda A. V oboch jej skupinách sa vyskytli 4 žiaci s nulovým počtom bodov za jednu malú recenziu a v skupine B2 mala ešte ďalšia žiačka nulové hodnotenie z dvoch recenzií. Iba jeden (B1) a dvaja (B2) žiaci získali plný počet bodov aj to len z jednej recenzie.

Na Obr. 3 uvádzame hrubé skóre účastníkov prvej fázy výskumu podľa tried a dosiahnutých počtov bodov od 0 po 5, pričom sa hodnotilo aj pol-bodmi.

Pri poslednej recenzii sa žiaci priznali, že aj keď boli tieto „malé písomky“ pre nich ťažké, pomohli im naučiť sa hľadať chyby v programe a rozmýšľať nad programom inak. Na hodine informatiky sa doteraz stretávali iba s prístupom: „mám zadanie, naprogramujem program“. Pri recenzovaní sa dostali na opačnú stranu: „mám program/kód, musím zistiť, aké bolo zadanie/čo program robí“.



Obrázok 3: Hrubé skóre žiakov strednej školy v prvej fáze výskumu podľa skupín

Nasledujúci školský rok sme sa snažili vykonať overenie so žiakmi štvrtého ročníka, ktorí absolvovali malé recenzie v treťom ročníku. Dostali identické malé recenzie, ako predchádzajúci rok. Výsledky však nepreukázali žiadne výrazné skutočnosti, ktoré by hovorili o tom, či sa žiaci zlepšili v hľadaní chýb a zisťovaní, čo projekt vykoná. Tabuľky z tretieho a štvrtého ročníka pre objemnosť dát neuvádzame, možno ich nájsť v diplomovej práci jednej z autoriek [10].

Z analýzy výsledkov malých recenzií v treťom aj štvrtom ročníku vyplýva, že prvý príklad, v ktorom mali žiaci zistiť, čo program vykoná, dopadol vo všeobecnosti lepšie, než druhý. Toto zistenie nám štvrtáci potvrdili aj v odpovediach na otázky v dotazníku, ktorý vyplnili po skončení všetkých aktivít súvisiacich s code review.

Ich názor na využívanie takýchto typov aktivít vo vyučovaní programovania sme v dotazníku zisťovali aj prostredníctvom otázky, či by zaradili takéto typy príkladov do vyučovania programovania aj v iných ročníkoch, prípadne školách. Tri štvrtiny žiakov odpovedali kladne so zdôvodnením, že je to dobrý tréning pri učení sa programovať. Zvyšní žiaci sa nevedeli k otázke vyjadriť.

Porovnávať výsledky získané v školskom prostredí s výsledkami z prostredia záujmového útvaru asi nie je celkom priamočiare. Programátorský krúžok navštevujú najmä ľudia, ktorí sa o programovanie naozaj zaujímajú, a preto vykazujú aj lepšie výsledky. Na druhej strane, informatiku a programovanie v škole, žiaci absolvujú povinne, bez ohľadu na to, aký majú k týmto predmetom vzťah. Okrem toho, väčšina škôl má na informatiku k dispozícii jednu vyučovaciu hodinu. Nové učivo je často prezentované na jednej-dvoch hodinách. Každé stretnutie krúžku trvalo 90 minút, pričom každej novej téme sa účastníci venovali minimálne jedno-dve krúžkové stretnutia. Za tento čas teda členovia krúžku preriešili omnoho viac príkladov, než účastníci nášho výskumu na strednej škole. Ďalší rozdiel je aj vo veľkosti skupiny, ktorej sa učiteľ (v CVČ vychovávateľ) venuje. Krúžok zvyčajne navštevuje omnoho menší počet účastníkov, ako je počet žiakov v bežnej triede, čo umožňuje realizovať metódu individuálneho prístupu jednoduchšie, než učiteľovi v škole.

Výsledky experimentálneho výskumu na programátorskom krúžku nielen potvrdili predpokladané zistenie, že účastníci krúžku, ktorí sa zaujímajú o programovanie a chcú sa mu venovať, sú ochotní vynaložiť viac úsilia i času na to, aby sa naučili programovať, ale ukázali aj to, že takíto jedinci pomerne rýchlo dokážu prijať aj menej tradičné edukačné aktivity a vykazovať v nich veľmi dobré výsledky. V našom prípade už po prvej skúsenosti s code review pochopili, čo sa od nich očakáva a svoju zručnosť v recenzovaní zlepšovali tak pri zisťovaní, čo daný program robí, ako aj pri odhaľovaní a opravovaní chýb.

5 PRÍNOS A PROBLÉMY CODE REVIEW

Implementácia code review do vyučovania nie je úplne bezproblémová a prináša so sebou menšie komplikácie. V školskom prostredí predstavuje problém najmä časová náročnosť. Realizácia code review vo forme malých recenzií si vyžaduje pravidelne vyčlenenú časť vyučovacej hodiny, čo nemusí byť jednoduché, keďže učiteľ má podľa ŠVP prebrať pomerne veľa učiva, pričom časová dotácia informatiky v škole je 45 minút a výučba prebieha len raz týždenne. Okrem toho, keďže

ide o typ aktivity, s ktorým sa žiaci bežne v škole nestretávajú, je potrebné im na začiatku podrobne vysvetliť, čo sa od nich očakáva, ukázať im vzorové riešenia podobných zadání, prípadne ich v istom zmysle „natrénovať“, na úlohu recenzenta. Všetky tieto aktivity si opäť vyžadujú čas navyše. Sú však naozaj nutné v prípade recenzovania väčších programov, napríklad projektov, ale určite veľmi nápomocné aj v prípade malých recenzií.

Pokiaľ uvažujeme o zapojení aktivít využívajúcich code review na krúžku, problémy s časovými obmedzeniami odpadajú, keďže krúžok má jednak väčšiu týždennú časovú dotáciu a aj voľnejší plán aktivít.

Recenzovanie programov taktiež prináša do vyučovania programovania isté benefity. Učiteľ takouto formou získa širší pohľad na žiaka, zistí, ako je na tom s programovaním, ale zároveň aj ako dokáže čítať cudzí kód, či ho dokáže pochopiť, zistiť, čo robí, nájsť v ňom chyby a pomôcť ich opraviť. Žiak môže získať alternatívny pohľad na riešenie častokrát aj rovnakých problémov, aké riešil už aj on sám, čím sa v podstate len iným spôsobom učí programovať, učí sa porozumieť cudziemu programu, správne popísať chyby a trénuje si vyjadrovacie schopnosti. Tieto i ďalšie výhody využitia code review pri vyučovaní programovania, či už v škole alebo na krúžku, sa ešte umocnia, ak sa takýto typ aktivity využije pri recenzovaní dlhších programových kódov, najmä ak ich autormi sú sami žiaci (viď napr. recenzovanie projektov prezentované v jednej z našich predošlých publikácií [9]).

6 ZÁVER

Náš výskum v súčasnosti pokračuje ďalej v Centre voľného času na programátorských krúžkoch. Niekoľko účastníkov z minulého školského roka pokračuje v programovaní v jazyku Python. Taktiež vznikla nová skupina začiatocníkov, ktorí sa učia programovať v jazyku Python.

S minulými aj novými členmi krúžku pokračujeme v realizácii aktivít code review. Programátorská prax, výskumy zahraničných univerzít, ale aj naše menšie výskumy poukazujú na to, že tieto aktivity postupne smerujú k zlepšeniu programátorských zručností a aj u začiatocníkov v programovaní prispievajú k zlepšeniu kvality nimi vytváraného programového kódu. Preto si i napriek problémom s časovou dotáciou predmetu myslíme, že by učitelia mali podobné aktivity zapracovávať aj do vyučovania programovania, aby si na ne žiaci postupne zvykli a mohli čím skôr využívať benefity, ktoré im tieto aktivity ponúkajú.

Pri zapájaní takýchto aktivít môže učiteľovi pomôcť, ak si počas hodiny všíma, aké chyby robia žiaci pri programovaní, s akými chybovými hláškami sa často stretávajú a pod. Na základe svojich pozorovaní potom môže zakomponovať nové prvky do vyučovania programovania, napríklad aj prostredníctvom code review. Taktiež môže takouto formou zistiť, či žiak dokáže nad programom rozmyšľať, pochopiť ho a abstraktne si vizualizovať, čo program vykoná a následne to popísať.

Vhodné je, ak sa takéto typy aktivít vykonávajú so žiakmi pravidelne a najmä dlhodobo. Aj toto im môže pomôcť hlbšie vniknúť do programovania a budovať si hneď od začiatku správne programátorské návyky.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] BOEHM, B. Software Engineering Economics. (Englewood Cliffs 1981). General Electric PRICE Systems: PRICE-Software Model Reference Manual, 1989.
- [2] KUBINCOVÁ, Z., HOMOLA, M. Code Review in Computer Science Courses: Take One. In: International Conference on Web-Based Learning. Springer, Cham, 2017. p. 125–135.
- [3] UWANO, H., et al. Analyzing individual performance of source code review using reviewers' eye movement. In: Proceedings of the 2006 symposium on Eye tracking research & applications. ACM, 2006. p. 133–140.
- [4] WIEGERS., K. E. Peer reviews in software: A practical guide. Addison-Wesley Boston, 2002.

- [5] SADOWSKI, C, et al. Modern code review: a case study at google. In: Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice. ACM, 2018. p. 181–190. Dostupné na: <https://sback.it/publications/icse2018seip.pdf>
- [6] TRYTTEN, D. A. A design for team peer code review. In: ACM SIGCSE Bulletin. ACM, 2005. p. 455–459.
- [7] TANG, M. Caesar: a social code review tool for programming education. 2011. PhD Thesis. Massachusetts Institute of Technology.
- [8] CSICSOLOVÁ, I., KUBINCOVÁ, Z. Code review v programovaní na strednej škole – predbežné výsledky. Sborník konference Didinfo, Liberec, 2018. ISBN 978-80-7494-424-6, ISSN 2454-051X. Dostupné na: http://www.didinfo.net/images/DidInfo/files/Didinfo_2018.pdf.
- [9] CSICSOLOVÁ, I., KUBINCOVÁ, Z. Nájdeš chybu v mojom programe? Zborník konferencie Didinfo, Banská Bystrica, 2019. ISBN 978-80-557-1533-9, ISSN: 2454-051X. Dostupné na: https://io.fpv.umb.sk/didinfo/Zbornik_Didinfo_2019.pdf.
- [10] CSICSOLOVÁ, I. Code review v informatike na strednej škole, Diplomová práca, FMFI UK, Bratislava, 2019.

Mikrosvet na výučbu programovania priateľský k nevidiacim študentom sekundárneho vzdelávania

Microworld for Learning Programming Friendly to Secondary Blind Students

Lucia Budinská, Ľudmila Jašková,
Michal Kováč, Karolína Mayerová
KDMFI FMFI UK Bratislava
Mlynská dolina
84243 Bratislava
Slovenská republika

lucia.budinska@uniba.sk, ludmila.jaskova@uniba.sk,
kovac254@uniba.sk, karolina.mayerova@uniba.sk

Mária Karasová
SŠI pre žiakov so zrakovým
postihnutím
Svrčia 6
842 11 Bratislava
Slovenská republika
maria.karasova@bee.sk

ABSTRAKT

V príspevku popisujeme časť nášho výskumu zameraného na rozvoj algoritmického myslenia a programátorských zručností nevidiacich žiakov nižšieho sekundárneho vzdelávania. Predstavíme nami vyvinuté programovacie prostredie Coshi, ktoré umožňuje nevidiacim programovať pohyb virtuálneho robota po obrazovke počítača. Okrem toho sme vytvorili aj edukačný scenár zameraný na postupné zoznamovanie sa s prostredím, príkazmi jazyka a základnými programátorskými konceptami. Popri overovaní prostredia a edukačného scenára sme skúmali, ako si žiaci konštruujú poznanie zložitejších programátorských konceptov, akými sú cyklus s podmienkou, vnorené cykly a do akej miery ich vedú použiť pri riešení komplexnejšieho problému.

ABSTRACT

In this paper we present a part of our research aimed at development of algorithmic thinking and programming skills of blind students attending lower secondary schools. We have introduced a programming environment Coshi for blind students where they can control a virtual robot in a square grid with assigned sounds. During verification of this environment, an educational scenario was created for six lessons. The aim of these lessons was to make students gradually acquainted with the programming environment and basic programming concepts (sequence of commands, counting loop, variable, conditional branching, conditional loop). Throughout the testing, we also examined how students construct their knowledge of programming concepts like conditional loop and nested loop and to what extent they can use them to solve a more complex problem.

Kľúčové slová

Programovanie, nevidiaci žiaci, nižšie sekundárne vzdelávanie, abstrakcia, prípadová štúdia, programátorské koncepty.

Keywords

Programming, blind students, lower secondary education, abstraction, case study research, programming concepts.

1 ÚVOD

V ostatných rokoch badať v mnohých krajinách narastajúce úsilie zakomponovať vo vhodnej miere a forme výučbu informatiky na všetkých úrovniach vzdelávania. Hromkovič et al. [1] uvádzajú, že „trvalo udržateľné vzdelávanie v oblasti informatiky by malo byť postavené na algoritmickom myslení, ako jeho primárnom ciele, z čoho by profitovalo široké a všeobecné vzdelávanie“.

Sme presvedčení, že rozvoj algoritmického myslenia je dôležitý aj pre nevidiacich žiakov a študentov. Osoby so zrakovým postihnutím majú kvôli svojmu postihnutiu obmedzené možnosti nájsť si vhodné zamestnanie. Tieto možnosti sa podstatne zväčšia, ak vedú používať digitálne technológie. Veľkou výhodou je, ak okrem používania rôznych softvérových aplikácií, vedú aj programovať. Osobne poznáme niekoľko nevidiacich programátorov, ktorí úspešne pracujú v počítačových firmách. Uvedomujeme si, že čím skôr sa u nevidiacich žiakov vzbudí záujem o túto profesiu, tým ľahšie sa rozhodnú pre štúdium informatiky.

Podľa Paperta [2] „*sa študenti učia efektívnejšie, ak môžu riešiť vlastné úlohy, čím si budujú svoje znalosti sami a vyjadrujú myšlienky cez médium, ktoré umožňuje získať priame skúsenosti*“. V súčasnosti najpopulárnejšie programovacie jazyky pre žiakov nižšieho sekundárneho vzdelávania – Scratch a Logo – umožňujú, aby sa žiaci učili programovať konštrukcionistickým spôsobom v zmysle Paperta. Tieto prostredia však nie sú použiteľné pre nevidiacich (kapitola 2).



Obrázok 1: Virtuálny robot (čierny kruh) sa pohybuje po políčkach, ktoré majú priradené rôzne zvuky

Vyvinuli sme preto vlastné programovacie prostredie Coshi (obrázok 1), ktoré umožňuje nevidiacim žiakom programovať pohyb robota po ozvučenej štvorcovej sieti (viac v kapitole 3). Vytvorili sme aj edukačný scenár na šesť vyučovacích hodín (kapitola 4.2). Cieľom bolo zoznámiť študentov s programovacím prostredím a základnými programovacími konceptami. Okrem toho nás zaujímalo, ako si študenti konštruujú poznanie zložitejších programátorských konceptov, akými sú cyklus s podmienkou, vnorené cykly a do akej miery ich vedú použiť pri riešení komplexnejšieho problému (kapitola 5).

2 PREHĽAD PROBLEMATIKY

Výučba tematickej oblasti **Algoritmické riešenie problémov** v bežnej triede s intaktnými žiakmi (bez akéhokoľvek postihnutia) sa obyčajne realizuje pomocou softvéru bohato využívajúceho vizuálnu reprezentáciu – jazyky typu *Logo* [3] a *Scratch* [4]. Tieto prostredia sa však nedajú použiť s nevidiacimi žiakmi, ktorí pracujú s počítačom pomocou čítača obrazovky. Tento im dokáže sprostredkovať iba textové a zvukové informácie. Okrem toho nevidiaci žiaci nepoužívajú myš a všetok vstup zadávajú len pomocou klávesnice. Spomínané prostredia neumožňujú všetky akcie vykonávať pomocou myši, aj pomocou klávesnice.

Blokové (resp. kartičkové) jazyky typu Scratch sú veľmi populárne, pretože odstraňujú problémy súvisiace so zložitou syntaxou a sú vhodnou voľbou pre začiatočníkov. Milne et al. [5] popísal výzvy súvisiace s prístupnosťou blokových jazykov a vyvinul prototyp prostredia *Blocks4All* ako inkluzívnejšiu alternatívu fungujúcu v prostredí Apple iPad. Bohužiaľ, iPad sa na slovenských školách ešte stále nevyužíva kvôli vysokej cene.

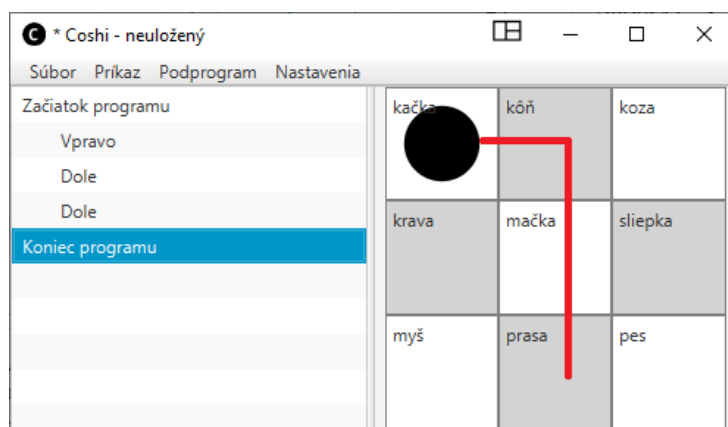
Prostredia *SonicPi* [6] a *MusicBlocks* [7] umožňujú programovanie hudby, takže by mohli byť vhodnou alternatívou pre nevidiacich. Prostredia však nie sú prístupné pre čítač obrazovky. Prototyp zvukového programovacieho jazyka špeciálne určeného pre nevidiacich vyvinula skupina

programátorov v Chile [8]. Pilotné testovanie s nevidiacimi programátorskými nováčikmi vo veku 17 až 20 rokov, ktoré autori realizovali, ukázalo, že aj pre nich bolo programovanie v tomto jazyku príliš abstraktné [9]. Tieto prostredia však boli pre nás inšpiráciou pri tvorbe nášho zjednodušeného textového zvukového programovacieho jazyka *Alan* [10] pre žiakov vo veku 11 až 15 rokov. Vhodným riešením pre nevidiacich žiakov našej vekovej kategórie je aj fyzický programovací jazyk *Torino* [11], ktorý vyvinuli v Microsoft Research Lab Cambridge. Obidve uvedené prostredia sme úspešne overili s našimi nevidiacimi žiakmi [12]. Nováčikovia v programovaní v nich môžu programovať jednoduché melódie, príbehy alebo riekanky. Ukázalo sa, že takéto kreatívne aktivity sú pre nevidiacich žiakov mimoriadne pútavé. Žiaci si môžu osvojiť všetky základné programovacie koncepty bez toho, aby boli nútení si predstavovať pohyb nejakého objektu po obrazovke, čo je náročné najmä pre kongenitálne (od narodenia) nevidiacich žiakov.

Pre povolanie programátora je však dôležité, aby sa u žiakov pestovala aj schopnosť orientácie na obrazovke počítača. Preto sme pre mierne pokročilých žiakov vyvinuli prostredie Coshi [13], ktoré popíšeme v nasledujúcej kapitole.

3 PROGRAMOVACIE PROSTREDIE COSHI

Prostredie Coshi umožňuje programovať pohyb robota po ozvučenej štvorcovej sieti na obrazovke počítača. Používa jednoduchý programovací jazyk, ktorého základ tvoria príkazy *Hore*, *Dole*, *Vľavo*, *Vpravo*. Tieto príkazy slúžia na presun robota o jedno políčko v príslušnom smere. Možno použiť aj príkaz *Prehraj* na prehranie zvuku priradeného k políčku, na ktorom sa robot aktuálne nachádza. Príkaz *Ticho* zapne tichý režim, teda robot sa bude pohybovať v tichosti a príkaz *Nahlas* opätovne zapne prehrávanie zvukov pri pohybe robota po políčkach. Okrem základných príkazov je možné použiť cyklus *Opakuj* so známym počtom opakovaní (podobne ako *FOR*), cyklus s podmienkou *Kým* (podobne ako *WHILE*), príkaz vetvenia *Ak* (*IF-THEN-ELSE*), podprogram bez parametra a s jedným parametrom a číselnú premennú.



Obrázok 2: Aplikácia Coshi, políčka majú priradené zvuky zvierat

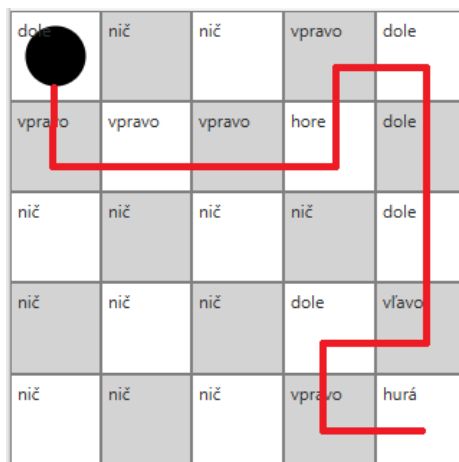
Prostredie aplikácie je plne ozvučené. Čítač obrazovky interpretuje všetky zamerané položky aplikačnej ponuky, ako aj vytváraný program. Jednotlivé políčka hracej plochy majú priradené zvuky. Každý zvuk sa prehrá, keď na políčko pri vykonávaní programu vstúpi robot. Spätnou väzbou počas vykonávania programu je pre nevidiaceho žiaka sekvencia zvukov, ktorá sa prehrá pri pohybe robota po hracej ploche. Názvy priradených zvukov sa dajú zobrazit' na políčkach, aby aj vidiaci žiaci a učitelia mali vizuálnu spätnú väzbu. Skôr ako začnú nevidiaci žiaci vytvárať program, môžu hraciu plochu preskúmať pomocou klávesov so šípkami a zistiť, aké zvuky sú priradené k jednotlivým políčkam. Pre každé políčko si môžu nechať vysloviť aj jeho súradnice (číslo riadku a stĺpca). Editor programu bol vytvorený s ohľadom na nevidiacich programátorských začiatníkov, ktorí nemusia písať celé príkazy, ale môžu ich vyberať z ponuky alebo vkladať do programu pomocou klávesových príkazov.

Na obrázku 2 je snímok okna z programu Coshi s hracou plochou o veľkosti 3x3 políčka. Jednotlivé políčka majú priradené zvuky zvierat. V ľavej časti okna je program. Pri jeho vykonávaní sa robot znázornený čiernym kruhom postupne presunie po políčkach, ktoré majú priradené zvuky koňa, mačky a prasťa. Pri pohybe robota sa zvuky prehrávajú v uvedenom poradí.

Prostredie štandardne ponúka niekoľko zvukových schém, ale k dispozícii je aj editor nových zvukových schém. K jednotlivým políčkam tak môže učiteľ priradiť rôzne iné zvuky podľa motivačných preferencií žiakov. Samotní žiaci sa môžu podieľať na tvorbe vlastných mikrosvetov a rozhodovať o tom, aké zvuky sa priradia k políčkam a aké typy úloh budú riešiť. Prostredie teda umožňuje vytvárať mikrosvet, v ktorých žiaci riešia úlohy zamerané na dosiahnutie rôznych vzdelávacích cieľov.

Najdôležitejším cieľom je samozrejme **rozvoj algoritmického myslenia** a zoznamovanie sa so základnými programovacími konceptami (ako je postupnosť príkazov, cyklus, príkaz vetvenia, podprogramy a premenné). S tým súvisí aj rozvoj schopnosti odladovať program – hľadať chyby v programe a odstraňovať ich. Okrem algoritmických a programátorských zručností môžu žiaci získavať aj mnohé ďalšie, ako napríklad nasledujúce.

- **Rozvoj hudobnej tvorivosti:** Ak jednotlivým políčkam priradíme tóny stupnice, žiaci môžu pomocou programovania pohybu robota po týchto políčkach naprogramovať rôzne melódie.
- **Rozvoj orientačných schopností** potrebných pre prácu s tabuľkami alebo mriežkami. Ak si napríklad jednotlivé políčka predstavíme ako domy a priradíme im zvuky vyslovujúce adresy domov (riadky budú názvy ulíc a stĺpce budú čísla domov), žiaci môžu programovať pohyb robota – poštára doručujúceho zásielku z jedného domu do iného.
- **Rozvoj logického myslenia a schopnosti riešiť problémy.** Ak napríklad, jednotlivým políčkam priradíme zvuk vyslovujúci smer pohybu robota, žiaci môžu programovať pohyb robota podľa trasy určenej priradenými zvukmi (obrázok 3).



Obrázok 3: Hracia plocha použitá v úlohe Programovanie trasy (kapitola 5)

4 METODOLÓGIA

4.1 Ciele, metódy a výskumná vzorka

V našom výskume sme sledovali, ako si nevidiaci žiaci konštruujú znalosť programátorských konceptov. V tomto článku sa pokúsime nájsť odpovede na nasledujúce výskumné otázky.

O1: Ako nevidiaci žiaci postupujú pri riešení úloh v prostredí Coshi, ak pracujú podľa nášho edukačného scenára (kapitola 4.2)?

O2: Ako nevidiaci žiaci používajú podmienený cyklus a cyklus so známym počtom opakovaní pri riešení komplexnejších úloh v prostredí Coshi?

Použili sme kvalitatívnu stratégiu prípadovej štúdie. Skúmanými prípadmi boli úlohy, ktoré nevidiaci žiaci riešili. Na zber a analýzu dát sme použili nasledujúce výskumné metódy [14], [15].

- Sledovanie žiakov pri riešení úloh počas hodín informatiky.
- Polo-štruktúrované rozhovory so žiakmi a ich učiteľom informatiky.
- Analýza našich poznámok vytvorených počas hodín informatiky.

Našu výskumnú vzorku tvorili štyria zrakovo postihnutí žiaci ôsmeho ročníka základnej školy vo veku 14–16 rokov (dvaja chlapci a dve dievčatá). Boli to žiaci z tej istej triedy, dobre sa poznali a boli zvyknutí spolupracovať v skupine. Pri práci s počítačom používali slúchadlá a čítač obrazovky. Jedna žiačka so zvyškami zraku používala aj zväčšovací softvér. Podľa vyjadrenia učiteľa informatiky boli ich počítačové zručnosti mierne nadpriemerné. Všetci žiaci vedeli používať základné funkcie textového, tabuľkového a zvukového editora, emailového klienta a webového prehliadača. Okrem toho už v minulosti programovali v prostrediach Alan a Torino, v ktorých vytvárali programy zložené z postupností príkazov a cyklov so známym počtom opakovaní. Žiaci tiež mali skúsenosti s riešením úloh mimo počítača zameraných na programovanie pohybu objektu po mriežke. S konceptom podmieneného cyklu sa žiaci predtým nestretli.

4.2 Edukačný scenár

Vytvorili sme edukačný scenár na šesť vyučovacích hodín. Cieľom **prvej hodiny** bolo zoznámiť žiakov s programovacím prostredím Coshi a základnými príkazmi jazyka. Použili sme zvukovú schému *Zvieratá* (obrázok 3). Porozprávali sme žiakom príbeh o robotovi menom Coshi, ktorý pomáha kŕmiť zvieratá na farme. Vyzvali sme ich, aby pomocou základných príkazov (*vlavo*, *vpravo*, *hore* a *dole*) programovali pohyb robota medzi zvieratami. Počas **druhej hodiny** žiaci riešili úlohy zamerané na použitie cyklu so známym počtom opakovaní. Mali vytvárať programy, aby sa robot pohyboval o presne určený počet políček v rovnakom smere. **Tretia hodina** bola zameraná na použitie podmieneného cyklu. Žiaci mali naprogramovať, aby robot prešiel na koniec riadka alebo stĺpca bez ohľadu na rozmery hracej plochy. Počas **štvrtej hodiny** žiaci vytvárali programy, v ktorých používali premennú na to, aby robot zistil počet určitých objektov (napríklad psov) na políčkach, cez ktoré prejde (obrázok 4).

nič	pes	nič
nič	nič	pes
pes	nič	nič

Obrázok 4: Zvuková schéma pre úlohu Počítanie psov

Piata hodina bola zameraná na použitie príkazu vetvenia. Úlohou bolo naprogramovať robota, aby spočítal susedné políčka alebo aby zistil, či sa môže pohnúť určitým smerom (nie je tam okraj hracej plochy). Počas **šiestej hodiny** žiaci riešili komplexnejšie úlohy zamerané na použitie vnorených programovacích štruktúr.

5 POZOROVANIA

Pozorovali sme, že žiaci v prostredí Coshi pracovali aktívne a s nadšením. Pri editovaní programu intenzívne používali klávesové príkazy, vďaka čomu nemali väčšie problémy.

Vytváranie programov pozostávajúcich z postupností základných príkazov bez cyklov nerobilo žiakom problémy (hodina 1). Keď mali žiaci vytvoriť program, v ktorom sa nejaký príkaz niekoľkokrát opakoval (hodina 2), neprišli na to, že by mohli použiť cyklus *Opakuj* so známym počtom opakovaní, hoci tento pojem už použili pri programovaní v jazykoch *Alan* a *Torino*. Cyklus začali žiaci používať, až keď príkaz *Opakuj* náhodou našla jedna žiačka v zozname príkazov. Žiaci si

spomenuli, že ním môžu nahradiť sekvenciu opakujúcich sa príkazov a potom už používali túto konštrukciu bez problémov.

Problémy sme pozorovali, až keď mali žiaci riešiť úlohy zamerané na použitie podmieneného cyklu *Kým* (hodina 3). Žiaci mali naprogramovať, aby robot prišiel na koniec riadku a zastavil sa. Rozhodli sa spočítať políčka, po ktorých má robot prejsť a použiť cyklus *Opakuj* so známym počtom opakovaní. Opýtali sme sa žiakov, či by dokázali program zmeniť tak, aby fungoval na hracej ploche ľubovoľnej veľkosti. Žiaci začali v ponuke príkazov hľadať vhodný príkaz, ktorý by mohli použiť a všimli si príkaz *Kým*. Vytvorili program uvedený na obrázku 5 vľavo.

Nekonečný cyklus	Správne riešenie
kým je voľné vpravo koniec kým	kým je voľné vpravo vpravo koniec kým

Obrázok 5: Programy žiakov obsahujúce podmienený cyklus

Keď žiaci program spustili, nič sa neudialo, ale program očividne bežal a museli ho ukončiť špeciálnym klávesovým príkazom. Vysvetlili sme žiakom, že vytvorili nekonečný cyklus, pretože jeho telo je prázdne, robot sa nehýbe, takže nikdy sa nedostane na koniec riadku a podmienka cyklu bude stále splnená. Žiaci pridali do tela cyklu príkaz *vpravo* (obrázok 5 vpravo). Počas následnej diskusie so žiakmi sme zistili, že slovo *vpravo* v podmienke pochopili tak, že sa má robot hýbať vpravo. Ďalšie principiálne podobné úlohy už žiaci riešili bez problémov.

Počas štvrtej hodiny, zameranej na použitie premenných, sme nepozorovali žiadne problémy. Žiaci sa naučili definovať a nastaviť premennú, zobrazit' jej hodnotu a meniť ju, keď sa niečo udeje – robot sa presunie, ozve sa nejaký zvuk (brechot psa) a podobne.

Príkaz s podmienkou (hodina 5) bol pre žiakov novým pojmom. V prostredí Coshi je tento príkaz implementovaný iba s dvomi vetvami – *Ak* vetvou a *else* vetvou, aj v prípade, keď tá druhá vetva nie je potrebná a teda je prázdna. Takáto implementácia sa ukázala ako problematická, pretože žiaci strácali v tejto komplikovanej štruktúre orientáciu.

Veľkou výzvou bola pre žiakov úloha **Programovanie trasy robota** (hodina 6). V úlohe sme použili zvukovú schému, kde mali jednotlivé políčka priradené vyslovované príkazy určujúce smer pohybu robota. Takto bola jednoznačne určená trasa pre robota z ľavého horného rohu hracej plochy do cieľového políčka so zvukom **hurá** (obrázok 3). Žiaci najprv vyriešili úlohu v priamom režime tak, že podľa priradených príkazov pohybovali robotom klávesmi so šípkami. Keď zistili, že priradené príkazy naozaj „dovedú“ robota do cieľa, začali vytvárať program.

Obrázok 6 obsahuje tri rôzne riešenia, ktoré žiaci postupne vytvorili. Najprv použili iba základné príkazy (obrázok 6, prvý stĺpec). Keď sme im pripomenuli, že môžu používať aj príkaz cyklu, nahradili postupnosti rovnakých príkazov cyklami (obrázok 6, druhý stĺpec). Niektorí študenti použili cyklus *Opakuj* so známym počtom opakovaní a niektorí podmienený cyklus *Kým*. Opýtali sme sa žiakov, či by dokázali vytvoriť taký program, ktorý by fungoval aj keby sa trasa pre robota zmenila. Žiaci najprv nevedeli, ako by mal taký program vyzeráť. Požiadali sme jedného z nich, aby sa správal ako robot Coshi a zvyšní žiaci mu dávali príkazy. Opýtali sme sa žiakov nasledujúce otázky: Ako Coshi zistí, že ešte nie je v cieľi? Čo musí Coshi ešte zistiť, ak vie, že ešte nie je v cieľi? Čo má Coshi spraviť, keď zistí aký príkaz je priradený k aktuálnemu políčku? Žiaci postupne popisali slovne, ako bude program fungovať. Potom napísali program v jazyku, ktorému Coshi rozumie (obrázok 6, tretí stĺpec). Pôvodne sme si mysleli, že žiaci namiesto vnorených príkazov *kým*, použijú príkazy vetvenia *ak – tak – inak*. Žiaci však uprednostnili podmienený cyklus. Pravdepodobne to bolo kvôli komplikovanej implementácii príkazu vetvenia v jazyku Coshi.

Zistili sme, že žiaci počas riešenia úlohy prešli nasledujúcimi štyrmi fázami:

1. Riešenie úlohy v priamom režime (hýbanie robotom pomocou klávesov so šípkami).
2. Tvorba riešenia pomocou základných príkazov pre špecifický prípad.
3. Nahradenie postupností opakujúcich sa príkazov cyklami.
4. Riešenie problému vo všeobecnosti pomocou vnorených cyklov.

Naše pozorovania prediskutujeme v nasledujúcej kapitole.

Bez cyklu	Riešenie špecifického prípadu pomocou sekvencie cyklov	Riešenie vo všeobecnosti s vnorenými cyklami
dole	kým je zvuk dole	kým nie je zvuk yes
dole	dole	kým je zvuk vpravo
vpravo	koniec kým	vpravo
vpravo	kým je zvuk vpravo	koniec kým
vpravo	vpravo	kým je zvuk dole
hore	koniec kým	dole
hore	...	koniec kým
vpravo		kým je zvuk vIavo
dole		vIavo
dole		koniec kým
dole		kým je zvuk hore
dole		hore
...		koniec kým
		koniec kým

Obrázok 6: Tri riešenia úlohy Programovanie trasy robota

6 ZÁVERY A DISKUSIA

Predovšetkým chceme zdôrazniť, že si uvedomujeme slabú stránku nášho výskumu, ktorou je nízky počet skúmaných nevidiacich žiakov. Na Slovensku nie je veľa nevidiacich žiakov nižšieho sekundárneho vzdelávania, na ktorí by mohli participovať na výskume. Pokúšali sme sa nadviazať spoluprácu aj s učiteľmi nevidiacich v Českej republike, ale podľa našich informácií sa výučbe programovania zatiaľ nevenujú.

Slabou stránkou nášho výskumu bola nielen malá vzorka, ale aj skutočnosť, že sme použili iba jeden vzdelávací scenár, a preto nezovšeobecňujeme naše zistenia. Považujeme to však za príležitosť pokračovať v tomto výskume s inými žiakmi (ktorí sú v súčasnosti v nižších ročníkoch) a s upravenými úlohami, ktoré budú lepšie reflektovať kognitívne nároky programovacích úloh na nevidiacich študentov.

Napriek uvedeným skutočnostiam považujeme niektoré naše zistenia za dôležité pre náš budúci výskum. Žiaci v našej výskumnej vzorke uprednostňovali použitie viacnásobne sa opakujúcej postupnosti základných príkazov pred cyklom, čo môže byť spôsobené tým, že každú úlohu riešia ako jeden konkrétny problém. Premýšľanie o všeobecnom riešení je pre nich veľmi abstraktné. Aby mohli nájsť všeobecné riešenie, problém sa musí rozdeliť na niekoľko podproblémov, ako sme to uviedli aj v [12]. V ďalšej iterácii našich úloh by sme chceli použiť lepšiu motiváciu, ktorá by pomohla študentom vidieť problémy vo všeobecnejšej rovine.

Žiaci tiež uprednostňovali podmienený cyklus pred príkazom vetvenia. Môže to byť spôsobené jednak komplikovanou syntaxou príkazu vetvenia v jazyku Coshi, ale aj neporozumením tohto pojmu. Mühling et al. [16] zisťovali programátorské zručnosti študentov 7. až 10. ročníka a zistili, že študenti často chápú „príkaz vetvenia ako cyklus, ktorý sa vykonáva, pokiaľ je podmienka splnená“. Bolo to pravdepodobne spôsobené tým, že študenti neboli zvyknutí používať samostatný príkaz vetvenia, ale väčšinou ho používali v cykle. V našom prípade je však pravdepodobnejšie, že žiaci si uvedomovali, že rovnako dobre ako príkaz vetvenia, môžu použiť podmienený cyklus, ale zvolili si alternatívu s jednoduchšou syntaxou.

Vo všeobecnosti môžeme konštatovať, že prostredie Coshi sa ukázalo ako vhodné pre nevidiacich žiakov. Má dobrú zvukovú odozvu a tvorba programu výberom príkazov z ponuky alebo pomocou klávesových príkazov je pre nevidiacich žiakov vyhovujúca. Príkazy jazyka sú pre žiakov

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitní a pedagogická zrozumiteľné, ale je potrebné implementovať nový podmienený príkaz bez vetvy *inak* (*else*). Možnosť vytvárať si vlastnú zvukovú schému, pomáha učiteľovi prispôbiť motiváciu pre svojich študentov. V budúcnosti plánujeme overiť aj ďalšie edukačné aktivity zamerané na použitie podprogramov bez parametra a s parametrom.

7 POĎAKOVANIE

Tento článok vznikol v rámci projektu financovaného z grantu Kega 018UK-4/2019. Ďakujeme učiteľom a žiakom základnej školy pre žiakov so zrakovým postihnutím, kde sme realizovali náš výskum.

8 BIBLIOGRAFICKÉ ODKAZY

- [1] HROMKOVIČ, J., et al. *Examples of algorithmic thinking in programming education*. Olympiads in Informatics, 10(1-2), 2016, s. 111–124.
- [2] PAPERT, S. What is Logo? And who needs it? In: *Logo philosophy and implementation*. Logo Computer Systems, 1999, s. iv–xvi.
- [3] BLAHO, A., et al. *Imagine Logo Primary workbook*. Cambridge: Logotron, 2004.
- [4] RESNICK, M., et al. Scratch: Programming for all. In: *Commun. Acm*, 52(11), 2009, s. 60–67.
- [5] MILNE, L. R. et al. Blocks4All: overcoming accessibility barriers to blocks programming for children with visual impairments. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ACM, 2018, s. 69.
- [6] AARON, S. *Code music with Sonic Pi* [online]. [vid. 16. 12. 2019]. Dostupné na: https://www.raspberrypi.org/magpi-issues/Essentials_Sonic_Pi-v1.pdf.
- [7] BENDER, W., et al. Music Blocks: A Musical Microworld. In: *Proceedings of the 2016 Constructionism*, Bangkok, Thailand, 2016.
- [8] SÁNCHEZ, J., et al. APL: audio programming language for blind learners. In: *International Conference on Computers for Handicapped Persons*, Berlin, Heidelberg: Springer, 2006, s. 1334–1341.
- [9] SÁNCHEZ, J., et al. Blind learners programming through audio. In: *CHI'05 extended abstracts on Human factors in computing systems*, ACM, 2005, s. 1769–1772.
- [10] KOVÁČ, Matúš. *Programovacie prostredie pre zrakovo postihnutých žiakov*, bakalárska práca, Bratislava: FMFI UK, 2016.
- [11] MORRISON, C., et al. Torino: A tangible programming language inclusive of children with visual disabilities. In: *Human-Computer Interaction*, 2018, s. 1–49.
- [12] KOVÁČOVÁ, N., et al. Dva programovacie jazyky priateľské k nevidiacim žiakom nižšieho sekundárneho vzdelávania In: Horváthová, D. eds. *Didinfo 2019*, Banská Bystrica: Univerzita Mateja Bela, 2019. ISBN 978-80-557-1533-9.
- [13] KOVÁČ, Michal. *Programovacie prostredie na ovládanie pohybu virtuálneho objektu prístupné pre nevidiacich žiakov sekundárneho vzdelávania*, bakalárska práca, Bratislava: FMFI UK, 2019.
- [14] HENDL, J. *Qualitative Research. Basic Methods and Applications*. Praha: Portál, 2005.
- [15] CRESWELL, J. *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research*. New Jersey: Pearson Education, 2008.
- [16] MÜHLING, A., et al. Design and First Results of a Psychometric Test for Measuring Basic Programming Abilities. In: *Proceedings of the Workshop in Primary and Secondary Computing Education*, ACM, 2015, s. 2–10.

Jazyk Python ako hlavný nástroj vo výučbe informatických predmetov na stredných školách

Python as the Main Programming Language in the Teaching of Computer Courses in High Schools

Miroslav Biñas
Katedra počítačov a informatiky,
Technická univerzita Košice
Letná 9
042 00 Košice
Slovensko
miroslav.binas@tuke.sk

Matej Madeja
Katedra počítačov a
informatiky,
Technická univerzita Košice
Letná 9
042 00 Košice
Slovensko
matej.madeja@tuke.sk

Danka Bruothová
Gymnázium sv. T. Akvinského
Zbrojničná 3
040 01 Košice
Slovensko
gta.bruothova@gmail.com

ABSTRAKT

O popularite jazyka *Python* a jeho využití v úvodných kurzoch programovania, sa nedá pochybovať. Dokazujú to rozličné štatistiky a prieskumy na serveroch ako *StackOverflow* [1], *Tiobe* [2] alebo *PYPL Index* [3]. Výsledky prieskumu [4] z roku 2014 hovoria, že v danom roku bol jazyk *Python* najpoužívanejším jazykom v úvodných kurzoch informatiky (tzv. kurzy *CS0* a *CS1*) na univerzitách v USA.

V posledných rokoch tento jazyk zažíva „boom“ aj na pôde stredných škôl na Slovensku, kde postupne vytlačá jazyk *Pascal*, ktorý si svoje prvenstvo drží veľmi dlho. Aj napriek vlastnostiam jazyka *Python* a jeho všestranosti však častokrát jeho použitie zostáva len v úvodných kurzoch algoritmizácie a programovania. Ďalšie témy, akými sú objektové programovanie, programovanie mikrokontrolérov alebo tvorba webových stránok, sú následne prezentované pomocou iných programovacích jazykov.

V tomto článku sa pokúsime predstaviť možnosť využiť jazyk *Python* ako jediný jazyk, ktorý bude žiakov sprevádzať počas štúdia na stredných školách. Predstavíme vhodné moduly, rámce a jeho varianty na zvládnutie najčastejších scenárov využitia jazyka *Python* v informatických predmetoch. Rovnako sa pokúsime z tohto pohľadu zhodnotiť aj výstupy projektu *IT Akadémia*¹, ktorý si dal za cieľ priniesť inovácie aj do vzdelávania informatiky na stredných školách. Vyhodnotíme aj experimentálne overenie niektorých prístupov vo výučbe stredoškolských žiakov.

ABSTRACT

Nowadays there is no doubt about the popularity of Python and its use in introductory programming courses. This can be seen in many surveys such as StackOverflow [1], Tiobe [2], or PYPL Index [3]. Results of the survey [4] from 2014 indicate, that Python was in that year the most used language in introductory programming courses (so-called CS0 and CS1) at universities in the USA.

In the last years, this language is raising also at the high schools in Slovakia, where it is slowly replacing the Pascal language, which was the top language for a long time. However, despite the features of Python, the language is used only in introductory programming courses. For the following topics, such as object-oriented programming, programming of microcontrollers, or web site development, other languages are used.

In this article, we will try to present the idea of using Python as the only language for the study at high schools. We will introduce suitable modules, frameworks and it's variants to handle most common usages of Python in IT courses. From this point of view, we will also try to evaluate the outputs from the project IT Akadémia 1, which aims to bring innovations to the education of informatics at high schools. We will also evaluate some approaches in the teaching of high school students.

¹ <http://itakademia.sk>

Klíčová slova

algoritmizácia, programovanie, objektové programovanie, mikrokontrolér, Python

Keywords

algorithmization, programming, object-oriented programming, microcontroller, Python

1 PREČO JE DOBRÉ VENOVAŤ SA JEDNÉMU JAZYKU MIESTO VIACERÝCH

Informačné technológie (IT) sú oblasťou, v ktorej sa trendy menia veľmi rýchlo. Ak sme v úvode na základe dostupných zdrojov uviedli, že jazyk *Python* je v súčasnosti veľmi populárny, nemusí to byť pravda o 10 rokov. Rovnako to nemusí byť pravda už o rok. Preto držať krok s aktuálnymi trendmi v predmetnej oblasti je veľmi náročné.

Ako učitelia sa v rámci priestoru, ktorý máme, snažíme žiakom predstaviť (podľa nás) tie najlepšie technológie, s ktorými sa budú vedieť najlepšie uplatniť na trhu práce. To samozrejme odráža aj naše možnosti a schopnosti. Pri predstavovaní viacerých programovacích jazykov sa však neúmyselne dopúšťame niekoľkých chýb. Aj v tomto prípade by bolo možné použiť ľudovú múdrosť, ktorá hovorí, že: „Menej je niekedy viac.“

Rudolf Pecinovský vo svojom článku [5], v ktorom predstavuje metodiku efektívnej výučby OOP, uvádza veľmi častý problém, ktorým trpia štandardné učebnice programovania, ktoré „*miesto výučby programovania učia svojich čitateľov a frekventantov len syntax preberaného jazyka*“. Toto tvrdenie je však možné vzťahovať aj na štandardné kurzy programovania. Aj napriek tomu, že dnes existuje množstvo rozličných metodík, ktoré sa snažia popularizovať problematiku programovania a zvýšiť tak prirodzený záujem o tento odbor, častokrát naše kurzy a učebnice zostávajú len v predstavovaní syntaxe preberaného jazyka. Tým, že sa budeme snažiť každý rok predstavovať žiakom v rámci výučby niektorú novú paradigmu alebo technológiu v novom jazyku, sme veľmi náchylní sklznúť práve do tohto problému.

Je tiež potrebné si uvedomiť, že pri zmene jazyka nie je jazyk to jediné, čo sa vo výučbe zmení, a čo, resp. s čím treba žiakov zoznámiť. Vždy sú to aj nástroje, ktoré budú žiaci používať. Opustíme nástroje, ktoré sme ich naučili používať predchádzajúci ročník a nahradíme ich novými, s ktorými ich musíme naučiť pracovať. A na to sú opäť potrebné hodiny práce, pretože ako usporiadanie používateľského rozhrania, tak aj klávesové skratky, môžu byť v nových nástrojoch rôzne.

Použitím jedného programovacieho jazyka vo výučbe sa zvyšuje aj zastupiteľnosť samotných učiteľov. Tento problém je možné kvantifikovať aplikáciou tzv. „*bus factor*“-u, ktorý sa používa v IT projektoch. *Bus Factor* je reprezentovaný číslom, ktoré predstavuje mieru rizika v strate informácií v prípade výpadku ľudského zdroja (obrazne povedané – človek je prejdený autobusom) [6]. Najnižšia hodnota je 1, kedy je za danú časť projektu zodpovedný len jeden človek. Čím vyššie je toto číslo, tým menej je projekt ohrozený. To isté teda platí aj v školstve – čím viac učiteľov pre daný predmet máme, resp. čím viac učiteľov ovláda predmetnú oblasť/technológiu, tým viac sme pripravení na potenciálny problém súvisiaci s výpadkom učiteľa. Ten nemusí byť nutne prejdený autobusom, ale môže ochoriť, otehotniť alebo len proste odísť zo školy.

Zastupiteľnosť učiteľov a teda vyššia hodnota *bus factor*-u nesie so sebou aj ďalší benefit, ktorým je možnosť konzultovať s kolegami pripravené materiály, poprípade riešiť vzniknuté problémy súvisiace s použitým jazykom a technológiami. Taktiež sa zvyšuje možnosť púšťať sa do spoločných softvérových projektov s vlastnými kolegami.

Naposlednou výhodou, ktorú so sebou prináša použitie jedného jazyka naprieč celým štúdiom na strednej škole, je možnosť stavať na už nadobudnutých znalostiach z nižších ročníkov. To môže byť napríklad použitie konkrétnej paradigmy pre zvolenú novú technológiu alebo kombinovanie softvérových projektov s hardvérom. Tým pádom sa žiaci nemusia stále zapodievať základmi a

syntaxou nového jazyka, ale môžu sa plne sústrediť na jeho nové rozšírenia a použitia prezentované učiteľom.

Výhody použitia jedného jazyka je možné zhrnúť v nasledovných bodoch:

- keďže žiaci vo vyšších ročníkoch štúdia už poznajú ako jazyk tak aj vývojárske nástroje, môžu **takmer okamžite začať vytvárať programy** v novej prezentovanej technológii
- **používanie jedného vývojového prostredie** umožní učiteľovi ušetriť čas súvisiaci s jeho inštaláciou (hlavne na domácich počítačoch) a problémami vzniknutými jeho pravidelným používaním, a sústrediť sa na nové veci
- zvyšuje sa **zastupiteľnosť učiteľov** informatických predmetov, kedy sa učiteľom stačí len dovzdelávať v predmetnej oblasti
- **zvyšujú sa možnosti využitia schopností a znalostí žiakov nadobudnutých v nižších ročníkoch**

2 INOVÁCIE VO VZDELÁVANÍ V NÁRODNOM PROJEKTE IT AKADEMIA

Národný projekt *IT Akadémia* odštartoval v roku 2016. Jeho strategickým cieľom je: *Vytvorenie modelu vzdelávania a prípravy mladých ľudí pre aktuálne a perspektívne potreby vedomostnej spoločnosti a trhu práce so zameraním na informatiku a IKT.* [7]

V rámci aktivity 1 sa projekt venuje stredným školám, kde vznikli nové materiály aj pre výučbu informatických predmetov. Výber predmetov zahrnutých do tohto projektu môžeme použiť pre analyzovanie súčasného stavu používaných programovacích jazykov vo výučbe na stredných školách. Ako odrazový mostík môže poslúžiť materiál označený ako *Model špeciálnej triedy so zameraním na informatiku* [8], ktorý obsahuje zoznam odporúčaných predmetov pre triedu gymnázia so zameraním na informatiku. Z tohto zoznamu je možné vybrať tieto predmety (s uvedením programovacieho jazyka vyučovaného predmetu):

- *Riešenie problémov a programovanie* [8] – jazyk *Python*
- *Programovanie mobilných zariadení* [9] – blokové, resp. grafické programovanie v prostredí *MIT APP Inventor 2*²
- *Objektový prístup k riešeniu problémov* [10] – jazyk *Java*
- *Programovanie webových stránok* [11] – jazyky *PHP* a *JavaScript*

Ako je možné vidieť, z uvedených predmetov je programovací jazyk *Python* použitý len v jednom predmete. Navyše každý z nich používa iný programovací jazyk. Pri takomto spôsobe výučby sa potom môžu prejavovať nedostatky uvedené v predchádzajúcej časti príspevku.

Tento zoznam predmetov je ešte možné rozšíriť o motivačný predmet *Internet vecí* [12], ktorý je príkladom práve prepojenia softvéru s hardvérom. Informačný list predmetu priamo menuje grafický programovací jazyk *Blockly*³ a jazyk *Python*. Vzhľadom na to, že obsahom predmetu je aj práca s prototypovacou doskou *Arduino*, je do zoznamu jazykov možné zahrnúť aj jazyk *C*. Na túto dosku je zamerané aj vzdelávanie pre učiteľov *Programovanie mikroprocesorových systémov na SŠ* [13], ktoré sa bežne vyučuje hlavne na stredných odborných školách.

Vo výsledku je teda možné zhodnotiť, že aj napriek popularite a univerzálnosti jazyka *Python* sú jeho vlastnosti v príprave predmetov v tomto projekte využité minimálne. V nasledujúcom texte sa pokúsime predstaviť možnosti jeho použitia aj v uvedených predmetoch. Z tohto zoznamu však vyradíme predmet *Programovanie mobilných aplikácií*, ktorý je kvôli cieľovej platforme veľmi špecifický. Pre úplnosť je možné uviesť, že aj v tejto oblasti existujú knižnice pre jazyk *Python*, pomocou ktorých je možné vytvárať mobilné aplikácie, ako napr. *Kivy*⁴ alebo *BeeWare*⁵.

² <https://appinventor.mit.edu>

³ <https://developers.google.com/blockly>

⁴ <https://kivy.org>

⁵ <https://beeware.org>

3 JAZYK PYTHON V PREDMETOCH MODELOVEJ TRIEDY

3.1 Základy algoritmizácie a programovania

Naučiť sa programovať je ťažké. Učiť programovať niekoho iného je však ešte ťažšie. O to je dôležitejšie používať tie správne nástroje, ktoré pomôžu žiakom nie len porozumieť konštrukciám, ale ich aj dostatočne nadchnúť, aby sami objavovali nové veci jednoduchými úpravami kódu.

Trend zvyšovania motivácie je vidieť na mnohých vzdelávacích portáloch, kde ako prostriedok na dosiahnutie cieľa sa používajú počítačové hry. Tak je to napr. na portáloch ako *code.org*⁶ alebo *CodinGame*⁷. Existujú však aj rokmi overené metodiky ako napr. *Robot Karel* [14], *Korytnačia grafika*, ktoré majú svoje implementácie aj v jazyku *Python*.

Je zaujímavé, ako často my učitelia týmto prístupom opovrhujeme, pretože považujeme hry za niečo zlé. A pritom sú častokrát počítačové hry prostriedkom, ako sa mnohí žiaci dostali k štúdiu technických odborov. Výborným a moderným príkladom je napríklad hra *Minecraft*⁸, ktorá je v špeciálnej edícii dostupná na mikropočítači *Raspberry Pi*. Je to hra, ktorú poznajú a hrajú aj deti v predškolskom veku. Miesto toho, aby sme zareagovali aplikovaním tejto hry do výučby programovania v jazyku *Python*, necháme žiakov radšej vypisovať postupnosť číslíc od 1 do 10 pri prezentovaní cyklov na čiernej obrazovke. Možnosť postaviť most z 10 kamenných blokov za sebou v prostredí, v ktorom sami trávia hodiny, tak pre nich zostane stále neobjavená, vid' obr. 1.

Minecraft Pi sme sa pokúsili vyskúšať vo výučbe jazyka *Python* na stredných školách niekoľkokrát. Rovnako sme sa ho snažili niekoľkokrát popularizovať v rámci tvorivých dielní na podujatiach *Pycon.sk*⁹ a *Namakaný deň*¹⁰ a vždy s úspechom. Hlavný nedostatok, ktorý bráni širšiemu nasadeniu takéhoto prístupu, je nedostatok dostupných materiálov. Dlhodobou jedinou knihou, ktorá sa venuje využitiu hry *Minecraft* ako prostredia pre výučbu úvodného kurzu programovania je [15].

3.2 Objektovo orientované programovanie

Pre potreby predmetu z projektu *IT Akadémia* bolo zvolené prostredie *Greenfoot*¹¹, ktoré je určené priamo na výučbu objektového programovania prístupom *Objects First Approach*. Nástroj vychádza z nástroja *BlueJ*¹², ale jeho cieľom je zvýšiť záujem o výučbu programovania tým, že žiaci budú vytvárať počítačové hry (a popri tom sa učiť princípy objektového programovania).

Nie všetky implementácie predmetov objektového programovania na stredných školách sú však orientované podobne. Pomerne často sa dá stretnúť aj s príkladmi, kedy sa výučba nezameriava ani tak na princípy objektového programovania, ako skôr na tvorbu grafických aplikácií. Objektový prístup je v tomto prípade potlačený, pretože žiaci sa výrazne viac venujú návrhu grafického používateľského rozhrania aplikácií a ošetrovaniu rozličných udalostí (napr. kliknutie na tlačidlo).

Ak by sme hľadali podobný nástroj ako je *Greenfoot* v jazyku *Python*, tak ho nenájdeme. Vlastnosti jazyka ako *REPL* v kombinácii s vhodnými knižnicami, však dokážu prístup výučby *Objects First* úspešne zabezpečiť aj v jazyku *Python*.

Jednou z takých knižníc je knižnica *Pygame Zero*¹³, ktorá je postavená na známej hernej knižnici s názvom *Pygame*¹⁴. Túto knižnicu vytvoril *Daniel Pope* ako reakciu na podnet učiteľov. Tí sa často sťažovali, že „knižnica *Pygame* je síce dobrá, ale kým sa dostanú k tomu, aby na obrazovku vykreslili čo i len obdĺžnik, musia napísať príliš veľa úvodného kódu (z angl. *boilerplate*)“ [16].

⁶ <https://code.org>

⁷ <https://www.codingame.com>

⁸ <https://www.minecraft.net>

⁹ <https://pycon.sk>

¹⁰ <http://namakanyden.sk>

¹¹ <https://www.greenfoot.org>

¹² <https://bluej.org>

¹³ <https://pygame-zero.readthedocs.io>

¹⁴ <https://www.pygame.org>

Knižnica je určená pre výučbu a umožňuje hry s jednoduchou mechanikou vytvárať veľmi rýchlo. Vzhľadom na skúsenosti z výučby učiteľov a študentov pomocou tejto knižnice alebo z rozličných prednášok a tvorivých dielní, vieme povedať, že hru je možné vytvoriť v priebehu 30 minút aj so spoločnou diskusiou s účastníkmi. Výsledok takejto aktivity sa nachádza na obrázku 2, kde sme za tento čas vytvorili základ hry *Flappy Bird* v rozsahu 43 riadkov kódu.

Knižnica je výborná aj na prezentovanie objektových princípov. Vzhľadom na to, že sa jedná o hernú knižnicu, pomáha zvyšovať motiváciu u samotných žiakov a podporuje ich kreativitu. To sme mohli vidieť aj pri výbere záverečných projektov žiakov počas dvoch rokov, kedy si v sumare 9 žiakov z 26 vybralo práve vytvorenie vlastnej hry pomocou tejto knižnice.

3.3 Programovanie webových aplikácií

Predmety na stredných školách, ktoré sa venujú tvorbe webových aplikácií, sú veľmi populárne. Táto skutočnosť odráža aktuálne trendy, keďže obrovské množstvo IT riešení je orientované na web a webové technológie. To je možné potvrdiť aj výsledkami prieskumu spoločnosti *Jetbrains* [17], ktorá vytvára nástroje pre programátorov. Tá na začiatku roku 2019 oslovila takmer 7000 vývojárov, aby sa opýtala na ich vývojársky ekosystém.

Podľa toho, kde je webová aplikácia spustená, rozlišujeme dva základné typy aplikácií:

1. aplikácie na strane servera, a
2. aplikácie na strane klienta.

Pri realizácii aplikácií na strane klienta má svoje nezastupiteľné miesto jazyk *JavaScript*. Aj napriek tomu však existujú snahy o nahradenie tohto jazyka aj jazykom *Python* (napr. projekt *Brython*¹⁵).



Obrázok 1: Most v hre Minecraft vytvorený v jazyku Python

Na strane servera je situácia pre jazyk Python výrazne lepšia. Existuje pomerne veľká množina projektov, pomocou ktorých je možné vytvárať webové aplikácie na strane servera (tzv. „back-end“ aplikácie). Obecne je možné tieto riešenia rozdeliť do dvoch základných kategórií:

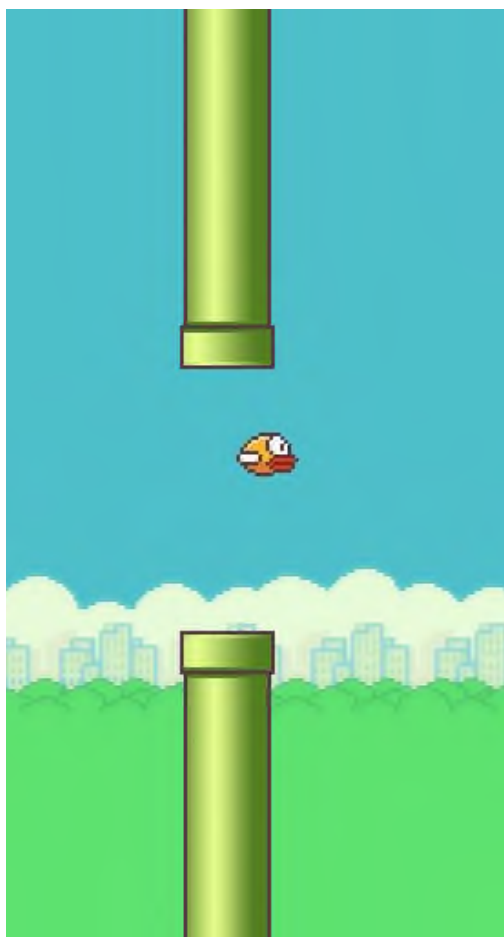
¹⁵ <https://brython.info>

1. tzv. *micro web* rámce, ktoré obsahujú len základnú funkcionality, ako napr. smerovač, a
2. tzv. *full-stack* rámce, ktoré v základnej inštalácii obsahujú všetko potrebné pre vytvorenie akéhokoľvek typu serverovej webovej aplikácie.

Z výsledkov spomínaného prieskumu [17] zameraných na jazyk *Python* vyplýva, že najpopulárnejšími webovými rámcami sú *Django*¹⁶ (*full-stack* rámec) a *Flask*¹⁷ (*micro web* rámec).

Webový rámec *Django* môže poslúžiť ako výborná pomôcka na výučbu tvorby serverových webových aplikácií, pretože núti programátora osvojiť si dobré návyky od architektúry projektu, cez aplikovanie návrhového vzoru *Model View Controller* (MVC), resp. *Model Template View* (MTV), až po tvorbu REST API. Rámec *Flask* naopak vývojára ničím nezaväzuje a poskytuje mu obrovskú voľnosť v tom, ako bude aplikáciu vytvárať. Pre neskúseného vývojára sa však táto voľnosť môže zmeniť na obrovský problém, ktorý bude narastať s komplexnosťou aplikácie. Preto je vhodné *Flask* používať skôr na menšie projekty.

V prípade výučby na stredných školách sa však na tvorbu serverových webových aplikácií najčastejšie používa jazyk *PHP*. V kontexte výučby je však voľba tohto jazyka asi tou najhoršou voľbou, pretože učíme žiakov jazyk, ktorý je možné použiť len na tvorbu serverových webových aplikácií. Stránka jazyka [18] však uvádza, že s ním je možné vytvárať aj skripty pre príkazový riadok a aj desktopové aplikácie pomocou knižnice *PHP-GTK*¹⁸. Posledná stabilná verzia tejto knižnice je však z roku 2015.



**Obrázok 2: Hra Flappy Bird
vytvorená v knižnici Pygame Zero**

¹⁶ <https://www.djangoproject.com>

¹⁷ <http://flask.palletsprojects.com>

¹⁸ <http://gtk.php.net>

Okrem uvedeného faktu aj realizácia predmetu častokrát pripomína obdobie okolo roku 2000, keď ešte neexistovali žiadne rámce, ktoré uľahčujú tvorbu webových aplikácií. Vývojári si teda museli všetko implementovať sami, čím sa častokrát neúmyselne dopúšťali rozličných chýb. Asi najznámejším problémom takéhoto prístupu je vzájomne pretkávaný *PHP* kód s *HTML* kódom, čo indikuje absenciu aplikácie návrhového vzoru *MVC*, resp. *MTV*, čím je možné oddeliť aplikačnú logiku od prezentačnej vrstvy. Na tomto vzore je dnes založený každý moderný webový rámec.

V rámci projektu *IT Akadémia* sa v predmete *Programovanie webových stránok* nachádza snaha pokryť obe oblasti – tvorbu klientskych aplikácií v jazyku *JavaScript*, a tvorbu serverových aplikácií v jazyku *PHP*. Aby žiaci mohli s novými jazykmi pracovať, musia sa zoznámiť so základmi syntaxe, ako aj s novými nástrojmi. Toto zoznámenie však podľa informačného listu predmetu [11] zaberá približne tretinu času celého kurzu.

3.4 Programovanie mikrokontrolérov a Internet vecí

Najmä stredné odborné školy so zameraním na elektrotechniku vyučujú predmety, v rámci ktorých učia žiakov programovať mikrokontroléry. Veľmi populárne sú v tejto oblasti prototypovacie dosky *Arduino*, konkrétne *Arduino Uno*. Táto doska bola vytvorená v roku 2003 a za krátky čas sa stala veľmi populárnou vďaka prijateľnej cene. Existencia obrovského množstva dokumentácie a rozširujúcich „shield-ov“ z nej robí ideálneho kandidáta pre použitie v školstve.

Túto dosku je možné programovať v dvoch jazykoch: v jazyku *C* a v jazyku *JavaScript* (projekt *Johnny Five*¹⁹). Programovať ju v jazyku *Python* však nie je možné kvôli obmedzeným parametrom mikrokontroléra *ATmega328P*. Toto obmedzenie je možné čiastočne obísť pomocou protokolu *Firmata*²⁰, kedy vie *Arduino* prijímať príkazy cez sériovú linku napr. aj z jazyka *Python*. To si ale vyžaduje mať navyše zariadenie, ktoré bude doske *Arduino* tieto príkazy posielať.

Dnes už však existujú náhrady, ktoré dokážu *Arduino* prekonať ako v parametroch, tak aj v cene. Jednou z nich je aj mikrokontrolér *ESP32*²¹, ktorý má dostatok pamäte a prostriedkov na to, aby v ňom mohol fungovať *MicroPython*²².

MicroPython je špeciálna verzia jazyka *Python 3*, ktorá obsahuje malú podmnožinu štandardnej knižnice a je upravená a optimalizovaná tak, aby mohla fungovať v mikrokontroléroch. Pomocou jazyka *MicroPython* je možné programovať napr. aj populárne zariadenia *BBC micro:bit*²³.

Výhodou mikrokontroléra *ESP32* oproti prototypovacej doske *Arduino Uno* je však aj jeho výbava. V základe totiž tento mikrokontrolér obsahuje aj podporu pre *WiFi* a *Bluetooth LE*, čo z neho robí ideálneho kandidáta na zariadenie pre oblasť *Internetu vecí*.

Počas posledných dvoch rokov sme mali možnosť školiť učiteľov stredných škôl práve v oblasti *IoT*. *Arduino Uno* sme postupne nahradili mikrokontrolérom *ESP32* a aj v tomto kurze sa tým pádom stal jazyk *Python* jediným programovacím jazykom, čo výučbu značne urýchlilo. Samotní učitelia, ktorí dovtedy mali skúsenosť len s doskami *Arduino* ocenili, že to, na čo potrebujú osobitné rozširujúce „shield-y“ majú rovno integrované v jednom zariadení. Oslovení však boli najmä jednoduchosťou implementácie výsledných riešení, ktorú priniesol práve *MicroPython*.

4 ZÁVER

V tomto článku sme prezentovali výhody použitia jedného programovacieho jazyka pre celé štúdium žiakov na stredných školách. Konkrétne sme zvolili jazyk *Python*, ktorý je možné použiť na tvorbu všetkých bežných typov aplikácií a je tiež výbornou voľbou na prvý programovací jazyk, s ktorým by sa mohli žiaci stretnúť. Je tiež veľmi populárny v oblasti dátovej analýzy a strojového učenia.

¹⁹ <http://johnny-five.io>

²⁰ <https://github.com/firmata/protocol>

²¹ <http://esp32.net>

²² <http://micropython.org>

²³ <https://microbit.org>

Pri analýze informačných listov vybraných predmetov sme tiež zistili, že čas potrebný na predstavenie základnej syntaxe nového jazyka, je na úrovni 25% až 30% z celkového trvania kurzu. To v súčte znamená, že ak v každom ročníku bude použitý nový programovací jazyk, jeden celý ročník štúdia bude venovaný opakovaným základom programovania, ale zakaždým v inom jazyku.

Okrem ušetrenia času opakovaním sme identifikovali aj ďalšie výhody. Jednou z nich je, že žiaci budú vedieť súčasne používať rozličné paradigmy alebo budú vedieť kombinovať softvérové riešenia s hardvérovými. Nie je teda problém vytvoriť grafickú počítačovú hru, ktorej výsledky sa budú prezentovať pomocou webovej stránky, a ktorá sa bude dať ovládať vlastnoručne vyrobeným bezdrôtovým ovládačom.

Ako riešitelia projektu *IT Akadémia* sme mali možnosť v rokoch 2018 až 2020 lektorovať 6 kurzov, ktorých sa zúčastnilo vyše 80 učiteľov stredných škôl. Okrem toho sme lektorovali aj dve tvorivé dielne počas výročnej konferencie projektu *IT Akadémia* v roku 2019, ktorých sa zúčastnilo vyše 50 učiteľov stredných a vysokých škôl. Všetky uvedené aktivity sme realizovali len pomocou jazyka *Python*. Mnohí z účastníkov vôbec netušili o možnosti programovať webové stránky a mikrokontroléry v jazyku *Python*. Aj táto skúsenosť prispela k tomu, že v diskusiách o možnosti využitia tohto jazyka ako jediného jazyka na stredných školách, videli obrovský potenciál.

5 POĎAKOVANIE

Príspevok bol spracovaný v rámci projektu OP Ľudské zdroje „*IT Akadémia – vzdelávanie pre 21. storočie*“ (ITMS: 312011F057).

6 BIBLIOGRAFICKÉ ODKAZY

- [1] StackOverflow: *Developer Survey Results 2019* [online]. [vid. 17.12.2019]. Dostupné na <https://insights.stackoverflow.com/survey/2019>.
- [2] *TIOBE Index for December 2019* [online]. [vid. 17.12.2019]. Dostupné na <https://www.tiobe.com/tiobe-index>.
- [3] *PYPL PopularitY of Programming Language*. [online]. [vid. 17.12.2019]. Dostupné na <https://pypl.github.io/PYPL.html>.
- [4] GUO, Philip: *Python Is Now the Most Popular Introductory Teaching Language at Top U.S. - Universities*. 7.7.2014 [online]. [vid. 17.12.2019]. Dostupné na <https://cacm.acm.org/blogs/blog-cacm/176450-python-is-now-the-most-popular-introductory-teaching-language-at-top-u-s-universities/fulltext>.
- [5] PECINOVSKÝ, Rudolf: *Jak efektivně učit OOP*, National Conference with International Participation on Software Creation, vol. 31, 2005, pp. 174–182. [online]. [vid. 17. 12. 2019]. Dostupné na <https://pdfs.semanticscholar.org/d50c/fa981ded2bf03fc6e3db8e7a85e82ce75180.pdf>.
- [6] BOWLER, M.: *Truck Factor*, (May 15, 2005). „Truck Factor“. Agile Advice.
- [7] *Domovská stránka projektu IT Akadémia* [online]. [vid. 17. 12. 2019]. Dostupné na: <http://itakademia.sk/zakladne-informacie>.
- [8] GUNIŠ, Ján a kol.: *Riešenie problémov a programovanie* (Informačný list predmetu) [online]. [vid. 17. 12. 2019]. Dostupné na http://itakademia.sk/wp-content/uploads/2018/07/Riesenie_problemov_a_programovanie.pdf.
- [9] ŠNAJDER, Ľubomír a kol.: *Programovanie mobilných zariadení* (Informačný list predmetu) [online]. [vid. 17. 12. 2019]. Dostupné na http://itakademia.sk/wp-content/uploads/2018/07/Programovanie_mobilnych_zariadeni.pdf.
- [10] VARGA, Michal a kol.: *Objektový prístup k riešeniu problémov* (Informačný list predmetu), [online] [vid. 17. 12. 2019]. Dostupné na http://itakademia.sk/wp-content/uploads/2018/07/Objektovy_prístup_k_rieseniu_problemov.pdf.

- [11] HRUŠECKÝ, Roman a kol.: *Programovanie webových stránok* (Informačný list predmetu), [online] [vid. 17. 12. 2019]. Dostupné na http://itakademia.sk/wp-content/uploads/2018/07/Programovanie_webovych_stranok.pdf.
- [12] JANITOR, Jozef a kol.: *Internet vecí* (Informačný list predmetu), [online] [vid. 17. 12. 2019]. Dostupné na http://itakademia.sk/wp-content/uploads/2018/07/Internet_veci.pdf
- [13] ŠNAJDER, Ľubomír a kol.: *Programovanie mikroprocesorových systémov na SŠ* (Informačný list predmetu), [online] [vid. 17. 12. 2019]. Dostupné na <http://itakademia.sk/wp-content/uploads/2018/09/A-15.pdf>.
- [14] PATTIS, Richard E.: *Karel the Robot*, John Wiley & Sons; 2. vyd., 1995, ISBN 0471597252.
- [15] O'HANLON, Martin a WHALE, David: *Adventures in Minecraft*. 2. vyd. Indianapolis, In: Wiley, [2018]. ISBN 978-1119439585.
- [16] MCMANUS, Sean. *Pygame Zero: Game coding made easy* [online]. 2015, 4 (35) [vid. 17. 12. 2019]. ISSN 2051-9990. Dostupné na: <https://www.raspberrypi.org/magpi/issues/35/>.
- [17] JETBRAINS: *The State of Developer Ecosystem 2019* [online]. [vid. 17. 12. 2019]. Dostupné na: <https://www.jetbrains.com/lp/devecosystem-2019>.
- [18] THE PHP GROUP: *What can PHP do?* [online]. [vid. 17. 12. 2019]. Dostupné na: <https://www.php.net/manual/en/intro-whatcando.php>.

Bádateľsky orientované metodiky pre použitie App Inventora na strednej škole

Exploration-Oriented Teacher Materials for Using App Inventor in Upper Secondary Schools

Monika Tomcsányiová
KDMFI FMFI Univerzita Komenského,
Mlynská Dolina
842 48 Bratislava
Slovensko
monika.tomcsanyiova@fmph.uniba.sk

Peter Tomcsányi
KDMFI FMFI Univerzita Komenského,
Mlynská Dolina
842 48 Bratislava
Slovensko
peter.tomcsanyi@fmph.uniba.sk

ABSTRAKT

V rámci Národného projektu IT Akadémie sme vyvinuli 6 bádateľsky orientovaných metodík na vyučovanie informatiky ako všeobecno-vzdelávacieho predmetu na strednej škole pre oblasť Algoritmické riešenie problémov. Uvádza žiakov, ktorí už vedú programovať, do vývoja mobilných aplikácií pre operačný systém Android (verzia pre iOS je vo vývoji) pomocou prostredia App Inventor 2.0. V našom príspevku zoznámime poslucháčov so samotnými metodikami, s výsledkami ich testovania slovenskými učiteľmi ako aj s postupnými vylepšeniami metodík na základe týchto výsledkov.

ABSTRACT

As part of the National IT Academy project, we have developed 6 research-oriented lesson plans for teaching informatics as a general education subject at the upper secondary school for the area of Algorithmic Problem Solving. They introduce students who are already able to program to the development of mobile applications for the Android operating system (iOS version under development) using App Inventor 2.0. In our paper, we are presenting the lesson plans themselves, the results of their testing by Slovak teachers as well as gradual improvements of the lesson plans based on the results of testing.

Kľúčové slová

Národný projekt IT Akadémie, App Inventor, metodický materiál, algoritmické riešenie problémov

Keywords

National Project IT Academy, App Inventor, lesson plan, algorithmic solution of problems

1 ÚVOD

Jedným z výstupov Národného projektu IT akadémie [1] sú inovatívne metodiky pre vyučovanie informatiky na stredných školách. V tomto rámci sme vytvorili aj sériu šiestich bádateľsky orientovaných metodík pre App Inventor (11A-INFSS/AI2). Metodiky sú zamerané na vyučovanie informatiky ako všeobecno-vzdelávacieho predmetu na strednej škole pre oblasť Algoritmické riešenie problémov. Uvádzajú žiakov, ktorí už vedú programovať do vývoja mobilných aplikácií pomocou prostredia App Inventor 2.

Programovanie je dôležitou súčasťou výučby informatiky. Pri výučbe programovania ako všeobecno-vzdelávacieho predmetu na školách, ktoré nie sú primárne zamerané na informatiku je dôležitým faktorom motivácia žiakov. A práve použitie App Inventora pomáha motivovať žiakov lebo mnohí z nich pri výučbe vytvárajú aplikácie bežiacie na ich vlastných mobiloch.

2 APP INVENTOR

App Inventor 2 ([2], [3]) je jednoduché prostredie pre vývoj aplikácií pre mobilné zariadenia s operačným systémom Android (verzia pre iOS je vo vývoji). Samotné prostredie používa programovanie pomocou blokov, podobne ako prostredie Scratch. Pre vývoj aplikácií je potrebný stolný počítač alebo notebook, naprogramované aplikácie bežia v mobiloch či tabletoch.

App Inventor pôvodne začala vyvíjať firma Google. Keď tento projekt časom zrušil, ujala sa ho známa americká univerzita MIT (Massachusetts Institute of Technology). Prevádzkuje a vyvíja ho v Centre pre mobilné vyučovanie (Centre of Mobile Learning), ktoré je súčasťou Laboratória médií (MIT Media Lab).

Na rozdiel od iných programovacích prostredí, App Inventor potrebuje dôkladnejšiu prípravu a vyskúšanie hardvéru aj softvéru. Pri inštalácii softvéru do počítačov je často potrebná spolupráca učiteľa so správcom počítačov, čo môže predĺžiť čas potrebný na samotnú prípravu vyučovania. Odmenou za tento strávený čas môže byť vyššia motivácia žiakov pri práci s App Inventorom než s inými programovacími prostrediami (primeranými ich veku a znalostiam), pretože im to umožní naprogramovať mobilné aplikácie pre ich vlastné mobily či tablety.

3 PREDPOKLADANÉ VEDOMOSTI

Aby sme pri rozsahu 6 vyučovacích (dvoj)hodín mohli ukázať zaujímavé časti App Inventora, nemôžeme sa zároveň venovať výučbe základov algoritmizácie a programovania. Preto predpokladáme, že žiaci ovládajú základy algoritmizácie a základy programovania v niektorom programovacom jazyku a rozumejú (aspoň intuitívne) pojmom:

- program,
- premenná, priradenie hodnoty premennej,
- aritmetický výraz, logický výraz,
- príkaz vetvenia,
- procedúra/funkcia s parametrami.

Pre rýchlejšie pochopenie spôsobu programovania pomocou skladania programu z blokov v App Inventore môže pomôcť, ak žiaci už programovali v prostredí Scratch ([4]), ktoré používa rovnaký princíp. Texty v žiackych materiáloch často mlčky predpokladajú, že žiaci majú skúsenosť s prostredím Scratch. Ak to tak nie je, môže učiteľ rozšíriť frontálnu ukážku v prvej hodine na podrobnejšie predvedenie spôsobu skladania programov. Na druhej strane, podobnosť s prostredím Scratch môže niekedy žiaka (či učiteľa) aj pomýliť, lebo mnohé detaily má App Inventor iné, preto aj v materiáloch niekedy upozorňujeme na takéto rozdiely.

Pretože App Inventor nie je lokalizovaný do slovenčiny, predpokladáme u žiakov aj učiteľov pasívnu znalosť anglického jazyka na úrovni, ktorá umožní orientáciu v používateľskom prostredí App Inventora a porozumenie názvov príkazov a chybových správ, ktoré sa môžu zobrazit'. V prípade potreby možno pre dlhšie texty použiť on-line prekladač (napr. Google Translate), ale preklad jednotlivých slov (názvov príkazov alebo udalostí) nemusí žiakom pomôcť.

Ďalej predpokladáme aj znalosť ovládania operačného systému Android na úrovni bežného používateľa mobilného zariadenia.

4 METODIKY

Na Internete nájdeme množstvo metodík pre App Inventor. Avšak väčšina z nich je v angličtine a takmer všetky z nich používajú striktné inštrukcionistický prístup (typu step by step guide), pri ktorom postupnými krokmi čitateľ vytvorí daný projekt, ale pri jeho tvorbe väčšinou len nasleduje kroky popísané v metodickom materiáli, nezapája svoje analytické schopnosti ani invenciu.

Naše metodiky sa preto líšia hlavne v tom, že sme sa držali metódy bádateľsky orientovaného vyučovania 5E [5]. V ňom sa kladie dôraz na to, že na začiatku nedáme žiakom presný návod ako

postupovať, ale dostanú artefakty, ktoré majú preskúmať a na základe ich skúmania nájsť postup, akým vyriešia nastolený problém.

Pri aplikovaní bádateľsky orientovaného vyučovania na prírodovedné predmety je celkom jasné, že artefakt, ktorý majú žiaci preskúmať je prírodný jav alebo prírodný výtvor (živý či neživý). Pri aplikovaní bádateľsky orientovaného vyučovania na hodiny informatiky sa musíme najprv zamyslieť, čo budú tie artefakty, ktoré budú žiaci v prvých fázach vyučovacej hodiny skúmať. Z viacerých možností uvádzaných v [5] sme sa pri tvorbe našich metodík rozhodli (a podobne sa rozhodli aj kolegovia – autori v iných predmetoch) nechať žiakov skúmať dva druhy artefaktov: čiastočne hotové aplikácie a programátorské prostredie (IDE) prostredie programovacieho jazyka.

Preto každá metodika (s výnimkou prvej, kde na jej je hlavným cieľom otestovať spojenie počítača s mobilným zariadením) začína (v časti Engage – zapojenie) predvedením hotovej alebo čiastočne hotovej aplikácie a pokračuje (v časti Explore – skúmanie) skúmaním komponentov a programu v čiastočne hotovej aplikácii. Žiaci sú navádzaní otázkami pracovného listu na skúmanie funkcie aj stavby konkrétnych častí programu tak, aby pochopili účel danej časti aj spôsob akým bola v prostredí skonštruovaná a aby vedeli podobnú konštrukciu buď použiť v inej časti programu alebo ju opraviť, upraviť, vylepšiť či rozvinúť. V tejto časti žiaci niekedy aj experimentujú – menia program a skúšajú, čo dané zmeny urobia.

Potom majú žiaci vysvetliť (v časti Explain – vysvetlenie), na čo prišli. Učiteľ môže skorigovať ich zistenia, ak nezodpovedajú realite resp. ak hrozí, že si vytvorili či vytvorila miskoncepciu. Vedomosti získané v prvých častiach vyučovacej hodiny žiaci potom použijú (v časti Elaborate – rozpracovanie) na opravenie a dokončenie nehotovej aplikácie, prípadne jej vylepšenie či rozvinutie aj na základe vlastných nápadov. V poslednej časti (Evaluate – hodnotenie) odporúčame učiteľom hodnotiť výsledné aplikácie, ktoré niektorí študenti predvedú, prípadne použiť aj samohodnotiace dotazníky, v ktorých žiaci sami odhadnú, čo sa na danej hodine naučili.

Tabuľka 1 uvádza stručnú charakteristiku všetkých šiestich metodík série.

Tabuľka 1: Stručná charakteristika šiestich metodík

Metodika	Názov	Stručný popis aktivity, použité komponenty a programátorské techniky
1	Zvuky zvierat	Vyskúšanie spojenia počítača a mobilného zariadenia. Orientácia v prostredí App Inventor. Jednoduchá multimediálna aplikácia s použitím obrázkov a zvukov. Komponenty Button a Player, ich vlastnosti, funkcie a udalosti.
2	Kreslenie	Pochopenie udalostí grafickej plochy a ich parametrov, naprogramovanie jednoduchého grafického editora. Tvorba grafického prostredia aplikácie pomocou objektov Button a Slider. Použitie vstavaného fotoaparátu, použitie fotografií uložených v mobile.
3	Akvárium	Programovanie pohyblivých objektov v prostredí App Inventor pomocou objektov typu ImageSprite a ich vlastností Speed, Heading a Rotates. Reakcia na dotyky, odrazy od okraja a kolízie. Vetvenie programu (blok if-then a if-then-else).
4	Triafaj	Animácia pomocou časovača. Naprogramovanie hry, v ktorej sa počítajú body a vyhodnocuje sa úspešný a neúspešný koniec hry. Komponenty Label a Ball. Globálne premenné a procedúry bez parametrov.
5	Nakláňaj a zbieraj	Implementovať využitie senzora orientácie (Orientation sensor) pri programovaní aplikácie. Naprogramovanie hry, v ktorej sa pohyb objektu typu Ball riadi pomocou nakláňania mobilu.
6	Povedz	Využitie senzora zrýchlenia (Accelerometer) na implementovanie akcie po zatrasení mobilu a objektu pre hlasový výstup (TextToSpeech) na vytvorenie hovoreného výstupu v programe.

V niektorých metodikách sa však nevyhneme ani kratším či dlhším úsekom inštrukcionistického výkladu. Je to hlavne tam, kde musíme korigovať niektoré vlastnosti prostredia, ktoré (podľa nášho názoru) neboli naprogramované tak, aby sa dali intuitívne pochopiť. Príkladom takého vysvetlenie je postup správneho zostrojenia príkazového bloku if-then-else (viď obrázok 1). Počas vývoja metodík sme mali možnosť viackrát pozorovať problémy, ktoré majú používatelia App Inventora keď chcú prvýkrát v programe použiť blok if-then-else. Tu nepomáha ani skúsenosť so Scratchom, lebo v ňom sú pre používateľa pripravené bloky if-then aj if-then-else priamo v ponuke riadiacich blokov (Control).

*Blok pre vetvenie typu if-then-else v ponuke blokov App Inventora nenájdeme v žiadnej skupine blokov, musíme ho vytvoriť. A to tak, že zo zoznamu **Control** vyberieme prvý z blokov **if-then** a keď ho umiestnime do programu (v oblasti **Viewer** obrazovky **Blocks**), tak tam pridáme **else** použitím jeho **mutátora** – modrej ikony s ozubeným kolieskom v ľavom hornom rohu bloku:*



*Všimnite si, že v okienku, ktoré otvorí **mutátor**, je okrem **else** na výber aj **else if**. Ten sa zide v prípade viacnásobného vetvenia. Pomocou **else if** z neho môžeme vytvoriť napríklad blok, ktorý vidíme na obrázku vyššie vpravo.*

Obrázok 1: Ukážka časti inštrukcií pre zostrojenie príkazového bloku if-then-else

5 VÝVOJ

Používaním App Inventora vo vyučovaní na druhom stupni základnej školy, na strednej škole ako aj pri vyučovaní budúcich učiteľov na našej fakulte sme sa začali zaoberať už pred našou účasťou v projekte IT Akadémie vďaka projektu KEGA 080UK-4/2015 Podpora tvorby vzdelávacieho digitálneho obsahu pre mobilné zariadenia ([6], [7]). S Programovaním v App Inventore sme oboznamovali aj študentov druhého ročníka magisterského štúdia učiteľstva informatiky v predmete 2-UIN-238/15 Programovanie pre mobilné platformy pre SŠ, ktorý vyučujeme od akademického roku 2016/2017. Z týchto skúseností sme čerpali aj pri vývoji metodických materiálov pre projekt IT Akadémie.

Samotný vývoj materiálov prebehol v roku 2017. Počas vývoja sme využili naše skúsenosti s App Inventorom. Veľa času sme strávili jednak výberom aktivít, jednak ich prezentovaním metódou bádateľsky orientovaného vyučovania.

Výzvou pri výbere aktivít bolo to, že sme mali predstaviť učiteľom a žiakom App Inventor len v šiestich vyučovacích hodinách (resp. dvojhodinových). Pri takomto malom rozsahu sme nemohli ukázať všetky zaujímavé možnosti prostredia (napríklad práca s mapami a GPS, možnosť ovládať LEGO roboty, posielat SMS, čítať čiarové kódy, ...) ani zísť hlbšie do programátorských techník, ktorých vyučovanie App Inventor umožňuje (napr. použitie zoznamov či databázy). Zároveň sme však chceli predstaviť použitie aspoň niektorých špecifických vlastností mobilných zariadení, ktoré sa dajú pomerne ľahko použiť v App Inventore, ale s ktorými sa žiaci bežne nestretnú v klasickom úvode do programovania. V piatej metodike sme preto použili senzor orientácie a v šiestej metodike sme použili akcelerometer a hlasový výstup.

Výzvou pri uplatňovaní metódy bádateľsky orientovaného vyučovania bola naša počiatočná malá skúsenosť s týmto spôsobom vyučovania a jej aplikovania na programovanie. Naše riešenie sme už popísali v podkapitole Metodiky.

Metodiky sme vyvinuli v priebehu roku 2017. V priebehu vývoja sme si ich čiastočne overovali na študentoch našej fakulty.

6 PRVÉ KOLO OVEROVANIA A ÚPRAV

Prvé kolo overovania metodík učiteľmi prebehlo od marca 2018 do mája 2019. Dotazník vyplnilo 12 učiteľov zo 7 gymnázií resp. spojených škôl, 2 odborných škôl, a z 3 neuvedených škôl. Z nich všetky aktivity odskúšalo a okomentovalo len 8 učiteľov. Na našu otázku kto si už tie aktivity odskúšal sa konferencii IT Akadémie v októbri 2019 však ozvali ďalší učelia, ktorí aktivity odskúšali, len nevyplnili dotazník.

Väčšina používala kombináciu mobilov žiakov a školských tabletov. Jedna pani profesorka nechcela použiť žiacke mobily aby predišla sťažnostiam rodičov: „*Ak sa im v najbližšej dobe niečo s mobilom stane, ťažko rodičom vysvetlím, že za to nemôže aplikácia...*“. V jednom prípade sa nepodarilo oživiť emulátor.

Z analýzy dotazníkov sme zistili, že najväčšie problémy mali učelia s časovou dotáciou. V metodikách sme síce uvádzali, že odporúčame realizovať každú metodiku na dvojhodinovke, ale učelia, ktorí nemohli venovať App Inventoru toľko času, nestihli prejsť celú aktivitu za jednu hodinu. Zmenšenie rozsahu jednotlivých aktivít by podľa nášho názoru nebolo správne, lebo potom by sme v celej sérii šiestich metodík nedokázali predstaviť ani zmysluplné minimum možností App Inventora.

Preto sme metodiky upravili tak, že sme ich rozdelili na *základnú časť* a *rozširujúcu časť*. Základná časť je podľa nás minimum, ktoré by mal učiteľ urobiť na vyučovaní aby daná aktivita zmysel a mala by sa stihnúť aj v rámci jednej vyučovacej hodiny. Rozširujúcu časť môže učiteľ použiť podľa svojho uváženia buď počas vyučovania (ak je dosť času alebo len pre tých, ktorí už základnú časť spravili) alebo na domácu úlohu či domáci projekt alebo ich vôbec nepoužiť.

Zároveň sme v úvodnom texte viac zvýraznili naše odporúčanie použiť, ak to čas dovoľí, kompletne metodiky na dvojhodinovkách. Len v takomto čase dokážu žiaci naozaj prejsť všetkými fázami bádateľského prístupu, zvládnuť všetky úlohy (šikovnejší žiaci dokážu navrhnúť aj svoje úpravy v úlohách) a učiteľ bude mať dosť času na zhodnotenie aktivity. Ak to nie je možné, tak sme odporučili buď použiť len prvé tri metodiky, každú z nich na dvoch 45minútových hodinách (alebo jednej 90minútovej) alebo použiť všetkých šesť metodík, každú na jednej 45minútovej hodine, ale vynechať, zadať paralelne rôznym skupinám rôzne úlohy, alebo dať na domácu úlohu časti resp. úlohu rozširujúce časti, ktoré sme pre lepšiu orientáciu označili hviezdíčkom. Ďalšou navrhovanou možnosťou, ktorú nám navrhli niektorí učelia počas prvého kola overovania metodík, je prejsť každú tému za jednu 45minútovú hodinu a pridať jednu hodinu (na koniec), alebo dve hodiny (jednu po 3. alebo 4. aktivite a jednu na koniec), na ktorých by mohli žiaci prezentovať svoje práce (ak učiteľ zadal domáce úlohy) a/alebo dorobiť to, čo nestihli.

Okrem problému časovej dotácie sme opravili preklepy a iné menšie chyby v materiáloch, aplikovali sme úpravy, ktoré nám prikázal resp. odporučil garant a líder pracovnej skupiny. Taktiež sme pre každú metodiku zrušili materiál „vzorový vyplnený pracovný list“, ktorého text sa opakoval v samotnom metodickom materiáli.

Tabuľka 2 uvádza priemerné známky (ako v základnej škole), ktoré dostali naše metodiky v piatich hodnotených kritériách. Znamky do 2,0 nás tešia. Jediná horšia známka je za zvládnuteľnosť piatej aktivity.

Tabuľka 2: Znamky, ktoré sme dostali za sériu metodík

Metodika	vhodne zvolené pre stanovených cieľov	primerané splnenie intelektuálnej úrovni žiakov	zrozumiteľne formulované	žiakmi zvládnuteľné	rozvíjajú schopnosť tímovej spolupráce žiakov
1	1,1	1,3	1,2	1,3	1,3
2	1,2	1,4	1,2	1,5	1,4
3	1,1	1,4	1,3	1,7	1,5
4	1,2	1,3	1,4	1,8	1,4
5	1,2	1,5	1,4	2,1	1,3
6	1,0	1,3	1,2	1,7	1,1

Potešili nás aj hodnotenia poslednej aktivity, ktoré niektorí učitelia poňali ako zhodnotenie celej série metodík a napísali aj:

Ďakujem za kvalitne spracovanú sériu metodík.

Celkové hodnotenie metodiky – veľmi hodnotné a kvalitné materiály, veľká pomoc pre učiteľov aj žiakov. Ďakujeme za poskytnuté materiály. Jediný nedostatok – časová dotácia je nedostatočná.

Myslím si, že metodiky budem v budúcnosti plne využívať, lebo vnímam App Inventor ako super nástroj pre vstup do sveta programovania a tvorby algoritmov. Využíva mobil ako bravúrnú metodickú pomôcku, pretože v dnešnej dobe sú deti na mobily hodne fixované.

... všetky metodiky hodnotím pozitívne a chcem poďakovať autorom.

7 DRUHÉ KOLO OVEROVANIA A ÚPRAV

Druhé kolo overovania práve prebieha v čase písania tohto príspevku. Doposiaľ len dve pani učiteľky vyplnili dotazník aj to len pre prvú úlohu. Po skončení overovania predpokladáme, že bude možnosť spraviť ďalšie úpravy materiálu. Nielen na základe overovania, ale aj ako reakciu na vývoj prostredia App Inventor. To je vlastne problém všetkých materiálov k programovacím prostrediam, kde autori prostredia a autori materiálov netvorí jeden kolektív. Po každej aktualizácii App Inventora jeho vývojári pridávajú, menia alebo aj rušia niektoré funkcie jazyka, atribúty komponentov alebo reakcie prostredia a vďaka tomu sa môžu stať niektoré naše odporúčania, postupy alebo obrázky neaktuálnymi.

8 ZÁVER

Programovanie mobilných zariadení v rámci informatiky ako všeobecno-vzdelávacieho predmetu môže prispieť k zatraktívneniu tohto predmetu aj pre žiakov, ktorí sa neplánujú stať programátormi. Vyššia motivácia žiakov keď programy bežia (alebo môžu bežať) na ich vlastných mobiloch bola potvrdená aj učiteľmi v prvom kole overovania. K naplneniu tohto cieľa sme sa snažili prispieť aj my sériou metodík, ktorú sme predstavili v tomto článku.

Veríme, že po skončení projektu budú tieto metodiky hojne využívané učiteľmi v slovenských školách.

9 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 a v rámci projektu KEGA 029UKF-4/2018 Inovatívne metódy vo výučbe programovania v príprave učiteľov a IT odborníkov. Ďakujeme učiteľom-overovateľom za ich cenné pripomienky a komentáre k metodickým a učebným materiálom.

10 BIBLIOGRAFICKÉ ODKAZY

- [1] UPJŠ. IT Akadémia [online]. [vid. 12. 1. 2020]. Dostupné na: <http://itakademia.sk/>.
- [2] MIT. *AppInventor* [online]. [vid. 12. 1. 2020]. Dostupné na: <http://appinventor.mit.edu/>.
- [3] UNIVERSITY OF SAN FRANCISCO. *appinventor.org* [online]. [vid. 12. 1. 2020]. Dostupné na: <http://appinventor.org/>.
- [4] MIT. *Scratch* [online]. [vid. 12. 1. 2020]. Dostupné na: <http://scratch.mit.edu/>.
- [5] LUKÁČ, Stanislav, Ľubomír ŠNAJDER, Ján GUNIŠ a Zuzana JEŠKOVÁ: *Bádateľsky orientované vyučovanie matematiky a informatiky na stredných školách*. Košice: Univerzita Pavla Jozefa Šafárika v Košiciach, 2016. ISBN 978-80-8152-471-4. [vid. 12. 1. 2020] Dostupné na: <https://unibook.upjs.sk/img/cms/2016/pf/bov.pdf>.
- [6] TOMCSÁNYIOVÁ, Monika a Peter TOMCSÁNYI: Creating mobile applications in App Inventor. In: KOSTOLÁNYOVÁ, Kateřina, ed. *InformationandCommunicationTechnology inEducation. Proceedings*. Ostrava: University of Ostrava, 2015, s. 231–239. ISBN 978-80-7464-763-5.
- [7] TOMCSÁNYIOVÁ, Monika a Peter TOMCSÁNYI: Vyučovanie programovania vývojom androidových aplikácií vAppInventore. In: TRAJTEL, Ľudovít ed. *Didinfo 2015* Banská Bystrica: Univerzita MatejaBela, Fakulta prírodných vied, 2015, s 160–163. ISBN 978-80-557-0852-2.

Ako spestriť výučbu informatiky na školách?

How to Brighten up the Teaching of Computer Science at Schools?

Patrik Voštinár
Katedra informatiky FPV UMB
Tajovského 40
974 01 Banská Bystrica
Slovensko
patrik.vostinar@umb.sk

Dana Horváthová
Katedra informatiky FPV UMB
Tajovského 40
974 01 Banská Bystrica
Slovensko
dana.horvathova@umb.sk

Jarmila Škrinárová
Katedra informatiky FPV UMB
Tajovského 40
974 01 Banská Bystrica
Slovensko
jarmila.skrinarova@umb.sk

ABSTRAKT

Cieľom nášho článku je prezentovať kľúčové informácie, ale najmä splnenie cieľov projektu KEGA 009KU-4/2017 „Inovatívne metodiky v predmete informatika v sekundárnom vzdelávaní“. Na katedre informatiky, Fakulty prírodných vied Univerzity Mateja Bela v Banskej Bystrici bolo vytvorených a v rámci informatického krúžku aj overených v priebehu 2 rokov riešenia projektu viacero metodík. Okrem toho bolo zakúpené množstvo učebných pomôcok a zrealizovaných niekoľko workshopov, určených na disemináciu vytvorených nových postupov a metód. Všetky metodiky a učebné materiály sme umiestnili na portál, vo forme LMS kurzu, aby ich mohlo začať využívať čo najviac učiteľov základných a stredných škôl na hodinách informatiky. Prostredníctvom článku prezentujeme ukážku jednej metodiky, ako aj úspešnosť overenia projektu KEGA v pedagogickej praxi.

ABSTRACT

The aim of our article is to present key information, but especially the fulfilment of the goals of the project KEGA 009KU-4/2017 “Innovative methodologies in the subject of informatics in secondary education”. At the Department of Informatics, Faculty of Natural Sciences, Matej Bel University in Banská Bystrica, several methodologies were created and verified within the Informatics club (an extracurricular activity of middle school students) during 2 years of the project solution. In addition, a number of teaching aids were purchased and several workshops were organized to disseminate the new created procedures and methods. We have placed all methodologies and teaching materials on the portal, in the form of an LMS course, so that they can be used by as many primary and secondary school teachers as possible. Through the article, we present a demonstration of one methodology, as well as the success of verification of the KEGA project in pedagogical practice.

Kľúčové slová

informatika, inovácia vyučovania, kontinuálne vzdelávanie, metodika, štátny vzdelávací program, sekundárne vzdelávanie, učitelia informatiky

Keywords

Informatics, teaching innovation, continuous teacher training, methodology, State Educational Programme, lower secondary and secondary education, teachers of informatics

1 ÚVOD

Vyučovanie informatiky na Slovensku prechádza neustále veľkými zmenami, najmä od reformy vzdelávania v roku 2008. V súčasnosti sa do praxe zavádza inovovaný Štátny vzdelávací program pre primárne a sekundárne vzdelávanie, platný od školského roku 2015/2016 [1].

Podľa prof. Kalaša [2] sa začína starostlivo odlišovať digitálna gramotnosť od informatického poznania. Mnoho krajín pripravuje nové „kurikulá“, v ktorých centrom koncepcie je žiak, okolo

ktorého sú koncepty a zručnosti, ktoré má žiak nadobudnúť a následne sa špecifikujú predmety, ktoré majú pomôcť naplniť tieto veľké vzdelávacie ciele. Pri tvorbe nových materiálov a metodík je potrebný cit pre vývinovú primeranosť a ak chceme udržať produktívne a udržateľné informatické vzdelávanie, musíme ho postaviť správne a systematicky už od materskej školy, resp. od prvého stupňa.

Učiteľov informatiky je extrémne málo a sú pod neustálym tlakom, ktorý je spôsobený prudkými zmenami stavu spoločnosti, digitalizáciou verejnej správy a pod. Na tieto zmeny je potrebné rýchlo reagovať a aktualizovať obsah učiva a hľadať vhodné metódy výučby v súvislosti so súčasnými trendami v používaní digitálnych technológií. Ďalšou komplikáciou najmä na stredných školách je rôzna úroveň digitálnych kompetencií žiakov prichádzajúcich zo základných škôl a rôzna schopnosť využívať nové technológie a digitálne vzdelávacie materiály vo vzdelávaní. Preto je dôležité prihliadať na tieto okolnosti a inovovať okrem obsahu predmetu aj metódy a formy výučby. Učitelia predmetu informatika často nemajú požadovanú kvalifikáciu a snažia sa ju doplniť formou rozširujúceho štúdia informatiky alebo akreditovanými kurzami. Často sú tieto kurzy zamerané viac na používanie IKT vo vzdelávaní ako na metodiku vyučovania informatiky. Navyše učitelia z niektorých oblastí Slovenska majú ťažšiu dostupnosť účasti na ďalšom vzdelávaní z dôvodov väčšej vzdialenosti k existujúcim aktivitám vo väčších centrách informatiky, ktoré sa viacej sústreďujú pri univerzitách s učiteľským zameraním (Bratislava, Nitra, Banská Bystrica, Ružomberok, Prešov, Košice a Komárno) [3].

V rokoch 2017 až 2019 sme z tohto dôvodu spolu s Katedrou informatiky na Katolíckej univerzite v Ružomberku podali a následne pracovali na projekte „Inovatívne metodiky v predmete informatika v sekundárnom vzdelávaní“. Projekt financovala Kultúrna a edukačná grantová agentúra Ministerstva školstva, vedy, výskumu a športu Slovenskej republiky. Projekt bol zameraný na rozvoj vyučovania informatiky v sekundárnom vzdelávaní v súvislosti s odporúčaniami v inovovanom Štátnom vzdelávacom programe.

2 CIELE PROJEKTU

Cieľom projektu bolo vytvoriť a overiť inovatívne metodiky a učebné materiály pre výučbu vybraných tematických celkov v predmete informatika na základných a stredných školách s dôrazom na inovácie obsahu a metód výučby v súlade s požiadavkami inovovaného Štátneho vzdelávacieho programu.

Čiastkové ciele projektu sme stanovili tak, aby sme mohli systematicky postupovať od analýzy súčasného stavu vyučovania informatiky na základných a stredných školách, cez overenie dostupnosti a aktuálnosti metodík a učebných materiálov k vybraným tematickým celkom. Následne nato bolo potrebné začať vytvárať jednotlivé metodiky, prípadne ich inovovať a vytvoriť učebné materiály k vybraným tematickým celkom.

Počas trvania projektu bolo potrebné uskutočniť rôzne vzdelávacie aktivity pre učiteľov informatiky z praxe vo forme seminárov, elektronického kurzu a v rámci konferencií o vyučovaní informatiky DidInfo (ale aj mimo nich) zrealizovať workshopy pre učiteľov informatiky na základných a stredných školách. Pri plnení čiastkových cieľov sa predpokladalo, že do riešenia projektu budú zapojení aj študenti učiteľstva informatiky formou záverečných prác a študentskej vedeckej odbornej činnosti, čo sa nám v priebehu projektu osvedčilo a vzniklo tak viacero zaujímavých prác, dokonca dve boli ocenené aj na Česko-Slovenskej študentskej vedeckej konferencii v didaktike informatiky.

3 AKTIVITY PROJEKTU

V roku 2017 naša študentka Nika Klimová vymyslela zaujímavý projekt v rámci Komenského inštitútu a založila vtedy informatický krúžok. Zaangažovala študentov učiteľstva informatiky a aplikovanej informatiky, aby sa pod jej vedením venovali deťom základných škôl z Banskej

Bystrice [4]. Skúsenosti z výučby tohto krúžku boli publikované v niekoľkých článkoch, ktoré prinášajú inšpirácie nielen pre učiteľov informatiky [5, 6, 7].

Základnú myšlienku projektu sa darí ďalej rozvíjať, a tak sa deti banskobystrických základných škôl v rámci krúžku aj v súčasnosti zoznamujú s novými technológiami, riešia spolu rôzne zaujímavé úlohy a hlavne majú k dispozícii vlastného tútora/učiteľa, ktorý sa prispôsobuje schopnostiam, individuálnym potrebám a tempu každého žiaka. Do dnešného dňa sa krúžku zúčastnilo 80 žiakov základných škôl z banskobystrického regiónu.

Ďalším nemenej významným prínosom krúžku je skutočnosť, že naši študenti majú možnosť v reálnom prostredí školy rozvíjať svoje pedagogické zručnosti. Ide o výnimočnú formu predprípravy na pedagogickú prax, ktorej sa môžu zúčastniť všetci študenti učiteľstva informatiky v kombinácii, ako aj študenti aplikovanej informatiky študujúcich na katedre informatiky na našej fakulte. Všetci si tak majú možnosť vyskúšať povolanie učiteľa a popri vysvetľovaní deťom si efektívne zdokonaľovať svoje vysvetľovacie a komunikačné schopnosti. Od začatia projektu sa v tejto aktivite vystriedalo 56 študentov katedry informatiky a záujem o takúto formu vzdelávacej aktivity z roka na rok rastie.

Informatický krúžok poskytol aj priestor na overovanie inovatívnych metodík a učebných materiálov, ktoré boli v rámci projektu vytvorené.

Počas trvania projektu boli realizované nasledujúce tri workshopy:

- Robotická prezentácia ZŠ Radvanská, Banská Bystrica 2018 – kedy si počas týždňa programovania žiaci mohli vyskúšať najnovších robotov – Phiro Pro, mBot a BBC micro:bit.
- ZŠ Bakošoková, Banská Bystrica 2018 – kedy počas celodenného workshopu mali žiaci možnosť vyskúšať si naprogramovať vlastného lietajúceho drona Airblock a skákajúceho robota Parrot Jumping Sumo.
- Počas konferencie DidInfo 2019 sa konal v Banskej Bystrici workshop, ktorý bol určený pre učiteľov informatiky na základných, stredných a vysokých školách a bol zameraný na programovanie edukačných robotov. Predstavili sme v rámci neho ukážku rôznych aktivít a úloh s nasledujúcimi robotmi a hrami, ako Phiro Pro Unplugged, CodeyBot, Scottie Go!, BBC micro:bit s joystickom, ako aj App Inventor na tabletoch. Workshop sa realizoval formou individuálnych konzultácií pri samostatných stanovištiach, kde boli pripravené jednotlivé aktivity. Pri každom stanovišti bol lektor/študent z informatického krúžku, ktorý názorne predstavil zadanie aj riešenie úlohy a delil sa o svoje skúsenosti z výučby. Niektoré aktivity sme prezentovali spolu so žiakmi z informatického krúžku pre žiakov 2. stupňa ZŠ. Celkovo sa workshopu zúčastnilo 31 učiteľov informatiky zo všetkých typov škôl. Veľmi nás tešia ich pozitívne ohlasy.

4 TECHNICKÉ ZABEZPEČENIE PROJEKTU

V rámci projektu sa zakúpilo na Katedre informatiky FPV UMB viacero didaktických pomôcok, ktoré sa využívali na informatickom krúžku pre žiakov ZŠ a tiež na workshopoch, ktoré sa vďaka projektu zorganizovali.

- Phiro Pro je didaktická pomôcka – programovateľný robot, ktorý sa môže programovať pomocou blokov kódu cez počítač, cez špeciálnu mobilnú aplikáciu pre OS Android, pomocou špeciálnych kartičiek na programovanie.
- BBC micro:bit je edukačná doska, ktorá obsahuje 25 led diód, dve programovateľné tlačidlá, kompas, akcelerometer, senzor intenzity svetla, Bluetooth, micro USB port a vstupno-výstupné piny. BBC micro:bit je možné programovať v blokoch kódu, v programovacom jazyku JavaScript a v jazyku MicroPython.
- mBot Ranger je robot, určený pre STEM metódu, teda žiaci si robota môžu postaviť aj sami a programovať ho v blokovom prostredí od firmy Makeblock.

- Stolová hra Scottie Go! obsahuje fyzické skladacie blokové príkazy, ktoré sa podobajú na štýl puzzle. Úloha v tvare hracej plochy sa objaví na tablete, príkazy si žiak vlastnoručne poskladá napr. na stole, tabletom zoskenuje kód a na základe toho sa mu vykoná program v aplikácii.
- Autíčko pre edukačnú dosku BBC micro:bit je možné programovať v prostredí Microsoft MakeCode. V prípade prepojenia s programovateľným herným ovládačom – gamepad je možné naprogramovať ovládanie autíčka na diaľku. Komunikácia medzi ovládačom a autíčkom prebieha pomocou bezdrôtovej komunikácie.
- Arduino uno mikrokontrolér je určený na vzdelávacie účely a tiež aj na menšie projekty v rámci IoT (Internet of Things).

4.1 Skúsenosti s využívaním vzdelávacích pomôcok

Všetky spomenuté pomôcky boli pre žiakov zaujímavé a osvedčili sa nám nielen v rámci informatického krúžku. Väčšinou sme používali blokový programovací jazyk Blockly, takže žiaci sa ľahko a rýchlo naučili ovládať jednotlivé pomôcky. Najväčšou prednosťou robota Phiro Pro je viacero spôsobov programovania, ktoré je možné prispôbiť veku a skúsenostiam žiakov (programovanie pomocou kláves na autíčko, SWISH kartičiek, jazyka Scratch, Snap4Arduino, či Pocket Code). Pri robotovi sme si ale všimli, že programovanie pomocou špeciálnych kartičiek (hlavne pri kartičkách s cyklami) nie je vždy presné a spoľahlivé.

Programovanie pomocou edukačnej dosky BBC micro:bit bolo veľmi zaujímavé pre žiakov, hlavne pre možnosť použitia rôznych rozšírení a teda žiaci mohli s touto edukačnou pomôckou robiť stále niečo nové, čo podnecovalo ich kreativitu.

Stolová hra Scottie Go! je vhodná pre samostatnú prácu žiakov. Aplikácia je vyladená a intuitívna, takže žiaci väčšinou vedia, ako majú postupovať, resp. kde urobili chyby.

Mikrokontrolér Arduino UNO je vhodný skôr na stredné školy, kedy žiaci už majú znalosti z programovania a hlavne zo zapájania káblov. V našom prípade to však zvládli aj šikovní žiaci zo základných škôl, ktorí majú vďaka informatickému krúžku určitý náskok.

5 VÝSTUPY PROJEKTU

V rokoch 2017 až 2019 bolo na katedre informatiky FPV UMB v rámci projektu vytvorených a odpublikovaných 21 prác na domácich a zahraničných konferenciách. Ďalej bolo vytvorených a overených 31 metodík a výučbových materiálov. Z toho je 26 určených pre vyučovanie informatiky na základných školách. Pričom 13 je cielených pre 5. a 6. ročník ZŠ a ďalších 13 pre 7. až 9. ročník ZŠ. 5 metodík a materiálov je určených pre stredné školy. Metodiky sú zamerané na rôzne tematické celky, predovšetkým na výpočtové myslenie, algoritmizáciu a programovanie.

Výstupy z projektu vo forme metodík a učebných materiálov sú dnes verejne dostupné v kurze InoMeto na LMS Moodle [8], zohľadňujú kritériá kladené na e-learningový kurz [5], takže ich môžu začať využívať učitelia informatiky na celom Slovensku.

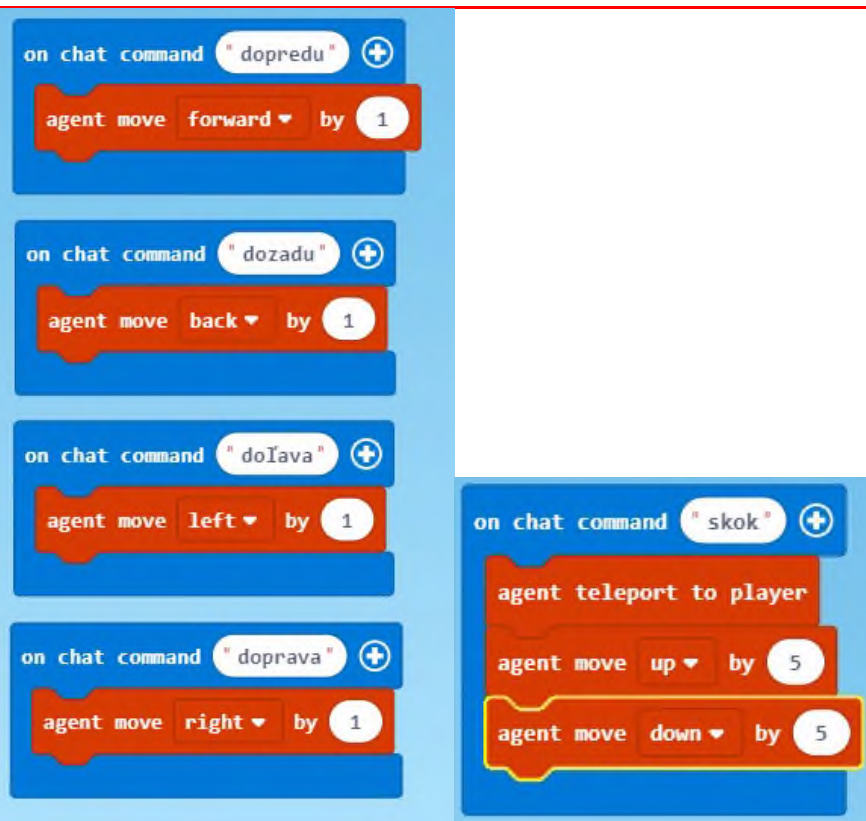
5.1 Ukážka jednej z metodík

Vzhľadom na to, že všetky vytvorené metodiky a učebné materiály sú k dispozícii v spomínanom kurze, dovolili sme si pre ilustráciu poukázať len na jednu, ktorá v dnešnej dobe zvýšeného záujmu o svet Minecraft veľmi rezonuje medzi deťmi a mládežou. Používanie Minecraft:Education Edition vo vyučovacom procese je veľmi aktuálna téma aj vzhľadom na prerušenie výučby počas pandémie Covid-19. Pomocou tohto prostredia môže učiteľ vyučovať na diaľku a byť v jednom svete so svojimi žiakmi.

Metodika – Programovanie vo svete Minecraft

Názov aktivity:	Minecraft: Education Edition – programovanie
Tematický celok:	Minecraft: Education Edition – programovanie

Cieľová skupina:	6.–9. ročník
Čas:	2 vyučovacie hodiny
Ciele:	Zoznámiť sa s nástrojom Minecraft: Education Edition a naučiť sa v ňom vykonávať úlohy na blokové programovanie
Metódy a formy:	<ul style="list-style-type: none"> • metódy bádateľské a výskumné, • metódy samostatnej práce žiakov, • metódy precvičovania a zdokonaľovania zručností, • práca žiakov – individuálna, frontálna (s učiteľom).
Príprava, učebné pomôcky:	PC s prístupom na internet, program Minecraft: Education Edition
Priebeh aktivity	<p>Úloha 1 <i>Vytvor si nový svet.</i> Vytvorenie nového sveta – Žiaci objavujú edukačné možnosti softvéru. Učiteľ vysvetlí jednotlivé funkcie softvéru po poradí. Pre možnosť programovania je potrebné stlačiť kláves „C“.</p> <p>Úloha 1.1 <i>Naprogramuj pohyb agenta.</i> Teleportujte agenta ku hráčovi (na výber máte dve možnosti – zavolanie on chat command príkazu, alebo po volaní príkazu „dopredu“) Naprogramujte ovládanie agenta pomocou príkazov on chat command:</p> <ul style="list-style-type: none"> ○ pohni o 1 dopredu („dopredu“) ○ pohni o 1 dozadu („dozadu“) ○ pohni o 1 doľava („doľava“) ○ pohni o 1 doprava („doprava“) ○ agent vyskočí o 5 pozícií vyššie a následne sa postupne vráti na svoju pôvodnú pozíciu <p>Žiaci samostatne vypracúvajú znenie úlohy. Počas každej úlohy sa žiaci snažia prísť na riešenie sami. Učiteľ ich navádza na správne riešenie slovným vysvetlením logickej postupnosti krokov, nie ukazovaním na správne bloky príkazov.</p> <p>Príkazy sa vykonávajú po zavolaní názvu príkazu v konzole. Konzola sa spustí pomocou stlačenia klávesu „T“.</p>

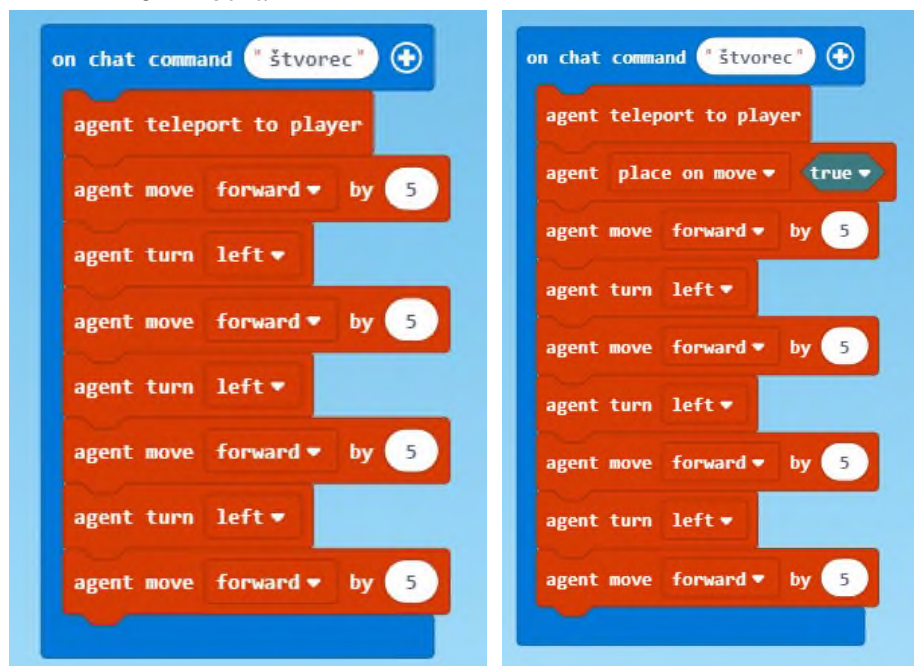


Úloha 2

Vykresli matematické útvary agentom.

Vykreslite nasledujúce geometrické útvary agentom:

- štvorec 5x5
- obdĺžnik 5x6
- trojuholník rovnostranný
- trojuholník pravouhlý
- kocka



Možnosti rozšírenia:

Upravte predchádzajúce príklady tak, aby agent pri pohybe položil nejaký objekt (napr. TNT). Upozornenie, agentovi musíte najskôr priradiť do inventára objekty (prídete k agentovi a stlačíte pravé tlačidlo myši).

Úloha 2.1

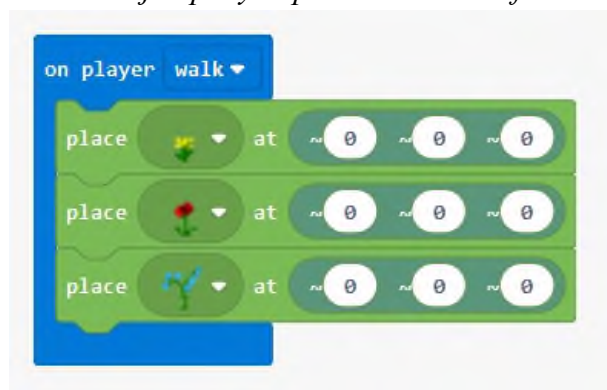
Keď už vieš, ako vykresľovať útvary (úloha č. 2), tak vykresli agentom aj nejaké texty:

Pomocou agenta a programovania vykreslite na obrazovke nasledujúci text:

- IT
- KI
- iniciály Vášho mena

Úloha 3

Počas Tvojho pohybu po svete Minecraft sad' kvety.



Možnosti rozšírenia:

Upravte predchádzajúci príklad tak, aby agent najskôr postavil pomocou programovania ohradu (pridajte mu objekt na stavbu ohrady = plot), následne umiestnite agenta do vnútra dvora a prikážte mu nech sadi kvety (všetko pomocou programovania). Agent musí posadiť kvety na celej záhrade.

5.2 Diskusia

Predstavená metodika sa osvedčila, nie len pre zvýšený záujem žiakov o imaginárny svet Minecraft, ale hlavne pre jednoduchosť programovania, pestrosť a variabilitu úloh, ako aj silnú podporu knižnice ďalších prostredí a objektov. V porovnaní s inými metodikami na programovanie Minecraft je táto metodika vhodná ako doplnok pre vytvorené návody, ktoré sú dostupné na stránke <https://minecraft.makecode.com>.

Pri riešení úloh predstavených v tejto metodike nemali žiaci žiadny problém s nájdením riešení. Vo veľkej miere je to dôsledok online prostredia MakeCode, ktoré je rovnaké pre programovanie edukačnej dosky BBC micro:bit, Lego EV3, či vytváranie hier Arcade. Najväčšou nevýhodou tohto prostredia je, že nie je zadarmo. K dispozícii je iba obmedzený počet vstupov na každú licenciu Office365. V prípade, že je potrebné zakúpiť ďalšie licencie, ktoré aj keď nie sú veľmi drahé, musia sa naviazať na konkrétny účet Office365. Z pozorovania a rozhovorov so žiakmi a s pomocnými učiteľmi na krúžku (našimi študentami z katedry informatiky) môžeme konštatovať, že všetky úlohy boli pre žiakov veľmi zaujímavé a chceli by pokračovať v riešení podobných úloh.

6 ZÁVER

V tomto článku sme sa pokúsili prezentovať kľúčové informácie, ale najmä splnenie cieľov projektu KEGA 009KU-4/2017 Inovatívne metodiky v predmete informatika v sekundárnom vzdelávaní. Výstupy z projektu predstavujú predovšetkým metodiky, ktoré môžu učitelia využiť na hodinách informatiky. Metodiky sme overili, ale aj prezentovali, v rámci rôznych konferencií, publikácií, workshopov a webových stránok. Mnohé metodiky sme konzultovali so zahraničnými partnermi. Snažili sme sa, aby sa výstupy projektu mohli ľahšie dostať k cieľovej skupine a aby jednotlivé metodiky a učebné materiály mohlo začať využívať čo najviac učiteľov na základných a stredných školách. Preto sme výstupy zosumarizovali a umiestnili na portál, vo forme LMS kurzu InoMeto [8], ktorý je otvorený a odkiaľ je možné stiahnuť si všetky metodiky.

7 POĎAKOVANIE

Príspevok bol čiastočne finančne podporený projektom KEGA 018UMB-4/2020 Implementácia nových trendov v informatike do výučby algoritmického myslenia a programovania v predmete informatika v sekundárnom vzdelávaní a vznikol aj za podpory 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.

8 BIBLIOGRAFICKÉ ODKAZY

- [1] Inovovaný Štátny vzdelávací program [online]. [vid. 2. 1. 2020.]. Dostupné na <http://www.statpedu.sk/sk/svp/inovovany-statny-vzdelavaci-program/>.
- [2] KALAŠ, I. *Školská informatika, kde sme....a kam ďalej*. Bratislava: s.n., 2019. Informatický deň 2019.
- [3] Portal VS [online]. [vid. 2. 1. 2020]. Dostupné na <https://www.portalvs.sk/sk/>.
- [4] JACKOVÁ, J., RADOBICKÁ, M., KLIMOVÁ, N., IVANČÁKOVÁ, S., MICHALČÍKOVÁ, K. *Možnosti zvyšovania pedagogickej praxe u študentov*. Liberec: FP TUL, 2018. Didinfo 2018. ISBN 978-80-7494-424-6.
- [5] KLIMOVÁ, N. JACKOVÁ, J. a RADOBICKÁ, M. *Čo ak vám katedra informatiky umožní realizovať vaše sny?* Liberec: FP TUL, 2018. Zborník Didinfo 2018. ISBN 978-80-7494-424-6.
- [6] KLIMOVÁ, N., HORVÁTHOVÁ, D. *Nadštandardné úlohy z multimédií pre žiakov ZŠ*. Liberec: s.n., 2018. ISBN 978-80-7494-424-6.
- [7] KLIMOVÁ, N., HORVÁTHOVÁ, D. *Robot Phiro ožíva v rukách detí*. Liberec: s.n., 2018. Dostupné na <http://www.fpv.umb.sk/katedry/katedra-informatiky/konferencia-didinfo/elektronicke-zborniky.html>.
- [8] Inovatívne metodiky v predmete informatika v sekundárnom vzdelávaní [online]. UMB. [vid. 2. 1. 2020]. Dostupné na <https://lms.umb.sk/course/view.php?id=2013>.
- [9] Inovatívne metodiky v predmete informatika v sekundárnom vzdelávaní [online]. FPV UMB, 13. 1. 2020. [vid. 22. 4. 2020]. Dostupné na <https://lms.umb.sk/course/view.php?id=2013>.

Prezentace vzdělávacích materiálů pro Scratch 3.0

Presentation of Educational Materials for Scratch 3.0

Ingrid Nagyová
Slovenská technická univerzita
Ilkovičova 2
842 16 Bratislava
Slovensko
ingrid.nagyova@stuba.sk

ABSTRAKT

Scratch je vizuální programovací jazyk, který je využíván pro rozvoj algoritmického myšlení a výuku programování žáků základních škol. Od ledna 2019 je dostupná verze Scratch 3.0, která má moderní design optimalizovaný pro využití na dotykových zařízeních. Oproti předchozí verzi Scratch 2.0 používá vývojové prostředí větší (vyšší) bloky pastelových barev s bílými tenkými popisky. Nový design Scratch 3.0 odpovídá cílové skupině uživatelů. Jeho velkou nevýhodou je ztížená čitelnost vzdělávacích materiálů v Scratch 3.0 při prezentaci v tištěné podobě nebo prostřednictvím dataprojektoru.

Príspevok ukazuje možnosti prezentace výukových materiálů pro Scratch 3.0 s ohledem na dobrou viditelnost a čitelnost. Popisuje postupy možných úprav scénářů vytvářených v Scratch 3.0. Postupy byly verifikovány při převodu vzdělávacích materiálů na novou verzi Scratch 3.0. V závěru prezentujeme možnost využití popisované techniky tvorby scénářů v práci učitele na základní škole.

ABSTRACT

Scratch is a block-based visual programming language that is used for teaching of algorithmic thinking skills and programming for primary school pupils. Since January 2019, a version of Scratch 3.0 is available, which has a modern design optimized for use on touch devices. Compare with the previous version 2.0, the Scratch 3.0 development environment uses larger (higher) blocks of pastel colors with white labels. The new design of the environment corresponds to the target group of users. The great disadvantage of the new design is the worsened readability of educational materials Scratch 3.0 in printed form or by presentation through a projector.

The paper shows the possibilities of the presentation of educational materials for Scratch 3.0 with regard to their good visibility and legibility. It describes how to edit code blocks created in Scratch 3.0. The procedures were verified by the transfer of educational materials for the new version of Scratch 3.0. In the end, we present the possibility of using the described technique of creating scenarios in the work of a primary school teacher.

Klíčová slova

Scratch 3.0, scénář, výuka programování, vzdělávací materiál.

Keywords

Scratch 3.0, code blocks, teaching programming, educational material.

1 ÚVOD

V rámci projektu PRIM [1] je mimo jiné připravována učebnice programování v prostředí Scratch pro žáky 2. stupně základní školy. Učebnice byla původně vytvářena pro verzi Scratch 2.0. Materiály pro žáky jsou zpracovány formou žákovských listů, které mohou být promítány ve třídě přes dataprojektor nebo využity v tištěné verzi.

Učebnice je pilotně ověřována již více než dva roky od září 2017 [2]. Původně jsme předpokládali, že žáci budou s učebnicí pracovat podle možností školy buď v offline verzi instalované na žákovských počítačích, nebo v online verzi dostupné přes internet. V lednu 2018 byl oficiálně spuštěn náhled nové verze Scratch 3.0 [3] s dodatkem, že tato verze po pilotním ověření nahradí tehdy aktuální verzi 2.0. Vzhledem k připraveným materiálům pro verzi 2.0 bylo ověřování ve školním roce 2018/19 realizováno výhradně v offline verzi. Tato situace potvrdila nezbytnost zabývat se překlopením připravených materiálů na novou verzi Scratch 3.0. Tuto nezbytnost potvrdily i ohlasy některých učitelů reagující na zhoršenou viditelnost některých scénářů. Vzpomeňme například komentář učitele k první kapitole „Některé scénáře na žákovských listech jsou rozmazané a nejdou při promítání přečíst. (např.: snímek 19 scénář c).“

Problém se nakonec ukázal komplexnější. Nová verze Scratch 3.0 je optimalizována pro využití na dotykových zařízeních. Bloky jsou vyšší, aby je bylo možné ovládat dotykem. Vybarvené jsou světlými pastelovými barvami s bílými texty psanými tenkým nevýrazným písmem. Obrázek 1 srovnává stejné scénáře vytvořené ve Scratch 2.0 a 3.0 včetně velikosti scénářů – oba obrázky byly pořízeny stejným postupem.



Obrázek 1: Srovnání scénářů Scratch – verze 2.0 vlevo a verze 3.0 vpravo

Při řešení problému prezentace scénářů Scratch 3.0 bylo nutné zohlednit několik požadavků, které vytanuly s ohledem na nutnost promítat scénáře prostřednictvím dataprojektoru nebo je prezentovat žákům v tištěné podobě (buď barevné):

- Barevnost bloků a nevýraznost barev při prezentaci prostřednictvím dataprojektoru.
- Velikost, typ a řez písma na blocích.
- Výška bloků – nový scénář překrývá větší plochu a materiály musí být i graficky upravovány.

Problém prezentace scénářů v nové verzi Scratch jsme konzultovali s tvůrci projektu Scratch a Mitchem Resnickem, který je jedním z iniciátorů a protagonistů tohoto projektu. Ti vyjádřili porozumění pro náš problém, ale k jeho řešení momentálně nemají kapacity.

Příspěvek popisuje dva způsoby, které jsme pro prezentaci scénářů postupně vyvinuli. První počítá s grafickou úpravou rastrových obrázků scénářů, druhý umožňuje tvorbu vektorových obrázků scénářů. Oba způsoby řeší zejména problém dobré viditelnosti a čitelnosti scénářů Scratch 3.0 při jejich prezentaci prostřednictvím dataprojektoru nebo při barevném tisku.

2 GRAFICKÁ ÚPRAVA SCÉNÁŘŮ

Původní scénáře ve Scratch 2.0 jsme snímali z pracovní plochy, která byla dvakrát zvětšena pomocí nástroje lupa. Nasnímaný a ořezaný obrázek měl tak dostatečné rozlišení pro zobrazení přes dataprojektor nebo pro vtištění – cca 150 DPI. Kromě ořezu bylo nutné pouze odstranit z obrázků

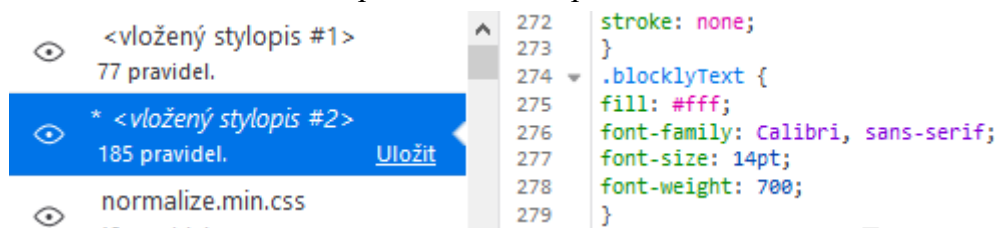
šedé pozadí vývojového prostředí Scratch. Podobné úpravy jsme předpokládali provést s obrázky pro novou verzi Scratch 3.0. Úpravy se pro další využití ukázaly nedostatečné, text na blocích byl nečitelný, barevné rozlišení bloků nebylo dostatečné.

2.1 Úprava textu

Zdálo se, že problém je snadno řešitelný změnou typu a řezu písma na blocích. Tu lze provést ve webovém prohlížeči, který umožňuje editovat styly použité na stránce. Osvědčil se prohlížeč Firefox, který má podporu a nástroje pro vývojáře a styly stránky lze editovat ve spodní polovině stránky (volba Nástroje pro vývojáře → Editor stylů).

Styl pro text bloků se nachází ve složce <vložený styl> #2 pod označením .blocklyText (řádek 274). Text bloků je standardně vypisován fontem Helvetica Neue, barvou #fff (bílá), velikosti 12 bodů a s tloušťkou 500. Snadno lze zjistit, že změna těchto parametrů způsobí změny ve vzhledu písma na blocích. Bohužel, změna velikosti písma je omezena programovou konstrukcí bloků: skutečná délka vypisovaného textu sice dynamicky ovlivňuje šířku bloku, výška textu na blocích je ale svázána s výškou bloků a nelze ji změnit. Výška textu tak přímo ovlivňuje jeho velikost, viditelnost textu lze proto ovlivnit hlavně výběrem vhodného fontu písma.

Po diskuzích v autorském týmu učebnice a konzultaci s učiteli lze pro text bloků doporučit font Calibri velikosti 14 bodů, s tloušťkou 700 – způsob nastavení parametrů lze vidět na obrázku 2.



Obrázek 2: Nastavení typu písma v prostředí Scratch 3.0

Po nastavení editoru stylů v prohlížeči lze snímat upravené obrázky scénářů opakovaně až do doby restartu webového prohlížeče.

Zvětšení a zvýraznění popisků bloků je nutnou podmínkou pro lepší čitelnost textů na blocích. Bloky přesto zůstávají barevně nevýrazné a těžko rozlišitelné, bílé písmo na blocích je přes dataprojektor špatně čitelné. Změna barvy písma na blocích (například na černou) se ukázala nevhodnou z důvodu výrazné změny ve vnímání jednotlivých bloků a zhoršení orientace v prostředí Scratch, kterou potvrdili na dotaz i učitelé ze základních škol. Zaměřili jsme se proto na zvýšení kontrastu barev a úpravu barevnosti bloků.

2.2 Barevnost bloků

Barevné rozlišení bloků je důležitým orientačním prvkem prostředí Scratch. Proto, pokud zvažujeme úpravy barevnosti bloků, je nutné myslet na zachování vzájemného barevného poměru mezi barvami bloků jednotlivých kategorií.

Zvýšení kontrastu nebo jasů obrázku mění barvy a jejich vzájemný barevný poměr. Vhodným nástrojem pro úpravu barev bloků se ukazuje nastavení úrovně v histogramu a změna tónového rozsahu. Tyto změny umožňují provádět i jednoduché grafické editory. Doporučit lze například FastStone Image Viewer, který umožňuje změny provádět i hromadně pro větší množství obrázků.

Pro prezentaci obrázků scénářů se nám osvědčilo upravit rozsah vstupní úrovně histogramu RGB v rozsahu 80 až 255 (místo standardního rozsahu 0 až 255). Obrázek se stane kontrastnějším, lépe viditelným a čitelným.

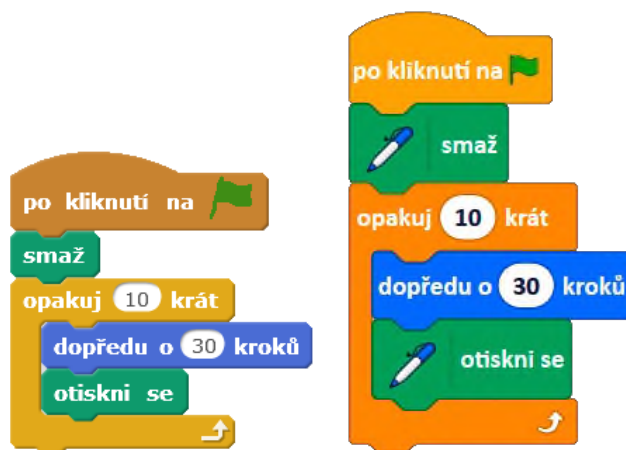
2.3 Tvorba rastrových obrázků pro prezentaci

Jak bylo naznačeno, rastrový obrázek scénáře vytvořeného v Scratch 3.0 vhodný pro prezentaci ve výukových materiálech lze vytvořit následujícím způsobem:

1. Nastavit styl textu bloků v editoru stylů ve vhodném webovém prohlížeči (Firefox), doporučujeme font Calibri, velikosti 14 bodů, tloušťka 700.
2. Zvětšit zobrazení v pracovní ploše Scratch pomocí lupy (dvakrát).
3. Nasnímat scénář, ořezat jej a odstranit šedé pozadí.
4. Upravit barevnost obrázku nastavením vstupní úrovně histogramu RGB v rozsahu 80 až 255.

Práce na úpravě rastrových obrázků pro jejich dobrou prezentaci ve výukových materiálech je náročná. Barevnost obrázků a jejich čitelnost ani po naznačených úpravách nedosahuje úrovně scénářů předchozí verze Scratch 2.0 – viz porovnání na obrázku 3. Přitom při srovnání prostředí Scratch 2.0 a 3.0 se projeví dvě zásadní změny ve vizuálním vzhledu a barevnosti obou prostředí:

- změna vzhledu (tvaru) a barev vlastních (uživatelé definovaných) bloků z fialové na růžovou (viz obrázek 1);
- prohození barevné škály mezi kategoriemi Události a Ovládání, záměna světle a tmavě okrové barvy (viz obrázek 3, bloky po kliknutí na a opakuj 10 krát).



Obrázek 3: Srovnání obrázků scénářů Scratch 2.0 a upravených obrázků Scratch 3.0

Převádět dobře viditelné a čitelné obrázky Scratch 2.0 na hůře viditelné a rozeznatelné obrázky ve Scratch 3.0 kvůli těmto dvěma změnám se nejevilo jednoznačně smysluplné. Při dalším hledání řešení problému se ukázala možnost využít vektorové obrázky a navíc proces přípravy obrázků alespoň částečně automatizovat.

3 VEKTOROVÉ OBRÁZKY SCÉNÁŘŮ

Prostředí Scratchblocks [4] umožňuje generovat obrázky bloků prostředí Scratch ve verzi 2.0 a 3.0. Je přímo navázáno na Scratch včetně lokalizace pro jednotlivé jazykové mutace. Vytvořené obrázky lze uložit v rastrovém formátu PNG nebo ve vektorovém formátu SVG. Prostředí je dostupné pod webovou službou GitHub [5] a jako takové poskytuje zdrojový kód projektu a umožňuje jeho úpravy a další využití.

V prostředí Scratchblocks nás zaujala možnost tvorby vektorových obrázků. Jejich výhodou je možnost prezentace při jakémkoliv zvětšení bez ztráty kvality obrazu. Rozhodli jsme se této možnosti využít. Při přípravě prostředí pro tvorbu obrázků bylo potřeba vyřešit několik problémů, které vyžadovaly zásah do zdrojového kódu prostředí Scratchblocks.

3.1 Úprava zdrojového kódu prostředí Scratchblocks

Prostředí Scratchblocks je dostupné ve formě webového rozhraní. Hlavní část prostředí je naprogramována v JavaScriptu. Po instalaci prostředí do počítače lze zdrojové kódy včetně jazykové lokalizace najít ve složce `build`, soubor `scratchblocks.min.js`. Zde byly nutné následující úpravy:

- Úprava barevností bloků a jejich součástí (rámeček bloků apod.). Využili jsme barev, které jednotlivé bloky ve scénářích získaly úpravou naznačenou v části 2 (viz obrázek 3). Tyto barvy jsme pro celkový barevný dojem mírně upravili. Současne jsme podle předchozích pravidel upravili font písma na blocích (font Calibri, velikosti 14 bodů, tloušťka 700). Barvy bloků a textů jsou definovány na řádku 4345 zdrojového souboru a lze je měnit.
- Koncept vlastních bloků (kategorie Moje bloky) využívá v angličtině jednoslovné pojmenování `define`. Stejně je tomu ve slovenštině – název `blok`. V češtině je použito dvouslovné pojmenování scénář `pro`. Ukázalo se, že pro definici bloků je podmínka jednoslovného pojmenování důležitá. Slovo `define` muselo být ve zdrojovém souboru nahrazeno slovním spojením scénář `pro`, kde mezi slovy „scénář“ a „pro“ je mezera zapsaná kódem 255 (na klávesnici psáno `<Alt> + 255`), viz řádek 1349. Tímto způsobem je nutné psát text i při definici bloků a vytváření obrázků scénářů.

Upravené prostředí Scratchblocks je dostupné ke stažení na adrese <https://1url.cz/zM7ai>. Soubor je nutné dekomprimovat, následně je možné prostředí využít.

3.2 Tvorba vektorových obrázků v prostředí Scratchblocks

Postup přípravy obrázků scénářů v prostředí Scratchblocks vyžaduje zacvičení, jistou zručnost a pozornost při práci. Po spuštění prostředí (soubor `index.html`) ve webovém prohlížeči je nutné nejprve vybrat používaný jazyk. Náhled scénářů je vytvářen v okně vlevo (viz obrázek 4).



Obrázek 4: Prostředí Scratchblocks při tvorbě vektorových obrázků

Scénáře musí být psány na klávesnici (okno vpravo na obrázku 4), nelze využít jazyk ikon, který standardně prostředí Scratch využívá. Texty bloků odpovídají textům v prostředí Scratch, jazykově jsou obě prostředí stejná. Při psaní textů bloků jsou drobná specifika:

- Speciální symboly (praporek, šipka pro blok otoč apod.) se vykreslí po zadání `@` a anglického názvu daného symbolu – `@greenflag`, `@turnleft`, `@turnright`.
- Slovní spojení „scénář `pro`“ (pouze v češtině) je potřeba zapsat se znakem 255 místo mezery, tj. podržet `Alt` a napsat 255. Když je definovaný scénář pro daný blok (hlavička bloku), pak je tento blok vykreslován ve scénářích růžovou barvou, jak je potřeba.
- Pole pro zápis hodnoty (například `dopředu o (90) kroků`) se píše do kulatých závorek `()`. Pro zápis textu používáme hranaté závorky `[]`, například `bublina [Ahoj!]`.

- Pole podmínky do podmíněného příkazu se vytvoří pomocí $\langle \rangle$, například když $\langle \text{pom} = (20) \rangle$ tak. Prostředí automaticky pozná, že `pom` je proměnná a vykreslí ji oranžově.
- Příkaz může být ukončen textem `end`. To je důležité zejména pokud následuje více příkazů za sebou, například více cyklů nebo podmínek.
- Pole pro výběr ze seznamu hodnot (například po stisku klávesy...) se vkládá mezi hranaté závorky `[]` a místo malé bílé šipky se zapíše znak `v`, například po stisku klávesy `[a v]`.
- Barva se zadává hexadecimálně, například `#FF0000`.

Příklad 1:

Text v češtině a ve slovenštině pro jednoduchý scénář z obrázku 3 musí být zapsán následovně:

po kliknutí na @greenflag	pri kliknutí na @greenflag
smaž	zmaž
opakuj (10) krát	opakuj (10)
dopředu o (30) kroků	dopredu (30)
otiskni se	opečiatkuj
end	end

Obrázek scénáře je možné exportovat ve formátu SVG nebo PNG. Obrázek je uložen podle nastavení webového prohlížeče pod jménem `scratchblocks`, jméno nelze změnit.

3.3 Úprava obrázků pro jejich využití při prezentaci

Pro prezentaci a tisk vytvořených scénářů používáme obvykle nástroje Microsoft Office. Ty nepodporují práci s vektorovým formátem SVG. Z vektorových formátů je podporován pouze formát EMF. Pro převod vektorových obrázků scénářů z formátu SVG do EMF lze doporučit open source grafický vektorový editor Inkscape. Jeho výhodou je i možnost dodatečné úpravy vektorových obrázků scénářů – ořez, úprava barvy či zvýraznění.

Problém s výškou bloků (bloky v prostředí Scratch 3.0 jsou vyšší) jsme řešili grafickou úpravou vzhledu jednotlivých snímků v prezentaci tak, aby vektorové obrázky měly na snímcích velikost 7–8 % původní velikosti generované prostředím Scratchblocks. Tato velikost se při promítnutí přes dataprojektor ukázala jako dostatečná.

Na obrázku 5 lze vidět jeden z žakovských listů Učebnice programování pro 2. stupeň ZŠ ve Scratch [2] s vektorovými obrázky scénářů vytvořenými podle uvedeného postupu.



Obrázek 5: Žakovský list doplněný vektorovými obrázky scénářů

Tvorba vektorových obrázků pro prezentaci ve výukových materiálech učebnice nebyla jednoduchá a vyžádala si poměrně dlouhý čas. Získali jsme materiál, který z grafického hlediska na vysoké úrovni je dobře čitelný a přehledný. To potvrzuje i testování učebnice v tomto školním roce, které probíhá prozatím bez připomínek ke vzhledu žakovských listů.

Největším problémem při práci s vektorovými obrázky se ukázala skutečnost, že prostředí Scratchblocks žádným způsobem nekontroluje korektnost vytvářených bloků a jejich grafický vzhled. Nejvíce chyb způsobují závorky, pro jejichž použití neexistují ustálená pravidla – viz příklad 2. Scénáře pro podobné úlohy jsme často kvůli drobným chybám z nepozornosti byli nuceni vytvářet opakovaně.

Příklad 2:

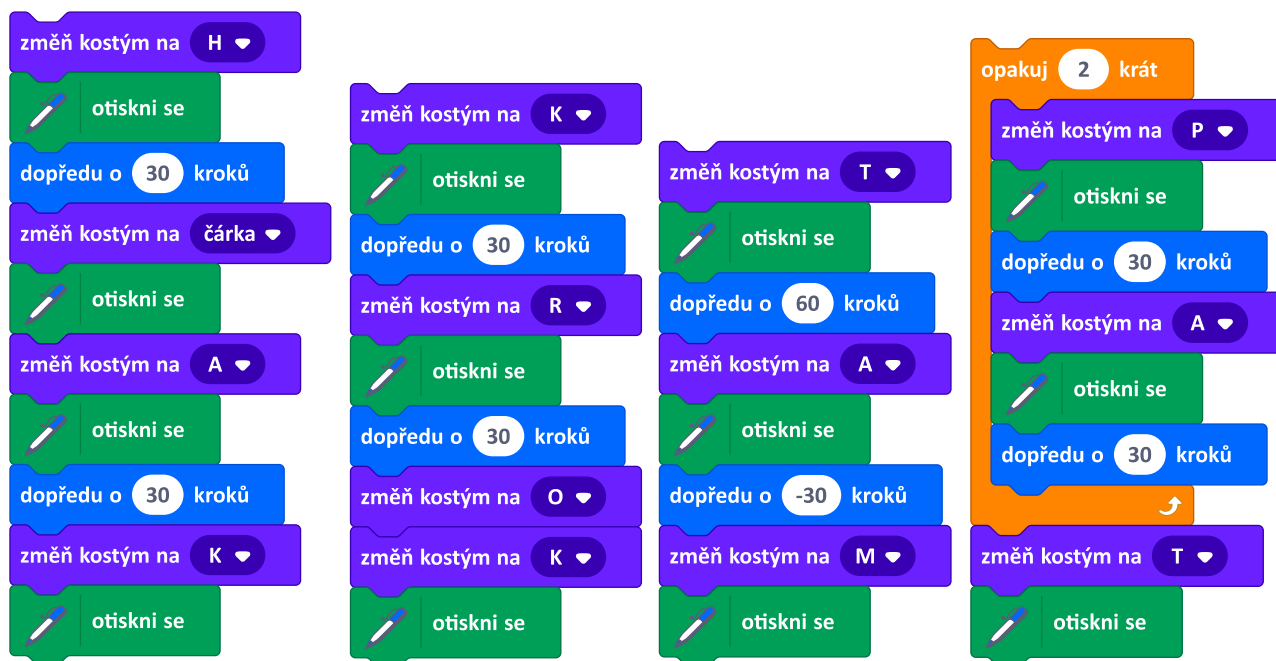
Text s odlišnými závorkami při zasílání (kulaté závorky) a přijímání zpráv (hranaté závorky):

```
po obdržení zprávy [zpráva 1 v]
bublina [Já jsem Honza.] (2) sekund
čekej (1) sekund
vyšli zprávu (zpráva 1 v)
```



4 VYUŽITÍ PŘI PRÁCI UČITELE

Přestože tvorba vektorových obrázků není jednoduchá, domníváme se, že může najít uplatnění v práci učitele. Jednou z možností je využití prostředí Scratchblocks při tvorbě návodů a dalších výukových materiálů. Daleko větší uplatnění lze očekávat při přípravě diskuzních problémových úloh, které se nám velice osvědčily v učebnici. Při jejich řešení žáci diskutují nad zobrazenými scénáři, snaží se zjistit, co se po spuštění scénáře stane, aniž by scénář skutečně vytvořili, a argumentují své domněnky před spolužáky. Scénáře pro zadání podobných úloh dokáže učitel v prostředí Scratchblocks po krátkém zacvičení vytvořit a následně použít. Na obrázku 6 je zobrazeno několik úloh, v nichž mají žáci za úkol určit, co se vypíše.



Obrázek 6: Příklady diskuzních problémových úloh pro žáky

Úlohy z obrázku 6 lze modifikovat změnou zadání a tak vytvořené scénáře použít ve třídě opakovaně. Například můžeme přemýšlet, jak upravit druhý scénář (obrázek 6) tak, aby se vypsal slovo KROK nebo jak upravit třetí scénář aby se vypsal slovo TAM nebo MAT. Čtvrtý scénář lze upravit tak, aby se vypsal slovo SPAPAT, PAPAYA nebo PAPARAZZI.

Výhodu předem vytvořeného scénáře v prostředí Scratchblocks shledáváme v těch případech, kdy by tvorba scénáře v hodině před žáky trvala dlouho nebo by mohla navádět žáky k správnému řešení, upozornit je na záměrné chyby ve scénáři apod. Výhodou je rovněž možnost opakovaného použití vytvořených scénářů, například při změně zadání úlohy.

5 ZÁVĚREM

Při přípravě materiálů pro Učebnici programování pro 2. stupeň ZŠ ve Scratch [2] jsme řešili problém převodu obrázků z verze Scratch 2.0 na novou verzi Scratch 3.0. Hlavním problémem se ukázal vzhled nové verze, který je optimalizovaný pro použití na dotykových zařízeních. Bloky jsou vyšší, ve světlých pastelových barvách s bílými tenkými popisky. Při prezentaci práce v prostředí Scratch 3.0 jsou bloky barevně nevýrazné a texty na nich nečitelné.

V první fázi hledání řešení uvedeného problému jsme se zaměřili na rastrové obrázky a vyvinuli jsme postup jejich úpravy tak, aby tyto byly při prezentaci přes dataprojektor barevně výraznější a čitelnější. Postup popisujeme v kapitole 2.

Při dalším hledání řešení problému se ukázala možnost konvertovat obrázky scénářů do vektorové podoby. Vhodné nastavení prostředí Scratchblocks, které umožňuje generovat vektorové obrázky blokových scénářů prostředí Scratch, umožnilo tvorbu upravených obrázků textovým přepisem scénářů do prostředí Scratchblocks a následným ořezem a převodem vektorových obrázků do formátu EMF v grafickém editoru Inkscape. Upravené prostředí Scratchblocks je dostupné ke stažení na adrese <https://1url.cz/zM7ai>. Příspěvek přináší podrobný návod na tvorbu vektorových obrázků pro Scratch 3.0 (viz kapitola 3) a na možnost jejich využití v práci učitele na základní škole (viz kapitola 4). Upravené výukové materiály jsou dle vyjádření učitelů barevně přijatelné a dostatečně čitelné.

Vyžití vektorových obrázků a prostředí Scratchblocks nabízí možnost prezentace výukových materiálů pro Scratch 3.0 na webu. Do budoucna proto zvažujeme také možnost převodu metodických listů do podoby webových stránek.

6 PODĚKOVÁNÍ

Vývoj učebnice byl realizován z prostředků projektu CZ.02.3.68/0.0/0.0/16_036/0005322 PRIM – Podpora rozvíjení informatického myšlení.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] *Web projektu PRIM*. [online] <http://www.imysleni.cz>. Jihočeská univerzita, 2017.
- [2] VANÍČEK, Jiří, NAGYOVÁ, Ingrid a Monika TOMČÁNYIOVÁ. Učebnice programování pro 2. stupeň ZŠ ve Scratchi, její teoretická východiska a charakter. In: *Recenzovaný sborník konference DIDINFO 2018*. Liberec: TU v Liberci, 2018, s. 25–32. ISBN 978-80-7494-424-6.
- [3] *Web prostředí Scratch 3.0*. [online] <https://scratch.mit.edu>. MIT Media Lab, 2019.
- [4] *Scratchblocks v3.4 by blob8108*. [online] <https://scratchblocks.github.io>. MIT Media Lab, 2018.
- [5] RADVAN Timothy. *Scratchblocks na GitHub*. [online] <https://github.com/scratchblocks/scratchblocks>.

Robotika ve vzdělávání

Robotics in Education

Online cloudové prostředí pro výuku informatiky a robotiky na ZŠ

The Online Cloud Environments for Teaching Computer Science and Robotics at Elementary School

P. Mališ*, P. Šaloun*, L. Bryndová*, P. Hruža**, T. Dragon*, M. Klement*

*Katedra technické a informační výchovy, Pedagogická fakulta, Univerzita Palackého v Olomouci,
Žižkovo nám. 5, 771 40 Olomouc, Česká republika

**VAHAL, s.r.o., Vrahovická 56c, 796 01 Prostějov, Česká republika

petr.malis@upol.cz, petr.saloun@upol.cz, lucie.bryndova01@upol.cz, phruza@vahal.cz,
tomas.dragon@upol.cz, milan.klement@upol.cz

ABSTRAKT

Příspěvek pojednává o praktických zkušenostech učitelů s teoretickou přípravou studentů na výuku robotiky a o prerekvizitách, které studenti musí splňovat pro úspěšnou participaci ve výuce robotiky. Základním tématem je přehled prostředí, které učitelé mohou využívat k výuce programování a porovnání těchto prostředí s ohledem na návaznost vlastní výukou programovatelné robotiky. Příspěvek zároveň nabízí příklady možnosti implementace edukační robotiky, využívající teoretických přístupů konstruktivismu a konstrukcionismu do koncepce výuky technických, technologických a informatických předmětů.

ABSTRACT

The paper deals with practical experiences of teachers with the theoretical preparation of students in the robotics education and with the prerequisites that students must fulfill for a successful participation in the robotics classes. The main topic is an overview of the environments that teachers can use to teach programming and a comparison of these environments with regards to the planned progression of their classes to using programmable robotic tools. The paper also offers examples of implementation of educational robotics, using theoretical approaches of constructivism and constructionism to the concept of teaching technical, technological and computer science subjects.

Klíčová slova

Výuka programování, edukační robotika, blokové programovací jazyky

Keywords

Programming in education, educational robotics, block programming languages

1 ÚVOD

V současné době více než dvacet evropských zemí, včetně Česka, implementovalo vzdělávání v oblasti programování do národních kurikul, případně plánuje takto učinit v perspektivě následujících let [1]. V Česku reaguje na potřebu revize RVP v oblasti informatiky a ICT *Strategie vzdělávací politiky České republiky do roku 2020*, která navrhuje změny kurikula z ICT pro střední a základní školy v souladu s moderními požadavky na digitální gramotnost, které doporučuje Evropský parlament a Rada [2].

Nové rámcové vzdělávací programy pro informatiku se blíže zaměří na výuku algoritmizace, programování a základů robotiky a do škol mají být plošně implementovány v roce 2020 [3]. Nově

zavedená témata pomohou žákům si osvojit a rozvíjet informatické myšlení. Tento soubor kompetencí a schopností by měl usnadnit žákům řešení problémů za pomoci využití algoritmů, a také zvýšit možnost uplatnění na trhu práce, který se postupně transformuje a zejména v technickém odvětví se očekává větší míra digitalizace a využití kyberneticko-fyzikálních systémů. Tento vývojový trend je označován jako Průmysl 4.0. Druhý nově implementovaný tematický celek základy robotiky propojuje poznatky nejen z informatiky (programování), ale také z techniky (návrh a konstruování), u některých programovatelných stavebnic (Arduino) je potřeba znát i základy elektrotechniky.

Se zavedením nových témat do výuky blízce souvisí volby adekvátních učebních prostředí, pomůcek a adekvátního propedeutického programovacího jazyka. V současnosti probíhají diskuze o způsobu implementace programování do škol a možnostech výběru programovatelných výukových platforem, které se pro tuto implementaci zvažují. V rámci této implementace se jako primární zdroj nabízí webová stránka celorepublikového projektu *PRIM* („Podpora rozvíjení informatického myšlení“) [4], v jehož rámci vznikají učebnice a podpůrné materiály pro výuku nové koncepce informatiky. Pro potřeby projektu jsme se zaměřili konkrétně na vzdělávací materiály a potřebné pomůcky/zařízení vztahující se k oblastem: „Programování a algoritmizace“ a „Základy robotiky“.

V článku se zaměřujeme na problematiku programování robotů pomocí vizuálních programovacích jazyků. Tento způsob programování můžeme označit i jako za aktuální trend v oblasti výuky programování. Nabízí totiž začátečníkům jednoduchý a přehledný způsob pro seznámení se s principy programování. S vizuálním programováním se také můžeme setkat v didaktických hrách pro výuku programování a algoritmizace.

2 VÝUKA ROBOTIKY

Jedním z významných trendů současné modernizace výuky na základních a středních školách je využívání specializované edukační robotiky jako výukové platformy pro výuku programování, algoritmizace a dalších kompetencí, jako je rozvoj informatického myšlení a digitální gramotnosti. Využívání programovatelných stavebnic je jedním ze způsobů jednoduchého rozvoje digitálních kompetencí u žáků, včetně jejich kritického a zodpovědného přístupu k digitálním technologiím a pochopení koncepce digitální společnosti.

Pro výuku robotiky mohou učitelé v současnosti volit z širokého výběru specializovaných programovatelných hraček a stavebnic, které jsou specificky vytvořeny pro žáky odpovídající věkové kategorie a účelně rozvíjejí jejich klíčové kompetence a schopnost informatického myšlení a z nichž většina byla již úspěšně ověřena v zahraničí.

V článku se zaměřujeme na problematiku programování pomocí blokových příkazů. Tento způsob programování podporují stavebnice a roboti, kteří jsou ve většině případů doporučeni pro věkovou kategorii dětí ekvivalentní 2. stupni základní školy a výš. V tabulce 1 uvádíme vybrané příklady robotických stavebnic, se kterými máme osobní zkušenost v praxi a na které se budeme dále odkazovat.

Tabulka 1: Podporované programovací jazyky u vybraných robotických stavebnic

Robotická stavebnice	Programovatelné pomoci:
LEGO Mindstorms ev3	Ikonický jazyk RCX Code, C/C++, Java, Visual Basic, Scratch
LEGO WeDo 2.0	Vlastní ikonický jazyk, Scratch
Ozobot EVO, Bit	Blokový jazyk
Arduino	C/C++, Scratch
mBot	Prostředí mBlock, C++
BBC micro:bit	MakeCode blokový jazyk, Scratch, Python, JavaScript

2.1 Robotické stavebnice a programovatelní roboti

2.1.1 LEGO WeDo

Při práci s touto stavebnicí se rozvíjí nejen algoritmické myšlení, ale také kreativita a konstruktérské schopnosti. Programuje se pomocí ikonického jazyka, ale podporuje i Scratch.

2.1.2 LEGO Mindstorms

LEGO Mindstorms je série programovatelných stavebnic navazujících na LEGO WeDo, která je určena pro zkušenější studenty. LEGO Mindstorms je obtížnější, ale nabízí širší možnosti konstruování i programování. Před samotnou prací s robotickou stavebnicí LEGO Mindstorms Education EV3 se doporučuje předchozí zkušenost s vizuálním programovacím jazykem Scratch.

Podle průzkumu České republiky používá v současnosti nejvíce škol právě produkty z řad LEGO, jmenovitě LEGO Mindstorms a LEGO NXT [5, s. 5–14] a tedy i adekvátní programovací prostředí. Jde o plně programovatelné roboty, kteří nabízí jak mnoho různých možností sestavení, tak programování.

2.1.3 Ozobot

Jedná se o mini robota (interaktivní didaktickou hračku), který byl vytvořen v souladu s filozofií STEAM výuky. Podporuje kreativitu a výuku kódování. Lze využít jako podporu při rozvoji kreativity a logického myšlení u žáků. Na spodní části nalezneme senzory pěti barev, díky kterým rozpozná vodící linku a barevné kódy. Vyšší verze EVO disponuje i senzory přiblížení.

Programování je možné realizovat několika způsoby: pomocí editoru OzoBlockly dostupného přes libovolný webový prohlížeč, zakreslení barevných kódů, přes specializovanou bezplatnou mobilní aplikaci pro chytré telefony a tablety.

2.2 Programovatelná elektronika

2.2.1 BBC micro:bit

BBC micro:bit je mikropočítač o velikosti platební karty. Jedná se o open-source kit. Je primárně určen pro výuku informačních technologií. Programovat lze v několika jazycích: Blocks (vizuální programovací jazyk podobný jazyku Scratch), JavaScript a MicroPython.

2.2.2 Arduino

Arduino je jednodeskový počítač. Opět se jedná o open-source platformu. Nelze k němu přímo připojit klávesnici, myš, či monitor, ale lze jej rozšířit o další hardwarové komponenty jako display, servomotory, různé senzory apod. Přímo na webu Arduino lze najít spoustu užitečných tipů a námětů pro práci s tímto mikropočítačem. Tato platforma je cenově dostupná, a také vhodná i pro začínající uživatele v oblasti programování a robotiky. Programování probíhá přes specializovanou mobilní aplikaci Arduino Web Editor nebo desktopovou aplikaci Arduino IDE. Na desce Arduino je pak i založen další z robotů mBot.

3 VÝUKA PROGRAMOVÁNÍ

V případě volby jazyka pro účely efektivní výuky programování a konceptům, které se k němu vážou, zvažujeme primárně propedeutické programovací jazyky. Současný tuzemský systém volby programovacího jazyka pro výuku není standardizován a školy mají možnost volného výběru adekvátního prostředí dle individuální potřeby školy, kvalifikace jejích pedagogů a potřeb studentů.

Přesto, že neexistují oficiální zásady, podle kterých programovací jazyk pro výuku volit, platí obecné zásady, vycházející z letitých zahraničních i tuzemských zkušeností.

Při výběru vhodného propedeutického programovacího jazyka by měla být zvažována vhodnost a přehlednost vývojového prostředí, jeho uživatelská přístupnost a náročnost s ohledem na věk žáka. Mnoho učitelů programujících na nižším stupni základních škol se přiklání spíše k aplikacím, které je možné použít na tabletech nežli k programovacím prostředím, které nutně vyžadují myš a klávesnici. Práce s tabletem, nebo podobnými dotykovými zařízeními lépe rozvíjí dětskou motoriku a nevyžaduje tolik vstupní koordinace.

Dalším základním pravidlem výběru, na kterém se mnoho autorů dále shoduje, je, že syntaxe jazyka by pro žáky základních škol neměla být příliš složitá. Problematické se zdá například i používání technického žargonu [6], je tedy nutné zvážit i jazykovou strukturu programovacího prostředí. To by ovšem mělo nabízet dostatek prostoru k rozvoji žáka, aby se usnadnil pozdější přechod studenta ke komplexnímu programovacímu jazyku. V českém prostředí je pochopitelně vhodnější volit prostředí, které funguje na principu českého jazyka. Angličtina může být pro žáky základních škol velkou překážkou.

Podobně i v oblasti samotného programování a algoritmizace by vybraný propedeutický programovací jazyk měl primárně efektivně podporovat další rozvoj žáka v oblasti informatiky a programování. Student měl rozvíjet minimálně základních pět oblastí informatického myšlení, které vychází ze standardu pro rozvoj informatického myšlení definovaných CSTA (*Computer Science Teachers Association*) a ISTE (*International Society for Technology in Education*). Vybrané programovací prostředí by mělo žákovi nabídnout minimálně možnosti rozvoje abstrakce, generalizace, dekompozice, algoritmizace a ladění [7]. Román-González [8], tvůrce CT-testu, který měří úroveň informatického myšlení, pak považuje za klíčové, aby studenti ovládali základní programovací koncepty jako smyčky, počet jejich opakování a přerušení, jednoduché podmínky if, komplexní podmínky if/else, podmínka while a jednoduché funkce.

3.1 Vizuální programování

Existuje mnoho programovacích prostředí, které byly specificky vytvořeny přímo pro žáky odpovídající věkové kategorie. Jde například o Scratch, Kodu, Lego We-Do a Etoys. [9, 10]. Tato programovací prostředí jsou většinou bloková a ověřená mnohaletou praxí s poměrně rozvinutou podporou. Jde o vizuální programovací jazyky, které fungují na principu „drag and drop“ = táhni a pusť. Uživatel má k dispozici nabídku bloků, ze kterých skládá části programu. Tudíž si nemusí pamatovat seznam dostupných příkazů. Barevné odlišení bloků podle kategorií a také různé typy spojů, které zabráňují nesmyslným spojením, zpřehledňují vizuální jazyky a také minimalizují výskyt syntaktických chyb. Tyto vlastnosti umožňují žákům se více zaměřit na řešení problémů, ale mohou taky přinést problémy s uchopením syntaxe u vyšších programovacích jazyků [11].

Je vhodné zmínit, že většina doporučovaných prostředí pro programování na základních školách je úzce svázána s použitím robotických stavebnic. Mladší žáci vyžadují vizualizaci problému a rychlou odezvu jejich práce, což je motivuje k další činnosti. Robotické stavebnice se v tomto ohledu jeví jako vhodné řešení výuky programování.

4 ONLINE NÁSTROJE PRO VÝUKU PROGRAMOVÁNÍ

Pro potřeby našeho projektu jsme zpracovali rešerši online nástrojů volně dostupných a použitelných ve výuce v oblasti programování a algoritmizace a robotických zařízení.

Online nástroje můžeme rozdělit na dvě kategorie, první prostředí nabízející volnost v tvorbě programu, mezi druhou skupinu zařadíme weby, které nabízejí didaktické hry pro výuku programování.

4.1 Prostředí pro volné programování

4.1.1 Scratch

Prvním z blokových programovacích jazyků vhodných k implementaci do výuky na českých základních školách je Scratch. Jedná se o vizuální programovací jazyk pro vývoj interaktivních her a multimédií vyvinutý na MIT. Je dostupný pro všechny prohlížeče s podporou HTML5, nebo si můžete stáhnout off-line studio. Obě prostředí jsou navzájem kompatibilní. Hlavní předností této platformy je překlad do 40 jazyků včetně češtiny a velká komunita uživatelů, se kterou lze sdílet vytvořené projekty. Díky tomu můžete nahlížet dovnitř kódů a zkoumat, jak byl vybraný program zkonstruován. Učitelé mají možnost založení účtu, který dovoluje vytvářet třídy a spravovat obsah pro žáky. Scratch také nabízí rozšíření, které lze využít pro programování Lego WeDo, Boost, Mindstorm nebo micro:bit. Na zařízení iOS a Android je k dispozici i verze využívající ikony pro mladší děti, Scratch Junior.

Scratch je založen na událostech (kód běží v reakci na akci uživatele; je zde „*click event*“ – kliknutí na zelenou vlajku). Události se používají v interaktivních programech, jako jsou grafické uživatelské rozhraní (GUI), webové stránky a hry, které reagují na vstup uživatele. Výstup mění vlastnosti jednoho nebo více postav (skřítků, víl atd.), jako je jejich poloha, orientace a velikost; nebo pozadí a zvuk.

4.1.2 Blockly

Blockly je vizuální programovací jazyk používaný několika systémy, včetně Code.org, Made with Code, MIT App Inventor a Grok Learning. Každý systém používá jinou sadu bloků, ale používají podobné bloky pro většinu kódovacích konceptů, včetně větvení a iterace. Každý systém běží na libovolném moderním prohlížeči pro stolní počítače a tablety, včetně iPadů.

V závislosti na systému může být vstup a výstup textový, grafický nebo jiný v závislosti na zařízení (například zařízení Android nebo BBC micro:bit). Mnoho systémů poskytuje indikátory, které automaticky kontrolují, zda je kód studenta správný.

4.2 Didaktické programovací hry

Společným rysem didaktických programovacích her je jejich zpracování pomocí jednotlivých úrovní. Ten umožňuje hráčům sledovat jejich postup a motivuje je k pokračování dál. S postupem ve hře se stávají jednotlivé úrovně těžší k vyřešení, zároveň hráč získává ale i nové možnosti, jak programovat. Postupně se tak učí využívat a sestavovat složitější algoritmy a funkce. Tyto didaktické hry podporují tvorbu vlastních znalostních konstrukcí, umožňují vlastní tempo procházení [12].

Asi největším portálem, co do počtu výukových úrovní, je projekt code.org, který se snaží o dostupné rozšíření výuky programování pro školy a minority. Kromě 4 ucelených výukových kurzů pro děti od 4 do 18 let nabízí stránka i kratší didaktické hry, které vznikly v rámci celosvětové akce Hour of Code. Code.org nabízí vytvoření učitelského účtu, přes který můžete zakládat a spravovat na portále třídy. V těchto skupinkách lze pak spravovat obsah kurzu, sledovat postup jednotlivých žáků. Podle věku cílové skupiny můžete vybrat různé způsoby přihlášení žáka do třídy – pomocí obrázku, sousloví nebo pomocí vytvořeného účtu. Další pozitivní vlastností je možnost synchronizace s Google Classroom. Výsledky výzkumu [13] poukázaly, že využití code.org pro výuku přispívá také k rozvoji geometrických a matematických znalostí.

Zajímavou didaktickou hrou je CodeCombat, který přímo nevyužívá skládání bloků, ale zanechává uživatelům nabídku s potřebnými příkazy, které pak opisují a plní úkoly jednotlivých úrovní. CodeCombat nabízí programování ve třech jazycích, Pythonu, JavaScriptu a CoffeScript, který vychází z JavaScriptu.

5 ZÁVĚR

Považujeme za vhodné se před výukou tematického celku základy robotiky zaměřit nejprve na procvičení algoritmizačních postupů a seznámit žáky se základy programování. Mnohé robotické stavebnice podporují programování pomocí blokových příkazů. Jejich užití je dnes rozšířeno i v různých aplikacích nebo didaktických hrách zaměřených na výuku programování. Na základě podobnosti mezi blokovými programovacími jazyky navrhujeme při výuce programování využít nejprve těchto vizuálních programovacích prostředí, na které mohou žáci dále navázat při programování vlastních robotických výtvorů, či v programování pomocí vyšších jazyků.

6 PODĚKOVÁNÍ

Výzkum byl částečně podpořen projekty TAČR TL01000299, TL02000050-VISNEP, IGA_PdF_2020_005 a IGA_PdF_2020_011.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] Balanskat, A., Engelhardt, K., Ferrari, A., ‘The integration of Computational Thinking (CT) across school curricula in Europe’, *European Schoolnet Perspective*, Vydání 2, 2017.
- [2] *RVP v oblasti informatiky a ICT* [online]. [vid. 23. 11. 2019]. Praha: NÚV – Národní ústav pro vzdělávání, 2019. Dostupné na: <http://www.nuv.cz/t/revize-rvp-ict>.
- [3] MŠMT, Strategie vzdělávací politiky České republiky do roku 2020 [online]. [vid. 23. 11. 2019]. Dostupné na: <http://www.vzdelavani2020.cz/>.
- [4] Informatické myšlení [online]. [vid. 23. 11. 2019]. Jihočeská univerzita v Českých Budějovicích, 2018. Dostupné na: <http://imysleni.cz/>.
- [5] Bařko, Jan. Educational robotics in the education at basic schools in the Czech Republic. *Journal of Technology and Information* [online]. 2018, 10(1).
- [6] TRAN, Yune. Computational Thinking Equity in Elementary Classrooms: *What Third-Grade Students Know and Can Do*. Journal of Educational Computing Research. 2017, 57(1), 3-31.
- [7] Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. A K-6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge. *Educational Technology & Society*. 2016, 19(3).
- [8] Roman-Gonzalez, M., a kol., Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test, *Computers in Human Behavior*, 2016
- [9] Bers, M. U., & Horn, M. S. Tangible Programming in Early Childhood. In M. J. Berson, & I. R. Berson (Eds.), *High-Tech Tots: Childhood in a Digital World Information Age Publishing*. 2010.
- [10] MIODUSER, David, Sharona T. LEVY a Vadim TALIS. Episodes to scripts to rules: concrete-abstractions in kindergarten children’s explanations of a robot’s behavior. *International Journal of Technology and Design Education* [online]. 2009, 19(1), 15-36.
- [11] Krpan D., Mladenović, S., Zaharija, G. Mediated Transfer from Visual to High-level Programming Language. 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, 2017, pp. 800–805.
- [12] Rugelj, J., Lapina, M. Game Design Based Learning of Programming. CEUR-WS. 2019, 2494.
- [13] Kalelioglu, F. A new way of teaching programming skills to K-12 students: code.org. *Computers in Human Behavior*, 2015, 52.

Ověřování učebnice LEGO robotiky pro podporu rozvoje informatického myšlení

Verification of the LEGO Robotics Textbook to Support the Development of Computational Thinking

Tomáš Jakeš
KVD FPE ZČU v Plzni
Klatovská tř. 51
306 14 Plzeň
Česká republika
tjakes@kvd.zcu.cz

Jan Bařko
KVD FPE ZČU v Plzni
Klatovská tř. 51
306 14 Plzeň
Česká republika
batko@kvd.zcu.cz

Filip Frank
KVD FPE ZČU v Plzni
Klatovská tř. 51
306 14 Plzeň
Česká republika
frankf@kvd.zcu.cz

ABSTRAKT

Článek se zabývá procesem vytváření učebnice LEGO robotiky pro 2. stupeň ZŠ v rámci projektu PRIM, která se zaměřuje na podporu rozvoje informatického myšlení. V prvotním návrhu učebnice byl maximalizován důraz na výuku programování metodou budování konceptů na úkor stavby robotických modelů. Při ověřování učebnice v praxi vyšlo najevo, že absence větší míry konstruování vyvolává u žáků pocit zklamání a nenaplnění očekávaného využití robotické stavebnice. Koncept učebnice tedy bylo nutné změnit. Nová koncepce se u žáků i jejich učitelů setkala s daleko lepší odezvou. I přes kladné hodnocení se však objevily různé připomínky, které byly následně do učebnice zapracovány. Autoři učebnice v příspěvku představují vize vytvářené učebnice a popisují postupný vývoj a proměny od prvotní testované verze po současnou.

ABSTRACT

This article deals with the process of making up the LEGO robotics textbook for upper primary schools in the project PRIM. The textbook focuses on the support of the development of computational thinking. In the original concept of the textbook, the emphasis on teaching programming via the method of building concepts instead of building robotic models was maximized. During pilot testing of the textbook in practice appeared that the absence of higher extent of constructing causes the feeling of disappointment and unfulfillment of the expected usage of robotic kit. Therefore, it was necessary to change the concept of the textbook. The new concept had much better response among pupils and their teachers. Even if the textbook evaluation was successful, some negative comments occurred which we used to improve the textbook. The authors of the textbook introduce in this article the visions of the textbook which is being prepared and described the progressive development and changes from the original tested version to the current one.

Klíčová slova

Robotika, učebnice, 2. stupeň ZŠ, programování, pilotní ověřování.

Keywords

Robotics, textbook, upper primary schools, programming, pilot testing.

1 ÚVOD

Informatické myšlení je pojem, který je v poslední době poměrně čteně skloňován. Přesto se dle Wang [1] nejedná o zcela novou dovednost, ale o způsob myšlení, který lidé pohybující se v informatice běžně používají už poměrně dlouho. Tento způsob myšlení však nemusí nutně používat jen informatici. Způsob uvažování v souladu s informatickým myšlením je možný použít i v běžném

životě. Ať už se bude jednat o předcházení chybám, nebo vyhodnocení nejefektivnějšího řešení. Jeanett M. Wing [2] pak definuje informatické myšlení, jako složitý myšlenkový proces zabývající se formulací problémů a formulací řešení problémů takovým způsobem, aby tato řešení mohla být využita zvoleným agentem pro jejich zpracování. Tato řešení pak mohou být provedena počítačem, člověkem, nebo jejich kombinací.

Informatické myšlení nám pak přináší dle webu projektu „Podpora rozvíjení informatického myšlení“ iMyšlení.cz [3] dovednost systematicky posoudit různá řešení a vybrat to nejvhodnější pro danou situaci. Dále dokážeme rozdělit velký problém na několik menších, které vyřešíme snáze a dojde tak k vyřešení původního rozsáhlejšího problému. Dokážeme plánovat a řídit činnosti, vytvářet a pečlivě popisovat postupy, které spolehlivě vedou k nějakému cíli, i když je vykonává někdo jiný. Tuto dovednost zmiňuje i Wang [1], který ji nazývá jako „opuštění bubliny“. Toto opuštění pak vysvětluje tím, že záleží na tom, s kým komunikujeme a musíme při komunikaci vnímat znalosti a obor příjemce informace. Web iMyšlení pak zmiňuje další tři dovednosti. Dovednost vybrat, které aspekty problému jsou podstatné pro jeho řešení a které lze zanedbat. Uspořádat i velké a nesourodé soubory dat tak, abychom je mohli využít. Použít jazyky, kterými se domluvíme s počítači, roboty a umělou inteligencí.

Rozvoj informatického myšlení je možné podpořit i pomocí robotických programovatelných stavebnic. Právě pro tento účel vzniká učebnice Robotika pro 2. stupeň ZŠ s LEGO Mindstorms [4].

2 VÝCHODISKA PRO TVORBU UČEBNICE

Učebnice je koncipována jako volitelné rozšíření rozvoje informatického myšlení pomocí robotických stavebnic na 2. stupni ZŠ. Využívá robotickou stavebnici LEGO Mindstorms EV3 a pro její programování i originální stejnojmenný zdarma dostupný ikonický programovací jazyk. Výběru robotické stavebnice předcházelo dotazníkové šetření zjišťující dostupnost vybavení základních škol [5] i ověřování možností a způsobu programování různých typů stavebnic (LEGO Boost, LEGO Mindstorms EV3, VEX IQ, MakeBlock mBot, OzoBot).

Učebnice byla původně určena pro žáky 7. až 9. ročníků základní školy s možným přesahem do nižších ročníků SŠ. Na základě výsledků ověřování však došlo k posunu a doporučení její vhodnosti až od 8. třídy.

Učebnice vychází z cílů připravovaného inovovaného RVP. Volně navazuje na učebnici „Edukační robotika s LEGO® WeDo 2.0 pro 1. stupeň základní školy“. Využívá však odlišné programovací prostředí a konstrukční prvky postavené na dílech LEGO Technic. Zkušenosti s programovacím jazykem (např. Scratch) jsou doporučeny, nejsou však vyžadovány. V tomto případě je však nutno počítat s pomalejší prací žáků a nutností pečlivějšího budování nových konceptů.

Učebnice vychází z konstruktivistického pojetí výuky [6]. Zaměřuje se na řešení úloh inspirovaných reálnými situacemi. Využívá prvky problémového vyučování a metody postupného budování konceptů. Žáci analyzují možnosti a navrhují a testují různá vlastní řešení. To je z části kombinací programu, dále pak konstrukce i vhodného vnímání okolí pomocí senzorů robotické stavebnice LEGO Mindstorms EV3. Přináší problémy a faktor reálného světa. Řešení tak může být značně různorodé, komplexní a zatížené reálnými podmínkami, např. prokluz kol na povrchu, odlišnost výkonu motorů, nepřesné měření vzdálenosti senzorem, výchozí pozice částí robota. Žáci jsou vedeni ke schopnosti dekompozice složitých problémů na menší a snáze řešitelné části. Podstatná je i práce s chybou. Žák musí identifikovat její projevy, detekovat chybu, objevit místo jejího vzniku, navrhnout opravu a tu následně otestovat. Učebnice by měla přispět k budování sebejistoty ve schopnosti řešit problémy.

Před samotnou tvorbou učebnice jsme řešili také její formát a podobu. Naším cílem bylo vytvořit produkt, který bude snadno dostupný, přehledný, jednoduše editovatelný a žáky i učiteli oblíbený. Rada tvůrčích týmů v projektu PRIM vytváří učebnice jako sadu PDF souborů nebo prezentací. My jsme však potřebovali učebnici doplnit o programy, konstrukční návody, fotogalerie a další soubory.

Původně jsme učebnici plánovali rozšířit o tištěné metodické listy pro žáky. Na základě negativních zkušeností týmu zpracovávajícího robotiku s LEGO WeDo s touto formou jsme však nakonec raději zvolili možnost vpisovat poznámky a řešení do učebnice v digitální podobě. Z uvedených důvodů jsme proto připravili učebnici v podobě webových stránek.

3 PŮVODNÍ KONCEPT UČEBNICE

Prvotní navržená verze učebnice se skládala z 11 kapitol různé obtížnosti.

Hlavním rysem celého konceptu byla snaha o postupné budování konceptů a eliminaci složitých konstrukcí, jejichž stavba by byla na úkor rozvoje dovedností směřujících k informatickému myšlení.

Prvním problémem, který jsme řešili, byla náročnost a návaznost jednotlivých aktivit. Při reálném využívání robota jsou i v základních programech běžně využívány řídicí struktury jako je cyklus či větvení. V rámci postupného budování konceptů jsme však nejdříve potřebovali, aby žáci experimentovali s bloky, měnili jejich parametry a postupně objevovali jejich vliv.

1. Zvuk a displej: Úvodní kapitola byla proto postavena na aktivitách se zvukem a displejem, kdy žáci mohli plně vnímat změny a jejich dopad, a to vše jen s využitím řídicí jednotky. Další část kapitoly byla zaměřena na hledání a opravu chyb různého typu. V závěrečné části pak žáci objevovali, proč program po odstranění bloku se zvukem zdánlivě neprobíhá. Chtěli jsme, aby si uvědomili, jak dlouho jednotlivé instrukce či celý program trvají.

2. Mixér: V druhé kapitole s názvem mixér žáci konstruovali a zprovožňovali jednoduchý mixér s motorem. K mixéru pak přidali ovládací tlačítko (vstupní senzor) a reagovali na jeho stisk. Dostali také připravený program, ve kterém hledali a opravovali zdánlivě neexistující sémantickou chybu. Ta vedla k nefunkčnosti jednoho z režimů mixéru. V programu byl prvně bez upozornění zakomponován i nekonečný cyklus zajišťující nepřetržitý provoz mixéru.

3. závora a 4. inteligentní závora: Ve třetí a čtvrté kapitole žáci sestavovali závora na parkovišti. Tato úloha již vyžadovala dekompozici složité úlohy na dílčí, snáze řešitelné části. Žáci si museli uvědomit funkce a režimy závoř, včetně chybových, a pojmenovat prvky, které k řešení budou potřebovat. Jednoduché ovládání závoř bylo v druhé části aktivity doplněno o automatickou detekci překážky pomocí ultrazvukového senzoru a detekci příjíždějícího auta vedoucí k větvení programu.

Řazení kapitol mělo význam i z hlediska konstrukcí. Vytvořené konstrukce byly jednoduché, šlo je snadno rozebrat a díly použít ke stavbě v dalších úlohách.

5. Pojízdný robot: V páté kapitole měli žáci sestavovat pojízdného robota dle návodu a vyzkoušet základní pohyb.

6. Robot ve městě: Pohyb byl dále rozvíjen i v šesté kapitole. Jednalo se o problémové úlohy vedoucí k různému řešení natočení o daný úhel. Důraz tak byl kladen na hledání možných řešení a jejich testování. Základem kapitoly je mapa města, ve které jsou znázorněna místa, jež jsou součástí níže popsaného příběhu. Úkolem žáků je plnit jednotlivé aktivity a provést robota po jednotlivých krocích městem. Vyřešení každé dílčí aktivity vyžaduje jiný způsob pohybu robota a jejich obtížnost má stoupající tendenci. Žáci navíc musí respektovat směrové značení v jednotlivých ulicích.

7. Adaptivní tempomat: V sedmé kapitole řešili žáci detekci překážek a automatizované přizpůsobení rychlosti robota před nimi jedoucím vozidlu. Vychází přitom ze svých předchozích znalostí. Aktivita vede k různým řešením s různou kvalitou od skokového řízení (např. stůj, 50 % a 100 % výkonu), po zcela plynulé řízení výpočtem. Při ladění programu žáci také objevují limity senzoru, jako je např. minimální a maximální rozsah či úhel snímání a řeší stabilizaci pohybu robota.

8. Inteligentní robot: V osmé kapitole se žáci seznámili s možností přesné navigace robota pomocí sledování černé vodící čáry s využitím senzoru barvy. Podstatný je zde zejména princip rozpoznání pozice vzhledem k čáře a odladění výsledného řešení. Ladění spočívá nejen v úpravě parametrů algoritmu rozpoznání čáry, ale též i v analýze vrácených hodnot ze senzoru, kalibraci i jeho fyzickém umístění.

9. Parkovací asistent: V deváté kapitole mají žáci vytvořit parkovacího asistenta, který vyhledá volné místo a automaticky vozítko zaparkuje. Je to první úloha, pro jejíž řešení je vhodné využít proměnné, přesto to není nezbytně nutné. Úloha je velmi dobře škálovatelná, od jednoduchého rozlišení parkovacího místa po parkování na místě s překážkami (např. strom, lampa, motocykl).

10. Hra „Videostop“: V desáté kapitole žáci vytvářeli vlastní hru na motiv videostopu. Úloha je postavena na dekompozici složitého problému na dílčí včetně testování těchto dílčích částí. V plné míře se zde již projevuje vhodnost využití proměnných (stav, skóre) i schopnost předvídání problémů (např. detekce výhry/prohry či kolize při současném stisku obou hráčů).

11. Spolupráce – parkoviště: Závěrečná kapitola rozvíjí schopnost spolupráce žáků. Jejich úkolem bylo vytvořit parkoviště s vjezdovou a výjezdni bránou informující o počtu volných míst na displejích obou jednotek. Základem byla komunikace více řídících jednotek pomocí Bluetooth. Žáci museli jednotky propojit a domluvit se na způsobu a pravidlech předávání informací. Na jejich základě pak vytvořit a otestovat přijímací a vysílací část svého zařízení.

4 OVĚŘENÍ UČEBNICE

Ověření v pilotáži mělo dvě fáze. Úvodní byla prováděna v období květen až červen 2018 u žáků 7. ročníku na 10. ZŠ v Plzni. Ve skupině bylo 14 žáků, z toho 4 dívky. Žáci se stavebnicí LEGO Mindstorms EV3 dosud nepracovali, měli však z předchozího ročníku zkušenosti s programováním v krátkém kurzu na stránce code.org. Primární využitou metodou bylo nezúčastněné otevřené pozorování doplněné o nestrukturované rozhovory s vyučujícím a vybranými žáky, které nám pomohly hlouběji porozumět problémům [7].

Druhou fází pilotáže bylo souběžné testování učebnice na 4 různých základních školách napříč Českou republikou. Testování probíhalo v období září 2018 až květen 2019. Hlavním nástrojem bylo dotazníkové šetření s otevřenými otázkami zadávané ke každé z kapitol a metodickým listům pomocí webového portálu vyvinutého přímo pro zadávání hodnocení. S učiteli zapojenými do testování jsme byli v kontaktu i prostřednictvím e-mailu a telefonu.

Doplňující metodou bylo opět nezúčastněné otevřené pozorování na 10. ZŠ v Plzni v období únor až březen 2019 doplněné o nestrukturované rozhovory s vyučující a vybranými žáky. Jednalo se o žáky 9. ročníku s rozšířenou výukou informatiky. Ve skupině bylo 11 žáků, z toho 3 dívky. Dva žáci měli zkušenosti se stavebnicí LEGO Technic, žádný však se stavebnicí LEGO Mindstorms EV3. Žáci v předchozích ročnících absolvovali krátkou výuku programování v prostředí Scratch neodpovídající nově vytvářeným učebnicím. Charakter ostatních škol byl velmi variabilní co do zkušeností žáků i vyučujících, počtu hodin i žáků ve třídě (až 24 žáků).

Nezúčastněné otevřené pozorování, které prováděli autoři učebnice, se zaměřovalo na činnost žáků i učitelů při hodině, práci s učebnicí, pochopitelnost a přiměřenost úloh a motivaci žáků. Rozhovory sloužily zejména pro zjištění zpětné vazby od učitele, jeho vnímání hodiny, vhodnost materiálů i práci žáků. U žáků jsme sledovali zejména jejich motivaci, porozumění úlohám i jejich způsob řešení problémů.

Vše bylo prováděno s cílem odhalit problémová místa a vylepšit učebnici a k ní příslušné metodické materiály.

5 HLAVNÍ PROBLÉMY ZJIŠTĚNÉ PILOTÁŽÍ

Na velice zásadní problém jsme narazili ihned při ověřování první kapitoly učebnice. Z důvodu systematického budování konceptů a větší efektivity bylo našim původním záměrem v úvodu co nejvíce omezit konstrukce a ve větší míře se tak zaměřit na rozvoj informatického myšlení žáků. Kýženou efektivitu to ovšem nepřineslo. Jak jsme z pozorování a rozhovorů zjistili, výrazně jsme podcenili očekávání žáků a jejich touhu konstruovat. Žáci čekali, že si na první hodině postaví vlastního pojízdného LEGO robota, který bude bez nutnosti dalšího programování autonomně

pracovat. Práce s řídicí jednotkou, zvukem a displejem tuto představu nenaplněovala a dostatečně je nemotivovala.

Další problém zklamání ještě umocnil. V učebně byla nainstalována nesprávná verze programovacího prostředí LEGO Mindstorm EV3 Home. Dle metodických pokynů přitom měla být instalována rozšířená a pro výuku lépe uzpůsobená verze Education. Verze Home je zaměřena na odlišnou cílovou skupinu domácích uživatelů. Tomu je také uzpůsobeno grafické provedení aplikace plně efektů, zvuků a animací různých předpřipravených robotů. Animace žáky ještě více odvedly od jejich činnosti.

Větší motivaci nepomohly ani na konstrukci bohatší aktivity z dalších kapitol, např. mixér či závora. Některé žáky více motivovala až stavba autonomního vozidla. I u té se však vyskytly problémy. Jeho stavba byla realizována pomocí PDF návodu. Ten byl vytvořen a do PDF exportován ve volně dostupném prostředí LEGO Digital Designer. Program export generuje autonomně a ve vlastním zvoleném pořadí neodpovídajícím postupu tvorby konstrukce modelu. Díky tomu se návod stal na pochopení velmi náročným. Žáci si s ním sice poradili, měli však velké množství dotazů a pracovali pomalým tempem. Při konstruování docházelo ke značným časovým rozdílům v dokončení modelu jednotlivými skupinami. Některé model sestavily za 35 minut, jiným skupinám konstruování zabralo téměř 80 minut.

Poněkud nejednoznačné výsledky jsme získali při ověření úloh využívajících zvukový výstup. Někteří žáci tuto úlohu zvládali dobře a dělala jim radost, jiným žákům dělalo problémy pracovat s rozsáhlejší písničkou složenou z tónů a odhalit v ní připravené chyby nebo naopak správné tóny doplnit.

Na velice zásadní problémy jsme naráželi také u pokročilejších kapitol, zejména u závěrečné kapitoly spolupráce pomocí Bluetooth, kterou žáci bez výrazné pomoci nedokázali dokončit. Problematická byla i hra „Videostop“, která vyžadovala hlubší porozumění i schopnost aplikace předchozích poznatků. Žáci velice obtížně chápali pravý účel proměnné a její využití.

U některých úloh jsme narazili na problém s jejich rozsáhlostí. Skupiny žáků nedokázaly všechny aktivity v námi odhadovaném čase vyřešit. Nyní je proto potřeba dále sledovat, čím je toto způsobeno a stanovit, která část je pro pokračování na další kapitolu podstatná a je tedy třeba ji absolvovat a kterou již lze považovat za rozšiřující.

Problémy jsme registrovali také ve spojitosti s grafickým provedením učebnice a pojmenováním jednotlivých záložek a prvků. Žáci velice často přeskakovali bloky obsahující teoretické informace a postupovali přímo k aktivitám. Zde poté často naráželi na chybějící souvislosti. Zároveň jsme také postrádali prostor pro zapisování poznámek z řešení úkolů nebo odpovědí na otázky.

Poslední skupina problémů objevujících se při testování, na které se podrobněji zaměříme, jsou problémy související s prací učitele a s návyky žáků. Vedení výuky robotiky bylo pro učitele začátečníky značně náročné. Žáci řeší aktivity individuálním tempem a vytvářejí odlišná řešení, která učitel ne vždy očekává. Naší vizí bylo vytvořit učebnici tak, aby žáci nemuseli s každým dílčím řešením čekat na učitele, mohli si ho poznamenávat a ve skupince společně diskutovat.

Z testování vyplynulo, že žáci měli často tendenci volat učitele. Často ho volali nejen, když měli problém něco pochopit, ale také v případě, kdy vyřešili nějaký, často jen dílčí úkol. Pokud se vyučující věnoval jiným spolužákům, žáci na něj většinou čekali a dále nepracovali. Tomuto stavu částečně napomáhala i práce učitele, který úlohy důsledně kontroloval a nepřenechával odpovědnost žákům. Ukázalo se, že pro zvýšení efektivity bude užitečné, pokud by učitel mohl zkontrolovat více dílčích aktivit najednou. Bylo taktéž potřeba vhodný přístup učitele co nejlépe zpracovat v rámci metodických pokynů.

Nová nepředvídaná řešení mnohdy vyústila v nejistotu vyučujícího. Z rozhovoru vyplynulo, že by nejistotu snížil přehled o programovacím prostředí, jeho ovládání a propojování programových bloků.

6 ÚPRAVY A SOUČASNÁ KONCEPCE UČEBNICE

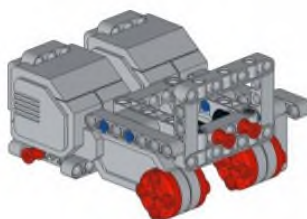
S ohledem na zjištěné skutečnosti byla koncepce učebnice zcela přepracována. Nyní jsou více reflektována očekávání žáků. Došlo k nahrazení úvodních aktivit za aktivity vycházející z konstrukce a ovládání pojízdného robota. Úvodní aktivity byly přepracovány a využity v další části učebnice.

Na základě testování byla poslední kapitola zcela odstraněna. Větší prostor byl věnován pohybu robota. Náročnost některých kapitol byla upravena vhodnějšími aktivitami, např. byla vyměněna závěrečná hra.

Tabulka 1: Porovnání původní a nové struktury učebnice

Původní struktura učebnice	Nová struktura učebnice
1. Zvuk a displej	1. Stavíme pojízdného robota
2. Mixér	2. Oživení robota
3. Závora	3. Robot ve městě
4. Inteligentní závora	4. Zvuk a displej
5. Stavíme pojízdného robota	5. Mixér
6. Robot ve městě	6. Závora
7. Adaptivní tempomat	7. Automatická závora
8. Inteligentní pojízdný robot	8. Adaptivní tempomat
9. Parkovací asistent	9. Inteligentní pojízdný robot
10. Hra „Videostop“	10. Parkovací asistent
11. Spolupráce – parkoviště	11. Hra „Kdo má lepší postřeh?“

1. Stavíme pojízdného robota: Nově je úvodní kapitola zaměřena na konstrukci pojízdného robota za pomoci návodu. Protože si některé školy zakoupily místo doporučené verze Education stavebnici ve verzi Home, vytvořili jsme i pro tuto na díly odlišnou stavebnici alternativní verzi návodu. Pro rychlé žáky je připraven úkol, ve kterém se pokusí rozpohybovat svého sestaveného robota.



Obrázek 1: Ukázka části návodu pro konstruování.

2. Oživení robota: Kapitola nahrazuje původní úlohy zaměřené na postupné poznávání programovacího prostředí i chování robota. Základem aktivity je pohyb – práce se servomotory. Žáci se naučí robota ovládat, měnit parametry jednotlivých motorů, ovlivňovat ujetou vzdálenost a měnit směr pohybu robota.

3. Robot ve městě: Aktivity byly drobně vylepšeny. Zpracováno však bylo na metodickém pokynu a nové mapě, která by měla usnadnit vyučujícím práci s přípravou mapového podkladu. Z testování např. vyplynulo, že na různých školách volili různé postupy, od hromadné práce žáků na novém plánu při výtvarné výchově, po tisk segmentů mapy na barevné tiskárně.

4. Zvuk a displej: Kapitola byla přesunuta a doplněna o vzhled do hardwarových součástí řídicí mikroprocesorové jednotky stavebnice EV3. Primárně se ovšem stále zaměřuje na práci se zvukovým výstupem řídicí jednotky a jejím displejem. Vybrána byla jiná písnička. Chyby byly více zdůrazněny.

5. Mixér: Kapitola byla upravena vzhledem k pozdějšímu zařazení.

6. Závora: Žák je více vybízen k aktivitě v podobě vlastní konstrukce závory. Přesný návod byl odstraněn, ponechány byly jen ilustrační fotografie.

7. Automatická závora: Kapitola byla na základě výsledků testování velmi výrazně přepracována. Žáci jsou více vybízeni k hledání řešení i dekompozici problému. V další fázi je ovládání závory realizováno za pomoci barevných kostek představujících myšnou kartu.

8. Adaptivní tempomat: Žáci většinou volili jednodušší skokové řešení řízení.

9. Inteligentní pojízdný robot: Pochopení principu sledování čáry žákům dělalo značné problémy. Rozhodli jsme se proto částečně odchýlit od zcela konstruktivistického přístupu a představili jsme symbolické grafiky možných řešení. Žáci z nich mají vybírat a zdůvodňovat ty nejlepší.

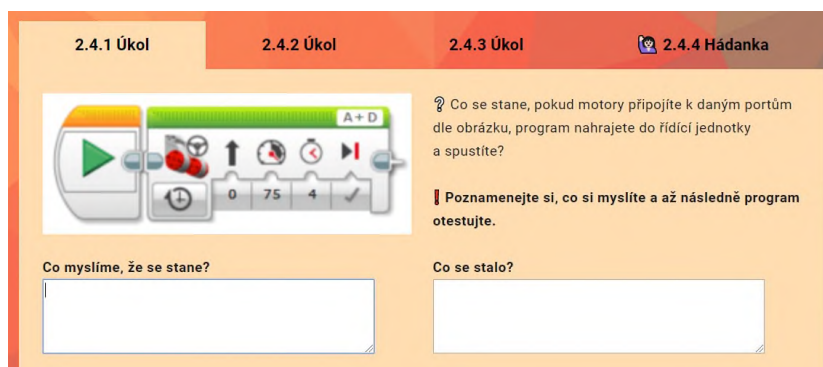
10. Parkovací asistent: Jedná se o náročnou, ale přínosnou kapitolu s dobrou škálovatelností. Kapitola proto byla zachována.

11. Hra „Kdo má lepší postřeh?“: Původní hra byla příliš náročná. Kombinovala navíc již tak značně problematickou tematiku proměnných s návrhem nové dosud neřešené komplexní úlohy. Byla proto nahrazena za jednodušší typ s lepší škálovatelností dle zkušeností žáků.

Z učebnice jsme zcela vypustili zamýšlenou kapitolu zaměřenou na komunikaci pomocí Bluetooth. Kapitola byla enormně náročná, zejména kvůli nutnosti párování jednotek, jejich rozlišování při komunikaci a složitějšímu principu zasílání zpráv. Docházelo ke značné řadě otázek a okamžiků nepochopení.

Vzhled učebnice a její funkční prvky

Značných změn doznala také vizuální podoba učebnice. V jednotlivých záložkách obsahujících úkoly a teoretické informace jsme sjednotili jejich názvy a popisky, aby nepůsobily zavádějícím dojmem. Přidali jsme také číslování záložek, které usnadňuje učitelům orientaci v kapitole. Stejné číslování se totiž nachází také v metodických pokynech. Ve většině kapitol jsou umístěna vstupní pole, do kterých si mohou žáci zaznamenávat svůj postup, odpovědi na otázky nebo řešení úkolů. Vyučující tak může úkoly snáze kontrolovat a žáci nepotřebují pro psaní poznámek a řešení papír. Vymezili jsme si také základní sadu emotikon, které jasně upozorňují, o jakou část učebnice se jedná (např. úkol pro rychlé žáky, tip pro řešení, teoretická informace). Jak vizuální podoba aktivity v učebnici v tuto chvíli vypadá znázorňuje obrázek 2.



Obrázek 2: Ukázka vizuální podoby části učebnice.

Metodické pokyny

Samostatnou částí učebnice jsou metodické pokyny pro učitele, které můžeme rozdělit do několika částí. Na webové stránce uvádíme ke každé kapitole některé základní informace a užitečné typy související s organizací výuky, popisem stavebnic nebo aktivitou žáků. Základem jsou metodické pokyny ke všem jednotlivým kapitolám. Každý pokyn seznamuje učitele s cíli dané kapitoly, je zde uveden předpokládaný časový rozsah aktivity, uvádíme, jak při řešení aktivity postupovat a co si před začátkem výuky připravit. Hlavní část každého dokumentu tvoří popis částí kapitoly s uvedením možných variant správného řešení úkolu včetně jejich zdůvodnění. Pro snazší orientaci je uváděno totožné číslování, které je využíváno v učebnici. Ukázku vysvětlení krátkého úkolu znázorňuje obrázek 3.

2.5 Souhrnný úkol - střídáme rychlosti

Zadání: Vytvořte program, pomocí kterého robot pojede 4 otáčky motorů rychlostí 50 % a poté na dvě otáčky motorů zrychlí na 100 %.

Řešení: K řešení jsou potřeba dva bloky pro ovládání motorů, každý s rozdílným nastavením (viz obrázek 5).



Obrázek 7 - Správné řešení souhrnného úkolu.

Obrázek 3: Ukázka části metodického pokynu.

Při testování na školách se ukázalo, že řada vyučujících nemá s robotikou příliš zkušeností a programování je tak pro ně zpočátku velice obtížné. Abychom jim poskytli ještě větší podporu, vytvořili jsme ještě příručku programových bloků. Tento dokument obsahuje popis všech programových bloků prostředí EV3 používaných v učebnici. Popisujeme jejich umístění, veškeré parametry a režimy (viz obrázek 4).



Obrázek 4: Ukázka popisu programového bloku.

7 ZÁVĚR

Učebnice robotiky pro 2. stupeň ZŠ s využitím LEGO Mindstorms EV3 prošla pilotáží, která pomohla učebnici výrazně vylepšit. V současnosti je publikována ve svojí beta verzi na adrese <https://lego.zcu.cz/ucebnice/>. K učebnici je dále připojena sada metodických pokynů pro učitele zveřejněná na adrese <https://www.lego.zcu.cz/ucebnice/metodicke-listy-3.html>. Učebnice ve školním roce 2019/2020 prochází dalším kolem testování, a to na pěti základních školách v ČR. Testování naznačuje, že provedené změny byly úspěšné.

8 BIBLIOGRAFICKÉ ODKAZY

- [1] Wang, Paul. *From computing to computational thinking*. Ohio: CRCPress, 2016. 978-1-4822-1766-7.
- [2] Wing, Jeanett M. www.cs.cmu.edu. *Carnegie Mellon University* [online]. 17. 11. 2010. [vid. 1. 12. 2019]. Dostupné na: <https://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>.
- [3] iMyšlení. Informatické myšlení: Co je IM. *iMyšlení!* [online]. Jihočeská univerzita v Českých Budějovicích. [vid. 1. 12. 2019]. Dostupné na: <https://imysleni.cz/informaticke-mysleni/co-je-informaticke-mysleni>.
- [4] Vzdělávací materiály: Robotika na 2. stupni základní školy s LEGO® Mindstorms. *iMyšlení!* [online]. Jihočeská univerzita v Českých Budějovicích. [vid. 1. 12. 2019]. Dostupné na: <https://imysleni.cz/ucebnice/robotika-na-2-stupni-zakladni-skoly-s-lego-mindstorms>.
- [5] BAŤKO, J. Robotika ve výuce na základních školách v České republice. Výzkumná zpráva projektu SGS-2016-042 Potenciál robotické stavebnice ve výuce na základní škole. Plzeň: Západočeská univerzita, Fakulta pedagogická, 2017.
- [6] MAŠEK, Jan, Petr MICHALÍK a Václav VRBÍK. *Otevřené technologie ve výuce*. Plzeň: Západočeská univerzita v Plzni, 2004. ISBN 80-7043-254-3.
- [7] ŠVARŤÍČEK, Roman a Klára ŠEĐOVÁ. *Kvalitativní výzkum v pedagogických vědách*. Vyd. 2. Praha: Portál, 2014. ISBN 978-80-262-0644-6.

Využití robotické stavebnice VEX IQ ve výuce na základních školách

Use of VEX IQ Robotic Kit in Primary Schools

Hana Hyksová
Univerzita Palackého v Olomouci
Žižkovo náměstí 5
771 40 Olomouc
Česká republika
hana.hyksova01@upol.cz

ABSTRAKT

Robotické hračky jsou v dnešní době jednou z cest, jak začít u dětí budovat vztah k technice a robotice už ve školce či na základní škole. Robotická stavebnice VEX IQ má oproti robotickým hračkám další přidanou hodnotu – podporuje manuální zručnost, konstrukční dovednosti, kreativitu a týmovou spolupráci. Toto jsou v dnešní době některé ze základních důležitých kompetencí k uplatnění na pracovním trhu. Robotická stavebnice umožňuje STEAM (spojení vědy, technologií, inženýrství, umění a matematiky) výuku v praxi.

Článek popisuje součásti a využití robotické stavebnice VEX IQ na základních školách – mozek robota, jednotlivé snímače, příprava a stavba robota, programování a ovládání robota. Autorka ukazuje možnosti využití této robotické stavebnice ve výuce na základní škole a při různých robotických soutěžích. Hlavní pozornost je věnována porovnání robotických stavebnic VEX IQ a LEGO MINDSTORMS.

ABSTRACT

Nowadays, robotic toys are one of the ways how to start building a relationship with children and technology, by using robotics in kindergarten or elementary school. The VEX IQ robot kit has an added value over robotic toys as it supports manual dexterity, design skills, creativity and teamwork. Nowadays, these are some of the basic important competences for the job market. The robotic kit allows STEAM (combining science, technology, engineering, art and mathematics) to teach in practice.

The article describes the components and applications of the VEX IQ robotics kit at elementary schools; the robot brain, individual sensors, robot preparation and construction, robot programming and control. The author shows the possibilities of using this robot kit in teaching at elementary schools and in various robotic competitions. The main attention is towards the comparison of robotic kits VEX IQ and LEGO MINDSTORMS.

Klíčová slova

robot, programování, programování robota, robotická stavebnice, VEX IQ

Keywords

robot, programming, robot programming, robotic kit, VEX IQ

1 ÚVOD

Robotika a roboti jsou v současné době velmi používaná slova. Ve školách se pracuje s různými robotickými hračkami a stavebnicemi a učitelé je zařazují ve školách různými způsoby. Nejvíce se s nimi pracuje v klubech robotiky, klubech logiky a badatelských klubech. Další aktivity s roboty

jsou zařazovány v rámci projektových dnů. Tyto aktivity jsou na základních školách financovány z projektu MŠMT – Šablony II. S robotickými hračkami a stavebnicemi žáci pracují i v rámci vyučovacích hodin v různých předmětech. S robotickými hračkami, u kterých nepotřebujeme žádné konstrukční dovednosti, mohou žáci pracovat již od prvního ročníku. Mezi tyto robotické hračky patří Bee-Bot, Blue-Bot či Ozobot. Využívání těchto robotických hraček je vhodné, neboť s nimi rozvíjíme u dětí různé dovednosti, např. tvořivost. Ve vyšších ročnících využíváme k rozvoji konstrukčních dovedností například stavebnice LEGO MINDSTORMS Education EV3 či VEX IQ Super Kit. Žáci staví jednoduché a později i složitější roboty, které navíc programují a programem řídí.

Proč robotika? Úkolem pedagogů není pouze výchova a vzdělávání, ale také příprava žáků na budoucnost a budoucí povolání. Žijeme ve 21. století a v souvislosti s tím rozvíjíme i klíčové kompetence a dovednosti 21. století [3].

Mezi klíčové kompetence 21. století patří:

- umět se rozhodovat, reagovat na změny
- pochopit problém a hledat řešení (informatické myšlení)
- logické myšlení a matematická gramotnost
- čtenářská a digitální gramotnost
- práce v týmu (soft skills)
- kreativita
- prezentace a obhajoba vlastních myšlenek a řešení (schopnost diskuze)
- pracovitost (dotahovat práci do konce)

Robotika rozvíjí logické myšlení a matematickou gramotnost. Žáci konstrukčně sestavují roboty a programují je. Musí vytvářet různé algoritmy. S tím úzce souvisí i čtenářská a digitální gramotnost, neboť žáci pracují s návody, postupy konstrukcí, s plány a musí se v nich správně orientovat. S tím souvisí i umět se rozhodovat a reagovat na změny (změny v programu, jiné řešení programu apod.) Žáci pracují v týmech při konstrukcích robotů i při programování. Při svých činnostech musí být kreativní (různé možnosti řešení), musí dotahovat práci do konce. Pokud nedokončí konstrukci či program, nebude jim robot správně fungovat. Své myšlenky a řešení vždy obhajují před ostatními spolužáky a diskutují o svém řešení.

Důležité kompetence k uplatnění na pracovním trhu jsou uvedeny na portálu Národního pedagogického institutu České republiky „Informační systém o uplatnění absolventů škol na trhu práce“ [4]. Jako nejdůležitější znalosti, dovednosti a schopnosti označili:

- čtení a porozumění pracovním instrukcím
- schopnost nést zodpovědnost
- ochotu učit se
- schopnost řešit problém.

2 UČEBNÍ MATERIÁLY S VYUŽITÍM STAVEBNIC LEGO MINDSTORMS EDUCATION EV3 A VEX IQ SUPER KIT

V rámci projektu PRIM – Podpora rozvíjení informatického myšlení byla vytvořena učebnice pro rozvoj informatického myšlení a vznikla řada materiálů, které mohou učitelé využít v hodinách informatiky [1]. Učebnice a vzdělávací materiály jsou v současné době pilotovány. Nedílnou součástí je i oblast robotiky. Pro tuto oblast byl vytvořen materiál pro práci s robotickou stavebnicí LEGO MINDSTORMS Education EV3. Učebnice je určena pro žáky 8. a 9. ročníku, ve které najdeme 11 výukových aktivit:

1. Stavíme pojízdného robota
2. Oživení pojízdného robota
3. Robot ve městě
4. Řídicí jednotka, zvuk a displej
5. Mixér, aneb pracujeme s motorem

6. Závora na parkovišti
7. Automatická závora
8. Adaptivní tempomat – detekce překážky
9. Inteligentní pojízdný robot
10. Parkovací asistent
11. Hra „Kdo má lepší postřeh?“

Ve školách je ovšem využívána i méně známá stavebnice VEX IQ Super Kit. Pro tuto stavebnici jsou též vytvořeny materiály [2]. Sedm výukových aktivit je určeno především pro žáky 8. a 9. ročníku. Těmito aktivitami jsou:

1. Robot Sumo
2. Robot s mostem
3. Robotická ruka
4. Třídící robot
5. Kreslicí robot
6. Robotický hudebník
7. 3D modelování robotů

Materiály pro obě robotické stavebnice jsou velmi dobře rozpracovány do výuky. Kapitoly obsahují návody na konstrukci robotů, metodické listy pro žáky a metodické pokyny pro učitele. Rozborem těchto kapitol a využitím se budu zabývat v jiném článku.

3 HARDWAROVÉ VYBAVENÍ STAVEBNIC LEGO MINSTORMS EDUCATION EV3 A VEX IQ SUPER KIT

V robotické stavebnici LEGO MINSTORMS Education EV3 se kromě různých součástek, které jsou nutné ke konstrukci a stavbě robota, nacházejí tyto hardwarové součástky: inteligentní kostka, dva velké a jeden střední servomotor, světelný senzor, ultrazvukový senzor, gyroskopický senzor a dotykový senzor.

V robotické stavebnici VEX IQ Super Kit z hardwarových součástek najdeme: mozek robota, tři motory, ovladač VEX, gyro snímač, snímač vzdálenosti, dotyková LED, spínač nárazníku, snímač barev. Všechny motory jsou zde rovnocenné.

3.1 Řídící jednotka

Kostka (mozek) ve stavebnici LEGO obsahuje 8 portů. Čtyři porty jsou vstupní (pro připojení senzorů) a čtyři porty jsou výstupní (pro připojení motorů).



Obrázek 1: Inteligentní kostka LEGO MINSTORMS, zdroj vlastní

Mozek robota VEX IQ má 12 portů a nezáleží na tom, do kterého portu zapojíme motor a do kterého čidlo. Do mozku robota VEX IQ můžeme připojit o čtyři součástky více.



Obrázek 2: Mozek robota VEX IQ, zdroj vlastní

3.2 Čidla (snímače)

Do obou robotů můžeme zapojit světelný senzor, gyroskopický senzor a dotykový senzor. Součástí robotické stavebnice VEX IQ Super Kit je i dotyková LED a ultrazvukový senzor. U stavebnice LEGO MINDSTORMS Education EV3 je infračervený senzor s měřením vzdálenosti. Funkce čidel je stejná. Obě stavebnice obsahují i ovladač. U obou robotických stavebnic můžeme programovat i zvuky.

3.3 Ostatní parametry

Ceny obou stavebnic jsou též srovnatelné a pohybují se okolo 12 000 Kč. K oběma stavebnicím lze dokupovat další díly.

Obsah stavebnice LEGO MINDSTORMS Education EV3:

- 601 dílků
- kostka EV3
- dva velké interaktivní servomotory s integrovaným rotačním senzorem
- střední interaktivní servomotor s integrovaným rotačním senzorem
- infračervený senzor
- světelný a barevný senzor
- gyroskop
- ovladač
- spojovací vodiče s konektory
- technické prvky a stavební materiál



Obrázek 3: Stavebnice LEGO MINSTORMS, zdroj vlastní

Obsah stavebnice VEX IQ Super Kit:

- více než 800 dílů
- mozek robota
- ovladač
- 4 motory
- gyrosenzor
- spínač nárazníku
- dotyková LED
- snímač vzdálenosti
- barvený senzor
- spojovací vodiče s konektory
- technické prvky a stavební materiál



Obrázek 4: Stavebnice VEX IQ, zdroj vlastní

3.4 Programování a programová prostředí

Obě robotické stavebnice můžeme programovat v různých programovacích prostředích.

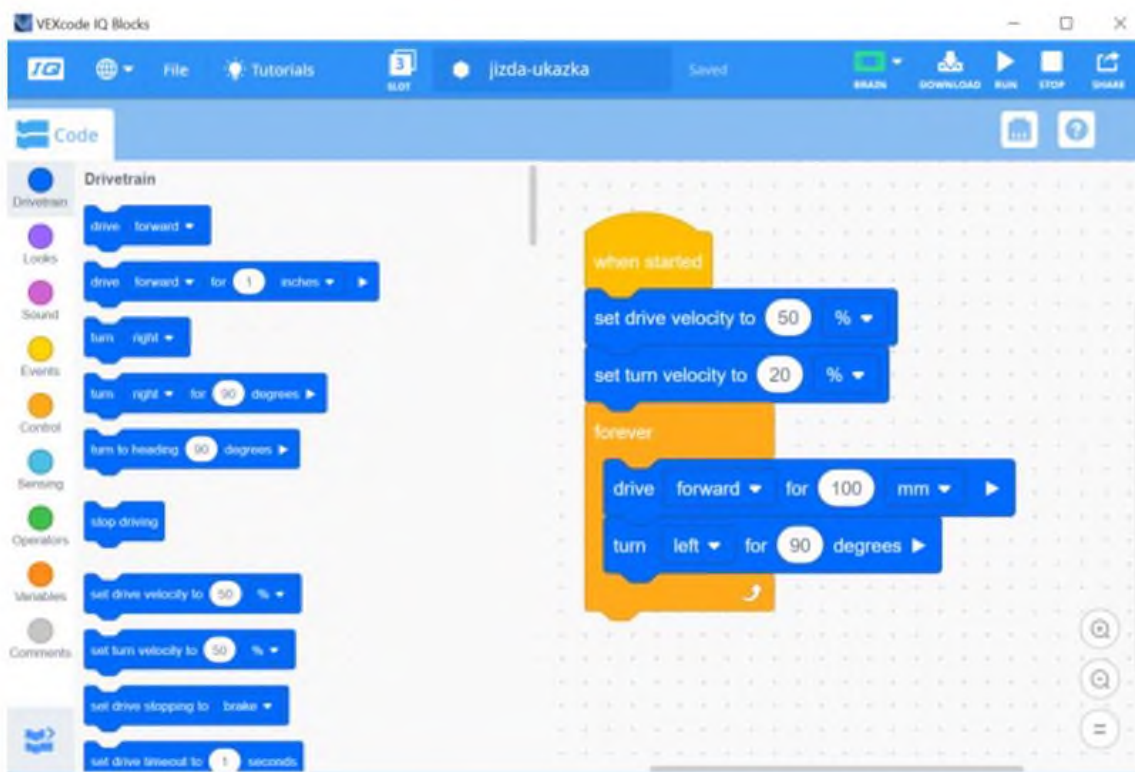
První možností programování s robotickou stavebnicí LEGO je ikonografické. Program sestavujeme z bloků a všechny vlastnosti nastavujeme na daném bloku.



Obrázek 5: Program LEGO MINDSTORMS, zdroj vlastní

Další možností je programování pomocí Pythonu, přesněji Pythonu for EV3. K programování můžeme využít i MakeCode či RobotC.

Programy a programová prostředí, která využíváme ve stavebnici VEX IQ, jsou VEXcode IQ či Robot C. Do července 2019 se využíval k programování program Modkit VEX. Toto programovací prostředí bylo nahrazeno novým prostředím VEXcode IQ. V programovém prostředí VEXcode IQ programujeme pomocí bloků či programových bloků (SCRATCH). V současné době existuje pouze blokové prostředí. Začátkem roku 2020 by mělo být zpřístupněno i textové prostředí. Dalšími možnostmi jsou Robot Mesh Studio a EasyC V6.



Obrázek 6: Program VEXcode IQ, zdroj vlastní

Obě robotické stavebnice nám nabízejí různé možnosti programování. Můžeme tak rozvíjet informatické myšlení u žáků různým způsobem.

4 SROVNÁNÍ STAVEBNIC LEGO MINSTORMS EDUCATION EV3 A VEX IQ SUPER KIT

Obě stavebnice LEGO MINSTORMS Education EV3 a VEX IQ Super Kit jsou využívány na základních školách. Stavebnice LEGO MINSTORMS Education EV3 je na trhu déle a je zastoupena ve školách více. Stavebnice VEX IQ Super Kit není na trhu tak dlouho a své místo si ve školách teprve získává. S tím i souvisí skutečnost, že materiálů ke stavebnici LEGO MINSTORMS Education EV3 je více.

Obě stavebnice jsou rovnocenné. Liší se v řídicí jednotce, resp. v zapojení motorů a čidel. U mozku robota VEX IQ Super Kit nezávisí na připojení, všechny porty jsou rovnocenné. Při zapojování kostky ve stavebnici LEGO máme porty pro motory a porty pro čidla. Z obou stavebnic můžeme stavět různé roboty, můžeme dokupovat a zapojovat další komponenty.

Obě stavebnice můžeme ve školách rovnocenně využívat. Nelze říci, která se stavebnic se hodí k některým činnostem více. K oběma stavebnicím jsou zpracovány materiály. Materiály můžeme využít k práci s oběma stavebnicemi. Metodiky stavebnice LEGO MINSTORMS Education EV3 jsou dobrým námětem pro práci se stavebnicí VEX IQ Super Kit a opačně. Učitelé je mohou využít pro další práci s danou robotickou stavebnicí.

5 MOJE PRVNÍ ZKUŠENOSTI

První použití sestaveného robota ze stavebnice VEX IQ bylo v hodině fyziky v 7. ročníku. Podle návodu jsem sestavila robota s názvem REX.

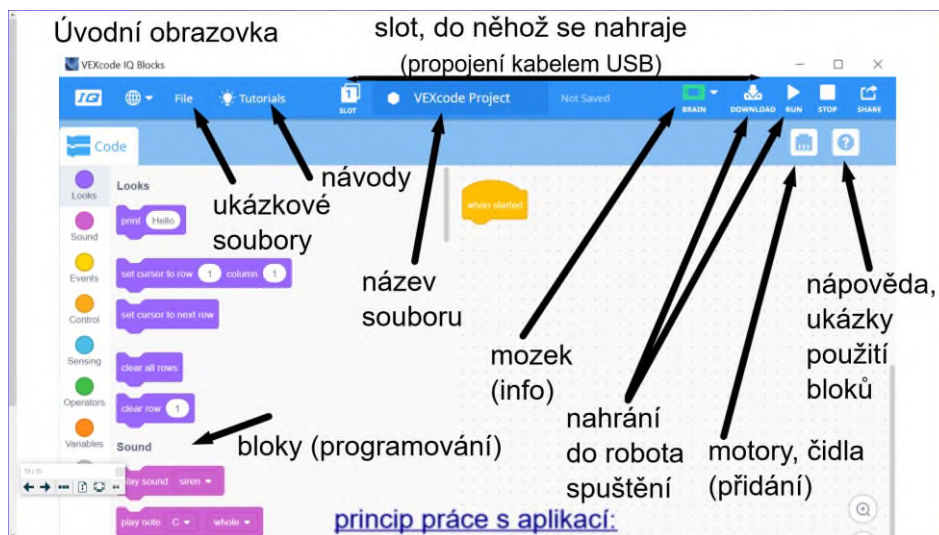
V této hodině jsem pustila program autopilota. Robota jsem ovládala pomocí ovladače. Žáci měli za úkol najít na robotu všechny jednoduché stroje (kladka, páka, klín, kolo, ozubené kolo, soukolí, ...) a sdělit, k čemu v robotu daný jednoduchý stroj slouží, k čemu ho robot využívá. Další část otázek byla zaměřena na těžiště a stabilitu tělesa. Poslední kapitola a otázky v této hodině se zaměřovala na Newtonovy zákony. I takovýto robot se může stát pomocníkem v domácnosti, např. při vytírání podlahy.



Obrázek 7: Robot REX pomocník v domácnosti, zdroj vlastní

V další hodině fyziky měli žáci za úkol postavit řetězovou dráhu. Úkolem bylo tuto dráhu postavit, rozpohybovat, popsat, vyřešit úkoly v pracovním listě. Zde byly propojeny mezipředmětové vztahy. S robotickou stavebnicí pracují žáci převážně v klubu robotiky. Zde je na tuto práci mnohem více času než v hodinách informatiky. Různé robotické hračky i stavebnice využíváme i v hodinách informatiky či fyziky.

Žáci nejdříve sestavují jednoduchý podvozek. Programují jej v programovacím prostředí VEXcode IQ. Nejdříve se seznamují s programovacím prostředím VEXcode IQ a jeho možnostmi.



Obrázek 8: Program VEXcode IQ, zdroj vlastní

Na daném podvozku se učí programovat motory a jízdu. Učí se nastavit rychlost otáčení a pohon motorů. Cykly již žáci znají z programování ozobotů či programovacího prostředí Scratch. Poté připojují jednotlivé senzory, které se učí programovat jednoduchým způsobem. Nejdříve je to dotyková LED dioda, potom senzor barev. Senzor barev užívají k jízdě po čáře. K dalším aktivitám patří programování senzoru vzdálenosti, senzoru nárazníku a gyrosenzoru. Následují programy, ve kterých využívají více senzorů najednou. K práci využívají již připravené materiály – robotická ruka, třídící robot apod. či materiály, které sami vytvoří.

6 ZÁVĚR

Při konstrukčních činnostech s robotickými stavebnicemi rozvíjíme u žáků manuální zručnost, při které je nutná trpělivost, přesnost, pečlivost a soustředěnost. Při programování robotů rozvíjíme logické myšlení či kreativitu. U všech činností žáci pracují ve skupinách, v týmech. 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.

Robotické stavebnice LEGO MINDSTORMS Education EV3 a VEX IQ Super Kit jsou srovnatelné cenově, technicky i využitelnosti při výuce. Obě robotické stavebnice můžeme programovat v různých prostředích. Společným programovacím prostředím je RobotC. Materiály, které jsou připraveny, můžeme využít u obou robotických stavebnic.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] Robotika na 2. stupni základní školy s LEGO Mindstorms. *Informatické myšlení*. [online]. Jihočeská univerzita v Českých Budějovicích. [vid. 10. 12. 2019]. Dostupné z: <https://imysleni.cz/ucebnice/robotika-na-2-stupni-zakladni-skoly-s-lego-mindstorms>. Jihočeská univerzita, 2017.
- [2] Návod. *Metodika VEX IQ*. [online]. AV MEDIA. [vid. 10. 12. 2019]. Dostupné na: <https://www.veskole.cz/?url=navody&strana=2>. AV MEDIA, 2020.
- [3] NEUMAJER, Ondřej. *Inovativní výukové aktivity pro rozvoj dovedností pro 21. století*. Praha: Univerzita Karlova, Pedagogická fakulta, 2014. 6–12 s. Dostupné online. ISBN 978-80-7290-653-6, ISBN 80-7290-653-4.
- [4] Jaké kompetence u absolventů očekávají zaměstnavatelé v průmyslu a ve stavebnictví. *Absolventi škol a trh práce* [online]. Národního pedagogický institut České republiky. [vid. 10. 12. 2019]. Dostupné na: <https://infoabsolvent.cz/Temata/ClanekAbsolventi/4-1-14/Jake-kompetence-u-absolventu-ocekavaji->. Národní pedagogický institut České republiky, 2013.

Simulácia procesov inteligentnej domácnosti s micro:bitom

The Simulation of Smart Home Processes with a Micro:bit

Martin Cápay
Katedra informatiky
Fakulta prírodných vied UKF v Nitre
Trieda A. Hlinku 1
949 01, Nitra
Slovensko
mcapay@ukf.sk

Magdaléna Bellayová
Fakulta informačních technologií
Vysoké učení technické v Brně
Božetěchova 2/1
612 00 Brno
Česká republika
xbella01@stud.fit.vutbr.cz

ABSTRAKT

Mikrokontrolér BBC micro:bit, programovateľný hardvér, vyvinutý pre edukačné účely, má v sebe zabudovaný svetelný a teplotný senzor. S ich využitím môžeme simulovať základné procesy v SMART domácnosti ako automatizované zhasínanie svetla, či spúšťanie klimatizácie. Pripojením micro:bita k rozširujúcej doske, pripojením externých senzorov a prídavných zariadení sa podstatne rozšíria možnosti micro:bita reagovať na dáta získané z reálneho sveta. Cieľom príspevku bude popísať sériu aktivít simulujúcich procesy v inteligentnej domácnosti s využitím integrovaných senzorov a senzorov zo sady Smart Home Kit.

ABSTRACT

The BBC micro:bit, a programmable hardware developed for educational purposes, has a built-in a light and temperature sensor. Using them, we can simulate basic processes in SMART houses such as automated switching off the light or starting air conditioning. By connecting the micro:bit to the expansion board, by connecting external sensors and attachments, the microbit will significantly expand the capabilities to respond to data from the real world. The aim of the paper will be to describe a series of activities simulating processes in a smart home using integrated sensors and sensors from the Smart Home Kit.

Klíčová slova

micro:bit, inteligentná domácnosť, senzory, automatizácia

Keywords

micro:bit, smart home, sensors, automatization

1 ÚVOD

Mikropočítačová doska micro:bit sa v čoraz väčšej miere využíva aj na našich školách. Žiaci a študenti môžu cez ovládanie hardvérových komponentov snímať a spracovávať údaje so senzorov, ovládať motorčeky, LED diódy, komponovať hudbu, či vytvárať komplexnejšie projekty ako napríklad iMath vhodný na precvičovanie základov matematiky, či model inteligentného domu. Konštrukcia riešenia, ktoré je vidieť, je možné s ním manipulovať a „dotknúť sa“ ho je pre žiaka motivujúce a názorné zároveň. BBC micro:bit je možné pripájať na rozširujúce špeciálne moduly

(napríklad externé senzory, wifi) alebo univerzálne moduly s prepojavacím poľom, cez ktoré sa pomocou káblov dá základná doska rozšíriť o množstvo doplnujúcich komponentov. Našou víziou je aby sa na hodinách programovania čo najviac pracovalo s reálnymi a vlastnými dátami získanými cez senzory z okolitého prostredia, aby sa prepájali projekty s vlastnými záujmami a záľubami, a aby programovanie dostalo reálny a hmatateľný význam a zmysel pre väčšinu žiakov. Cieľom tohto príspevku je poskytnúť praktické návody na zaujímavé a pritom jednoduché riešenia využívajúce mikropočítačové dosky, ktoré sa dajú aplikovať pri výučbe informatiky na mnohých našich školách. Budeme riešiť otázku: „Ako funguje inteligentná domácnosť?“. Témou inteligentné domácnosti (SMART houses) chceme priblížiť fungovanie senzorov pri automatizovanom spúšťaní alebo vypínaní komponentov, ako napr. rozsvetovanie svetiel, zapínanie klimatizácie, či zavlažovanie kvetov. Článok neposkytuje vedeckú štúdiu, ale poskytuje námety a overenú sadu úloh pre mikrokontrolér BBC micro:bit. Poskytuje aktivity simulujúce činnosť Smart Home technológie vhodné pre školské prostredie.

2 PHYSICAL COMPUTING

Dnešné technológie umožňujú vyrobiť počítače, senzory a ovládače dosť malé na to, aby sa dali umiestniť napr. do hračiek. Projekty, kde je možné technológie skombinovať s tradičnými materiálmi ako papier, či textilie nám dávajú možnosť viac zapájať deti do programovania na neformálnej úrovni, cez prístupnejšiu, prirodzenú formu vzdelávania. [1] Programovaniu malých mikropočítačov s rôznymi senzormi, kde programy zaznamenávajú dáta získané zo svojho vlastného okolia (teplota, vlhkosť, pohyb) a vhodným spôsobom na ne reagujú, sa venuje tzv. physical programming (ďalej PP). Žiak, programátor, nepracuje iba so vstupmi a výstupmi na obrazovke, ale rieši reálne problémy. Programy už nie sú len texty a čísla. PP môže žiakov tak vo výsledku motivovať k programovaniu prostredníctvom riešenia hmatateľných projektov [2] s využitím programovateľného hardvéru so senzormi (meteostanica, sledovacie zariadenie, zavlažovacia stanica, detektor pohybu a pod.). Študent má pri PP možnosť sa viac zamerať na realizáciu svojich nápadov ako na sledovanie obmedzení, ktoré často súvisia s tradičným prístupom v programovaní [3, 4]. Programovanie hardvéru je viac než kódovanie, súvisí s reálnymi programovateľnými interaktívnymi systémami alebo zariadeniami. Je to prepojenie medzi fyzickým svetom a virtuálnym svetom počítača. [5] Ak školopovinným deťom poskytnete správne nástroje a platformy, môžu si v hale vybudovať inteligentný zavlažovací systém pre umierajúci strom v kvetináči, či implementovať snímač, ktorý sleduje aktivitu psa, keď nie je doma. Je to motivujúce, byť schopný vytvoriť niečo nové, čo má preukázateľnú hodnotu, či dokonca vytvoriť niečo, čo vaši priatelia predtým ani nevideli [7].

2.1 BBC micro:bit

Školský mikropočítač, ako môžeme BBC micro:bit nazvať, je výsledkom iniciatívy zastrešenej britskou verejnoprávnou televíziou, ktorej cieľom bolo vytvorenie lacnej platformy na výučbu informatiky, ktorý sa má stať aj povinnou súčasťou informatického kurikula [7]. V roku 2016, tak každý žiak siedmeho ročníka v Spojenom Kráľovstve dostal BBC micro:bit, mikropočítač polovičnej veľkosti kreditnej karty, ktorý sa pohodlne zmestí do vrečka košeľe. Micro:bit má zanedbateľnú hmotnosť a vďaka množstvu zabudovaného hardvéru aj variabilné možnosti využitia. Micro:bit má dve programovateľné tlačidlá, maticový displej z 25 LED diód, senzor pohybu, snímač naklonenia, kompas, senzor teploty a intenzity svetla. Na micro:bit-e sú v skutočnosti dva typy procesorov. Kód nahrávaný do KL26Z sa automaticky nahráva do procesora nFR51 (ARM Cortex-M0 32 bit procesor, Flash ROM 256KB, RAM 16KB, 16MHz). Po skopírovaní sa hex kód v micro:bitovi neukladá a tak nie je možné micro:bit použiť ako prenosné zariadenie. Procesor KL26Z obsahuje zabudovaný regulátor, ktorý znižuje napätie USB na 3,3V, ktoré je vhodné na napájanie zvyšku micro:bit-u, a z tohto regulátora procesora môžeme čerpať 120mA. Tento procesor nemá žiadne pripojenie k pinom GPIO na micro:bit-e. Hlavný procesor nRF51 spúšťa beh programu. Všetky GPIO piny na sú obsluhované týmto procesorom. Micro:bit má 3 veľké vstupno výstupné piny, napájací pin (3V), zemnenie (GND) a 20 malých pinov, z ktorých niektoré sú zdieľané hardvérovými komponentami.

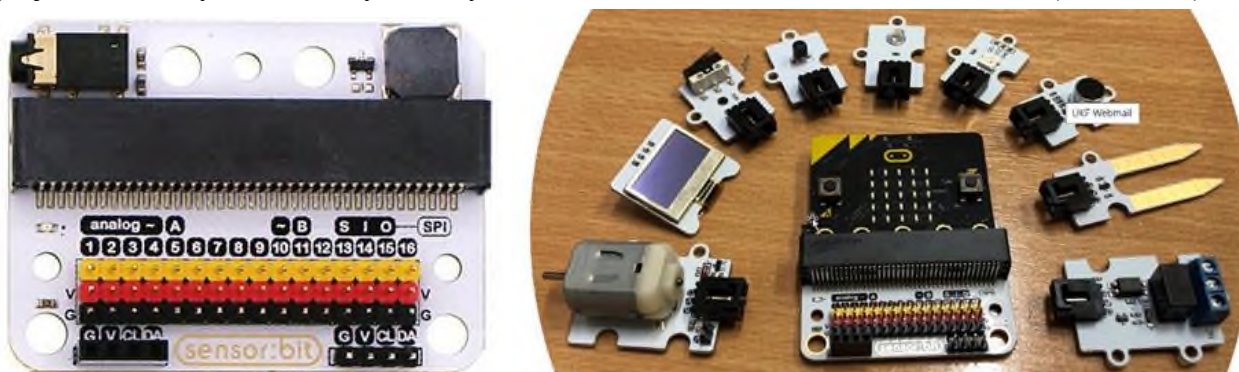
K veľkým pinom s pokovanými otvormi je možné pripnúť krokosvorkové vodiče alebo vodiče s „banánikmi“. Možnosť bezdrôtovo prepojiť viac micro:bit dosiek rozširuje množstvo, pre žiakov, atraktívnych a interaktívnych scenárov. K dispozícii je aj veľké množstvo príslušenstva, ktoré okrem iného umožňuje pripojiť BBC micro:bit k wifi a povýšiť túto školskú mikropočítačovú dosku na plnohodnotnú IoT platformu [6] [9]. Žiaci používajú BBC micro:bit vo vzťahu k programovaniu a riešeniu problémov za užitočnú učebnú pomôcku [8].

3 SMARTHOME

Smarthome je technológia, ktorá klasickej domácnosti umožní, aby sa stala inteligentnou a automatizovanou. Táto technológia má zvyčajne automatizovaný systém na osvetlenie, kontrolu teploty a bezpečnosť [10]. Cenovú prístupnosť Smart Home technológie je možné dosiahnuť použitím WLAN lokálnej siete založenej na technológii mikrokontroléra Arduino [11][12][13]. Spoločný prienik všetkých riešení je založený na tom, že potrebujeme sadu senzorov a mikrokontrolér, ktorý v nekonečnej slučke kontroluje vstupy získané zo snímačov a na základe vopred navolených inštrukcií na ne reaguje.

3.1 S čím sme pracovali (Sensor:bit doska, senzory)

Modulárnosť mikropočítačovej dosky BBC micro:bit nám dáva možnosti pripojenia k špecializovaným doskám, ktoré úzke kontakty plošného spoja vyvedie na špeciálne piny, cez ktoré môžeme pomocou prepojovacích vodičov s dutinkami pripojiť rozširujúci hardvér. My sme využili „ElecFreaks sensor:bit for micro:bit“. Sensor:bit rozširuje všetky dostupné IO porty na micro:bit a vyvedie ich vo forme GVS. Doska tiež ponúka možnosť pripojenia aj cez IIC. Na doske je integrovaný bzučiak a zvukový konektor, na pripojenie slúchadiel alebo reproduktorov. Na realizáciu projektov sme využili senzory zo sady ELECFREAKS micro:bit Smart Home Kit (Obrázok 1).



Obrázok 1: Sensor:bit doska a komponenty zo sady Smart Home Kit

4 INTELIGENTNÁ DOMÁCNOSŤ S MICRO:BITOM

Každú úlohu sme si rozdelili na nadväzujúce podúlohy, kde sme postupne pripájali komponenty, prezentovali ako fungujú a zároveň sme sa snažili nájsť názorné prepojenie s reálnymi procesmi. Následne sme sa snažili komponenty spojiť do väčšieho celku a naprogramovať tak, aby sme dosiahli automatizáciu procesov.

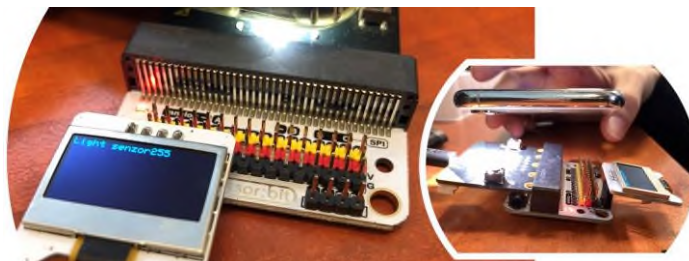
4.1 Inteligentné osvetlenie

Čo sme použili: senzor intenzity osvetlenia (light sensor), programovateľná LED (rainbow LED), senzor intenzity zvuku (noise sensor), OLED displej.

4.1.1 Zabudovaný senzor intenzity svetla

Aby sme prezentovali, že niektoré úlohy je možné realizovať aj bez sady Smart Home Kit, tak sme najskôr ukázali prácu so zabudovaným senzorom. Pri tejto aktivite sa účastníci snažili nájsť, kde na micro:bit sa senzor nachádza, ako citlivý je a aké hodnoty sa dajú namerať. Tu sme si aj prvýkrát

DIDINFO 2020, Technická univerzita v Liberci, Fakulta prírodovedne-humanitná a pedagogická ukázali vypisovanie hodnôt na OLED displeji, ktorý sme si zapojili k sensor:bit doske cez IIC rozhranie (Obrázok 2). OLED displej sme v úlohách používali kvôli lepšej prehľadnosti výpisu, aj možnosti jednorazového zobrazenia väčšieho množstva dát.



Obrázok 2: Zobrazovanie hodnôt zo zabudovaného senzora intenzity svetla

Čo sme sa naučili:

- Zabudovaný senzor nie je veľmi citlivý na rozptýlené svetlo. V skutočnosti micro:bit nemá vstavaný plnohodnotný senzor. Meria sa len intenzita svetla dopadajúce na vstavaný LED displej.
- Snímač nám vracia hodnoty v intervale 0–255.

4.1.2 Externý senzor intenzity svetla zo Smart Home Kitu

V ďalšej časti sme pripojili už aj externý senzor cez GVS na sensor:bit dosku. Aj v tomto prípade sme použili prehľadnejší výpis hodnoty na OLED displej. Rovnako ako v prvej úlohe účastníci skúmali citlivosť senzora a rozsah v akom senzor meria.

Čo sme sa naučili:

- Externý senzor je veľmi citlivý na rozptýlené svetlo. V rôznych častiach učebne snímač dával rôzne hodnoty. Bolo to spôsobené tým, že niektorí účastníci sedeli pri okne, kde dopadalo priame denné svetlo, niektorí zase sedeli pod umelým osvetlením. Zdôraznili sme, že v ďalších úlohách musia na rozdielnosť v nameraných hodnotách vopred myslieť.
- Snímač nám vracia hodnoty v intervale 0–100.

4.1.3 LED svetielko

V tejto aktivite sme na dosku pripojili už aj naše „osvetlenie“ v podobe jedného programovateľného LED. Ukazovali sme si ako ho inicializovať, rozsvietiť a zhasnúť. Účastníci si potom skúšali svoje svetielko naprogramovať tak, aby ho vedeli zapnúť a vypnúť pomocou tlačidiel na micro:bitovi. Pri tejto aktivite sme ukázali aj možnosť namiešania si vlastnej farby svetielka pomocou RGB.

Čo sme sa naučili:

- Naučili sme sa zažínať a zhasínať svetlo podobne ako pri klasickom vypínači.
- LED pásik musí byť inicializovaný na presný počet diód (v našom prípade len jedna) a konkrétny pin.
- Klasické biele svetlo vznikne keď sú všetky časti svetla na plnej intenzite (255, 255, 255).
- Vypnutie simulujeme nastavením farby na čiernu, kde všetky zložky svetla sú na minime (0, 0, 0)

4.1.4 Automatické osvetlenie na intenzitu svetla

Automatizáciu sme zaviedli spojením predchádzajúcich častí. Už sme nerozsviecovali svetielka manuálne pomocou tlačidiel, ale LED svetielko sa rozsvietilo na základe nameranej intenzity svetla. Takto sme v podstate simulovali automatické pouličné osvetlenie, ktoré sa zapne v momente, keď sa zotmie. Hodnotu intenzity svetla pri ktorej sa svetielko rozsvietilo si určil, každý účastník sám, na základe predošlých pozorovaní.

Čo sme sa naučili:

- Automatizácia funguje tak, že systém v nekonečnej slučke meria vstupy so senzora. V prípade, ak hodnota zo senzora splní žiadanú podmienku, vykoná sa predprogramovaná úloha.
- Keď chceme hodnotou zo senzora použiť viackrát, musí sa uložiť do samostatnej premennej.

4.1.5 Senzor intenzity zvuku

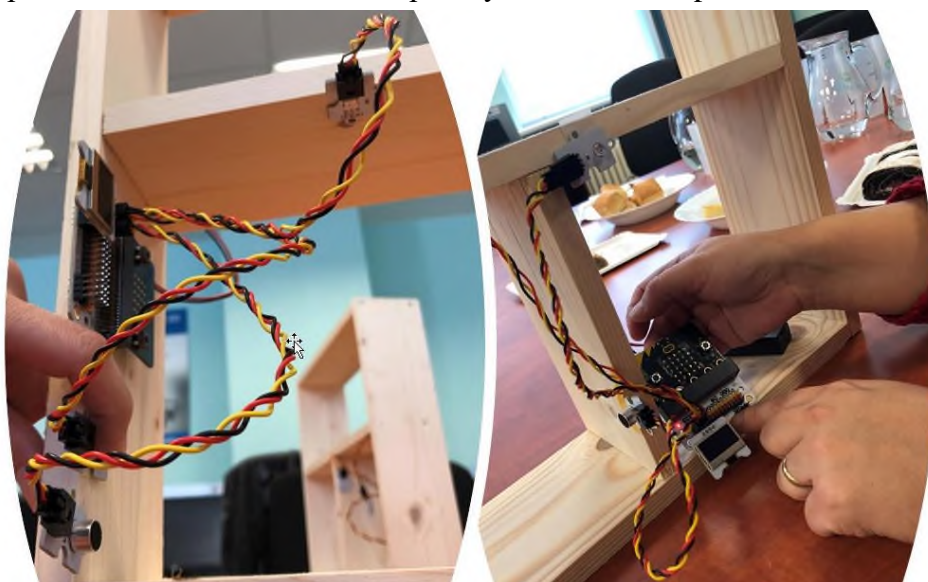
Ďalší komponent, ktorý sme pridali na senzorickú dosku bol senzor intenzity zvuku. Tu sme tiež experimentovali s citlivosťou a možným rozsahom nameraných hodnôt.

Čo sme sa naučili:

- Zvukový senzor meria hodnoty v decibeloch (dB). Nerozpoznáva slová, ale reaguje len na hlučnosť v okolí senzora.
- Snímač nám vracia hodnoty v intervale 0–100.

4.1.6 Automatické osvetlenie reagujúce na intenzitu svetla aj zvuku

Automatické osvetlenie sme zefektívnili pridaním senzora intenzity zvuku (Obrázok 3). Keďže doma nepotrebujeme, aby nám bolo svetlo zapnuté celú noc, tak potrebujeme ďalší podnet, na základe ktorého sa bude svetlo zapínať. Zaviedli sme ďalšiu podmienku rozsvetovania a to intenzitu zvuku nad určitou hodnotou, ktorú si zase určili účastníci. Na konci aktivity sme mali osvetlenie, ktoré sa po zotmení zaplo na niekoľko sekúnd ale len pri zvýšenom hluku a potom sa zhaslo.



Obrázok 3: Zapojenie pre automatické osvetlenie reagujúce na intenzitu svetla aj zvuku

Čo sme sa naučili:

- Automatizácia môže podliehať viacerým podmienkam. V našom prípade sa svetlo zažne len ak je vonku tma a zároveň bol vydaný zvukový povel (potlesk, dupnutie, či slovo). Aj to len na krátky čas. Inak nemá svietenie význam.

4.2 Inteligentná klimatizácia

Čo sme použili: jednosmerný motor (DC motor) s vrtuľkou, teplotný senzor (temperature sensor), OLED displej.

4.2.1 Spúšťanie „klimatizácie“

Najprv sme si zapojili jednosmerný motor a na neho vrtuľku, čo predstavovalo našu „klimatizáciu“. Motor má len dva stavy a to buď stojí alebo sa točí. Týmto stavom sme priradili čísla a to 0 pre stav kedy stál a 1 pre stav kedy sa točil. Tieto hodnoty sme potom digitálne zapisovali na pin, kde sme

mali motor zapojený. Hodnoty sme na pin zapisovali pri stlačení tlačidla. Simulovali sme tak zapnutie a vypnutie „klimatizácie“ manuálne cez vypínač.

Čo sme sa naučili:

- Na piny môžeme zapisovať digitálne hodnoty 1 alebo 0. V skutočnosti sa na pin privedie nejaké napätie.

4.2.2 Zabudovaný teplotný senzor

Najprv sme si ukázali prácu so zabudovaným teplotným senzorom. Pri tejto aktivite sa účastníci snažili nájsť, kde na micro:bitovi sa senzor nachádza, ako citlivý je a aké hodnoty sa dajú namerať. Hodnoty sme si zase dali vypisovať na OLED displeji.

Čo sme sa naučili:

- Zabudovaný senzor nie je veľmi citlivý na okolité teplo. V skutočnosti tam nie je žiaden reálny senzor. Procesor ARM, ktorý je zabudovaný aj v micro:bitovi, pracuje „za studena“ a tak má len takú teplotu aká je teplota okolia. Micro:bit teda len meria teplotu procesora. V prípade, ak používateľ zakryje prstom procesor, zvýši sa aj teplota „senzora“.
- Meria sa len teplota procesora, ktorá je o pár stupňov (cca o 4 °C) vždy vyššia ako teplota prostredia.
- Snímač nám vracia hodnoty v intervale 0–50 °C.

4.2.3 Externý tepelný senzor zo Smart Home Kitu

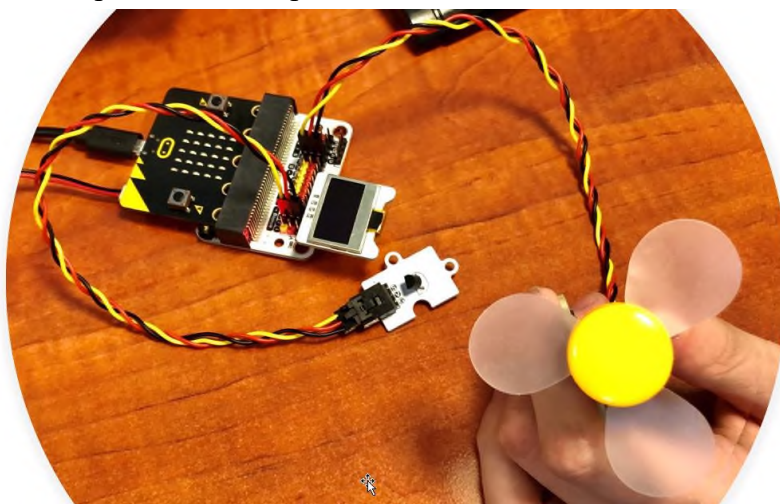
V ďalšej časti sme si pripojili externý teplotný senzor na dosku. Aj v tomto prípade sme použili na výpis hodnoty OLED displej. Účastníci zisťovali citlivosť senzora a rozsah v akom senzor meria.

Čo sme sa naučili:

- Externý senzor je presnejší a na teplotu reaguje citlivejšie.

4.2.4 Automatické zapínanie „klimatizácie“

Pri automatizácii sme zase všetky časti spojili, čiže manuálne spúšťanie motora sme nahradili podmienkou automatického zapnutia. Podmienku, pri akej teplote sa spustí „klimatizácia“ si zase určili účastníci. Po dosiahnutí určitej teploty sa motorček na niekoľko sekúnd zapol a zase vypol, toto sa opakovalo, až kým sa teplota neznížila pod nastavenú hodnotu.



Obrázok 4: Zapojenie pre automatické zapínanie „klimatizácie“

Čo sme sa naučili:

- Opäť sme si overili, že automatizácia musí prebiehať v slučke, v ktorej sa reaguje na hodnoty zo senzora. Podľa hodnoty z teplomera sa automaticky na pin zapisuje buď 1 alebo 0.

4.3 Inteligentný Alarm

Čo sme použili: senzor nárazu (crash sensor), OLED displej.

4.3.1 Senzor nárazu

Najprv sme si zapojili senzor nárazu. Tento senzor má tiež ako jednosmerný motor dva stavy, ktoré vieme vyjadriť digitálne a teda aj čítať pomocou 1 a 0 (0 pre stav kedy je senzor stlačený a 1 pre stav kedy nie je stlačený). Stav senzora sme vypisovali na OLED displej.

Čo sme sa naučili:

- Nárazový snímač pri stlačení zapisuje na pin hodnotu 0, inak zapisuje hodnotu 1. Preto je potrebné nastaviť pin, na ktorom je pripojený. Hodnotu na pine môžeme čítať a spracovať.

4.3.2 Automatický alarm

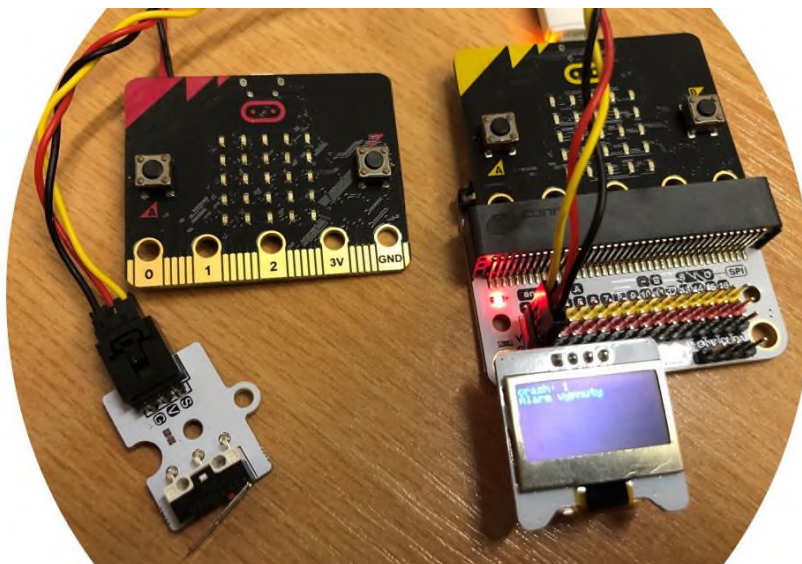
V tomto kroku sme si senzor nárazu predstavili ako dvere (alebo akoby bol na dvere pripevnený). Keď sú dvere zatvorené, tak je aj spínač stlačený. Keď sa dvere otvoria, tak sa spínač tiež „otvorí“. Na základe toho sme si nastavili podmienky, keď sú dvere otvorené alarm je zapnutý a keď sú zavreté tak je vypnutý.

Čo sme sa naučili:

- Test stlačenia senzora môžeme vykonať aj cez stav „crash sensor pressed“, ktorý má hodnotu True alebo False.

4.3.3 Automatický alarm s diaľkovým ovládaním

V poslednej časti sme uvažovali ako funguje skutočný alarm. Ten musí viesť majiteľ domu zapnúť, keď ide z domu preč, ale aj vypnúť, keď je doma aby sám alarm nespustil. Ďalšia vec, nad ktorou sme uvažovali bola, že keď niekto spustí alarm tým že otvorí dvere, nechceme aby ho dokázal vypnúť len tým, že ich za sebou zavrie. Deaktivovať alarm by mal teda len majiteľ domu. Na toto všetko sme si pomocou druhého micro:bita vytvorili diaľkový ovládač, ktorým sme vedeli alarm aktivovať a deaktivovať a upravili aj náš pôvodný automatický alarm (Obrázok 5).



Obrázok 5: Zapojenie pre automatický alarm s diaľkovým ovládaním

4.4 Inteligentné zavlažovanie

Čo sme použili: senzor vlhkosti pôdy (soil moisture sensor), relátko (relay), čerpadlo (submersible pump), OLED displej.

4.4.1 Senzor vlhkosti pôdy

Najprv sme si pripojili senzor na dosku a ukázali si dva spôsoby ako z neho čítať hodnoty. Prvý bol cez príkaz, ktorý je určený pre prácu s týmto kitom a ďalší cez analógové čítanie z pinu, na ktorom je senzor pripojený. Účastníci zisťovali citlivosť senzora a rozsah v akom senzor meria vlhkosť pôdy v kvetináči. Aj v tomto prípade sme použili na výpis hodnoty OLED displej.

Čo sme sa naučili:

- Snímač vlhkosti pôdy pôsobí ako rezistor. Čím viac vody je v pôde, tým je lepšia vodivosť medzi kolíkmi, čo spôsobí aj zníženie odporu a vyššie namerané hodnoty.
- Pri použití príkazu z knižnice smarthome meria senzor v rozsahu 0–100 a pri použití analógového čítania 0-1023.

4.4.2 Spúšťanie čerpadla

Ako ďalšie sme zapojili čerpadlo, pri ktorom sme museli použiť relátko, aby sme dokázali čerpadlo poháňať. Tak ako motor, tak aj čerpadlo má dva stavy a to buď čerpá alebo nečerpá. Týmto stavom sme si zase priradili čísla a to 0 pre stav kedy nečerpá a 1 pre stav kedy čerpá a tieto hodnoty sme potom digitálne zapisovali na pin, kde sme mali čerpadlo pripojené. Hodnoty sme na pin zapisovali pri stlačení tlačidla. Simulovali sme tak manuálne spúšťanie a vypínanie čerpadla.

Čo sme sa naučili:

- Úlohou relátka je spúšťať a vypínať ponorné čerpadlo, pôsobí ako spínač. Relátko ovláda zdroj energie pre vodnú pumpu. Keď je na pine hodnota 1 pumpa je zapnutá, keď je na pine 0 je naopak vypnutá.

4.4.3 Automatické zavlažovanie

Spojením týchto dvoch komponentov sme vytvorili automatické zavlažovanie, ktoré sa spúšťalo ak vlhkosť pôdy klesla pod hodnotu, ktorú si určili účastníci. Po klesnutí pod túto hodnotu sa čerpadlo zaplo na pár sekúnd a potom sa zase vyplo a to sa opakovalo kým hodnota dostatočne nestúpila.



Obrázok 6: Zapojenie pre automatické zavlažovanie

5 ZÁVER

Všetky popísané aktivity boli overené v rámci celodenného workshopu pre učiteľov v Klube informatikov pod záštitou občianskeho združenia eTeacher (www.eteacher.sk) a priebežne realizované aj v rámci detského krúžku. Z dotazníkovej spätnej väzby vyplynulo, že učiteľom sa najviac páčila aktivita s vodnou pumpou a alarm. Pozitívne hodnotili praktickosť úloh, reálne ukážky práce so senzormi a odskúšanie „vlastnými rukami“. Výhody použitia samotného micro:bita vo

DIDINFO 2020, Technická univerzita v Liberci, Fakulta přírodovědně-humanitní a pedagogická
vzdelávaní vidia vo viditeľnom fyzickom výsledku, kombinácii práce s hardvérom a jeho následným
programovaním, priestoru na využitie medzipredmetových vzťahov a realizáciu žiackych projektov
a ich prezentáciu a najmä v odpovedi na univerzálnu otázku žiakov „Na čo mi to bude?“.

6 POĎAKOVANIE

Za podporu ďakujeme Komenskému inštitútu a Katedre informatiky FPV UKF v Nitre.

7 BIBLIOGRAFICKÉ ODKAZY

- [1] BLIKSTEIN P. 2013. Gears of our childhood: constructionist toolkits, robotics, and physical computing, past and future. In Proceedings of the 12th International Conference on Interaction Design and Children (IDC '13). Association for Computing Machinery, New York, NY, USA, 173–182.
- [2] CÁPAY, M., KLIMOVÁ, N. Dokáže programovanie hardvéru zaujať študentov? In. DidInfo 2019, Banská Bystrica, s. 52–57.
- [3] CÁPAY, M., BELLAYOVÁ, M. Aktivita s BBC micro:bit podporujúce kolaboratívnu prácu žiakov ZŠ, 2019. In. DidInfo 2019, Banská Bystrica, s. 47–51.
- [4] SENTANCE S., WAITE J., YEOMANS L., and MACLEOD E. Teaching with physical computing devices: the BBC micro:bit initiative, In Proceedings of the 12th Workshop on Primary and Secondary Computing Education (WiPSCE '17), Erik Barendsen and Peter Hubwieser (Eds.). ACM, 2017, New York, NY, USA, 87–96.
- [5] SCHULZ S. and PINKWART N. Physical Computing in STEM Education. In Proceedings of the Workshop in Primary and Secondary Computing Education (WiPSCE '15). ACM, 2015 New York, NY, USA, 134–135.
- [6] LACKO, Ľ. Programovanie hardvéru, robotika a IoT s Microbitom. Dostupné na: <https://www.pcrevue.sk/a/IoT-prakticky--E2-80-93-stiahnite-si-e-knihu-Programovanie-hardveru--robotika-a-IoT-s-Microbitom>.
- [7] BALL T. et al. Microsoft Touch Develop and the BBC micro:bit. 2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C), Austin, TX, USA.
- [8] SCHMIDT, A. Increasing Computer Literacy with the BBC micro:bit. IEEE. In Pervasive Computing, Volume 15, Issue 2, p. 5–7.
- [9] KNOWLES, B. , FINNEY, J. , BECK, S., DEVINE, J. What Children's Imagined Uses of the BBC micro:bit Tells Us About Designing for their IoT Privacy, Security and Safety. Conference: Living in the Internet of Things: Cybersecurity of the IoT – 2018.
- [10] SRIPAN, M., LIN, X., Ketcham, M. Research and Thinking of Smart Home Technology. In International Conference on Systems and Electronic Engineering (ICSEE'2012), December 18–19, 2012, Phuket, Thailand.
- [11] ADRIAMSYAH A., DANI A. D. Design of Small Smart Home system based on Arduino. In Electrical Power, Electronics, Communications, Control and Informatics Seminar (EECCIS) 2014, Malang, Indonesia.
- [12] BARAKA K., GHOBRIEL M., MALEK S. Low Cost Arduino/Android-Based Energy-Efficient Home Automation System with Smart Task Scheduling. In Fifth International Conference on Computational Intelligence, Communication Systems and Networks 2013.
- [13] A, J. CHANDRAMOHAN et al. Intelligent Smart Home Automation and Security System Using Arduino and Wi-fi. In International Journal Of Engineering And Computer Science. 2017.

Rozšířené abstrakty a názory

Extended Abstracts and Opinions

Zhmotňovanie nápadov pomocou digitálnych technológií výroby

Eva Kalužáková
CVTI SR
Lamačská cesta 8/A
811 04 Bratislava
Slovenská republika
eva.kaluzakova@cvtisr.sk

Jozef Vaško
CVTI SR
Lamačská cesta 8/A
811 04 Bratislava
Slovenská republika
jozef.vasko@cvtisr.sk

ABSTRAKT

Inšpirovaní skúsenosťami z katedier technológie a sociológie MIT a ich tvorivými dielňami ukážeme, ako sa rôznymi spôsobmi dá vytvoriť 3D model, ako ho je možno spracovať rôznymi technológiami digitálnej výroby, napríklad 3D tlačiareň, laserový vyrezávač alebo CNC fréza. Vysvetlíme princípy práce týchto strojov. Účastníci workshopu si môžu vyskúšať viaceré spôsoby modelovania v Tinkercade. Ukážeme aj možnosť vytvorenia kópie reálneho objektu (napr. sochy človeka) fotogrametriou napr. pomocou voľne dostupného softvéru ALICE VISION. Predstavíme kompletne návody na vytvorenie a programovanie robotických hračiek, ktorých základ sa dá vytlačiť na 3D tlačiarňu (pred tlačou ich je možné modifikovať práve v Tinkercade). Vytvorenie hračiek zvládajú deti od 10 rokov, zaujmú aj sedemnásťročných. Dievčatá si po vytvorení kraba MoKraRosa nahrajú do MP3 hudbu a naprogramujú si tanec, ktorý sa im páči. Pre iných je zaujímavý robotík OTTO, ktorého telo si môžu poskladať z dielov vyrezaných na laserovom vyrezávači alebo vytlačených na 3D tlačiarňu. Predstavíme aj ukážky prác detí na vinylovom vyrezávači a digitálnom vyšívacom stroji. Počas celého workshopu budeme tlačiť na 3D tlačiarňu malé motivačné darčeky, ktoré na školách rozdávame deťom a učiteľom.

ABSTRACT

Inspired by the MIT departments of technology and sociology and their creative workshops, we will show you how to create a 3D model in various ways, and how it can be processed by a variety of digital manufacturing technologies such as a 3D printer, laser cutter or CNC cutter. We will explain the principles of these machines. Workshop participants can try several modeling methods in Tinkercade. We will also show the possibility of making a copy of a real object (eg a human statue) by photogrammetry eg. with the free software ALICE VISION. We will present complete instructions for creating and programming robotic toys, the basis of which can be printed on a 3D printer (they can be modified before printing in Tinkercade). Children from 10 years of age can create toys and these toys are interesting also for seventeen-year-olds. After creating the MoKraRosa crab, the girls download MP3 music and program a dance they like. For others, the OTTO robot is interesting, whose body can be assembled from parts cut on a laser cutter or printed on a 3D printer. We will also present demonstrations of children's work on a vinyl cutter and digital embroidery machine. Throughout the workshop, we will print small motivational gifts on 3D printer that we give to children and teachers at schools.

Klíčová slova

digitálne technológie výroby, fotogrametria, Alice Vision, Tinkercad, Micro:bit,

Keywords

digital production technology, photogrammetry, Alice Vision, Tinkercad, Micro:bit

1 ÚVOD

V intenciách holistického vzdelávania J. H. Pestalozziho, *ktoré v harmónii rozvíja intelektuálne sily, morálne vlastnosti, sebahodnotu, prácu v kolektíve a praktické zručnosti*, sme pripravili workshop pre žiakov a študentov, ktorý je jednou z ukážok, ako ich motivovať k autentickému poznávaniu cez vlastnú skúsenosť. Naším cieľom je ukázať im nové možnosti ich uplatnenia o 5–10 rokov, na ktoré sa musia pripraviť už teraz. Dievčatám a chlapcom ukazujeme, ako je možné vďaka nadšeniu z digitálnych technológií zhmotniť svoj vlastný nápad. Na bezplatných 14-dňových alebo len 5-dňových táboroch si deti od 10 do 17,5 roka po zvládnutí základov 3D modelovania, programovania a robotiky, vytvoria pod vedením lektora svoju robotickú hračku – kraba MoKraRosa alebo robota OTTA.

2 PRIEBEH WORKSHOPU

Ukážeme, ako sa dá rôznymi spôsobmi vytvoriť 3D model, ako ho ďalej možno spracovať rôznymi technológiami digitálnej výroby, ako napríklad 3D tlačou, laserovým vyrezávačom alebo CNC frézou. Vysvetlíme princípy práce týchto strojov. Účastníci workshopu si môžu vyskúšať rôzne spôsoby modelovania v Tinkercade, ukážeme aj spôsob vytvorenia kópie reálneho objektu (napr. sochy človeka) fotogrametriou pomocou voľne dostupného softvéru ALICE VISION. Predstavíme aj kompletné návody na vytvorenie a programovanie robotických hračiek. Ich základ možno vytlačiť na 3D tlačiarňi (pred tlačou sa dajú modifikovať práve v Tinkercade) alebo poskladať z dielov vyrezaných na laserovom vyrezávači. Na jednotlivé dieliky si môžu napríklad napísať svoje motivačné heslá.

3 K REÁLNYM MODELOM BEZ SKENERA – POCHOPENIE PRINCÍPU FOTOGRAMETRIE

Rýchly nástup digitálnej fotografie približne od roku 2000 so sebou priniesol vzostup fotogrametrie. Je to metóda, vďaka ktorej možno jednoducho preniesť reálny objekt do počítača aj v takom prípade, že nemáte prístup k drahším 3D skenerom. Pomocou fotogrametrie sa dá vytvoriť trojrozmerný model z fotografií reálneho objektu, ktorý by nebolo možné inak jednoducho naskenovať (môže ísť aj o pomerne veľký objekt – napr. Beckovský hrad, ktorý máme vyfrézovaný a vytlačený 3D tlačou). Pri praktickom učení na workshope urobíme 20–30 fotiek objektu pod rôznymi uhlami a zo všetkých strán. Tie presunieme do bezplatne dostupného softvéru (napr. AliceVision). Program hľadá spoločné prvky na všetkých fotografiách a snaží sa pomocou nich vypočítať, z akého uhlu bol na danej fotografii predmet vyfotografovaný. Z týchto informácií potom dokáže softvér vytvoriť body v 3D priestore.

Ak Vás takéto zhmotňovanie nápadov zaujalo – a mali by ste záujem pomocou spomenutých digitálnych technológií zhmotniť aj svoj nápad – pozývame Vás do bezplatnej tvorivej dielne FabLab v Bratislave. Naučíte sa u nás pracovať s jednotlivými technológiami a vyrobený model si môžete vziať domov iba za cenu materiálu.

Východiská stratégie vzdelávania pre potreby IT odvetvia v regióne Košického samosprávneho kraja

Martin Šechný
Prešovská univerzita v Prešove
Prešov
Slovenská republika
martin.sechny@shenk.sk

Juraj Ťapák
občianske združenie Sapiente
Košice
Slovenská republika
jtapak@yahoo.com

Martin Džbor
T-Systems Slovakia
Košice
Slovenská republika
martin@dzborovci.com

Viliam Fedák
občianske združenie Sapiente
Košice
Slovenská republika
fedak@sapiente.sk

ABSTRAKT

Slovenské IT odvetvie s prevažne stabilným vývojom sa mení, kopíruje moderné trendy v EÚ a vo svete. Región Košického samosprávneho kraja má významný potenciál rastu odvetvia, založeného na inováciách a nových technológiách. Pod vplyvom nových technológií dochádza k postupnej digitálnej transformácii hospodárstva a celej spoločnosti. Úspešnosť zmeny v odvetví závisí najmä od dostupnosti kvalifikovanej pracovnej sily. Identifikovali sme hlavné problémy vo vzdelávacom systéme z pohľadu potrieb trhu práce. Medzi východiská tvorby verejnej stratégie vzdelávania a prípravy pracovnej sily možno zaradiť aj doterajšie národné a regionálne politiky, národné projekty a iniciatívy neziskového sektora.

ABSTRACT

The Slovak IT sector, with its predominantly stable development, is changing, copying modern trends in the EU and the world. The region of Košice has significant potential for growth of the sector, based on innovations and new technologies. Under the influence of new technologies, a gradual digital transformation of the economy and society as a whole is taking place. The success of the change in the sector depends mainly on the availability of a skilled workforce. We have identified the main problems in the education system from the perspective of labour market needs. Existing national and regional policies, national projects and initiatives of the non-profit sector can be included as starting points for the creation of a public education and training strategy.

Kľúčové slová

stratégia, vzdelávanie, IT, Košice, kvalifikovaná pracovná sila

Keywords

strategy, education, IT, Košice, qualified workforce

1 ÚVOD

Vo všetkých odvetviach slovenského hospodárstva prebieha digitálna transformácia. Predvojom zmien v celom hospodárstve sú zmeny v IT odvetví, lebo práve IT technológie umožňujú digitalizáciu. Slovenské IT odvetvie má prevažne stabilný vývoj, ktorý postupne kopíruje moderné trendy v EÚ a vo svete. Všetky tieto nové trendy budú mať dopad na dynamiku a štruktúru trhu práce. Región Košického samosprávneho kraja má významný potenciál rastu odvetvia, založenom na inováciách a nových technológiách, k čomu potrebuje kvalifikovanú pracovnú silu. [1]

2 TRH PRÁCE A VZDELÁVANIE

V súčasnom období ekonomickej expanzie slovenského hospodárstva problémy na trhu práce nevyplývajú z nedostatočnej tvorby pracovných príležitostí, ale z nedostatočného prepojenia systému vzdelávania s potrebami zamestnávateľov. SOŠ nevyučujú zamestnávateľmi požadované odbory, alebo ich vyučujú nevyhovujúco. Najväčšiu skupinu uchádzačov o zamestnanie tvoria stredoškolsky odborne vzdelaní ľudia, ktorí školu ukončili v priebehu posledných 10 rokov, no nemajú požadované spôsobilosti. Aktuálne je na trhu práce v IT odvetví alarmujúci nedostatok kvalifikovanej pracovnej sily a vo všeobecnosti pracovný trh čelí vážnej disparite medzi dopytom a ponukou. Podľa prognóz vývoja na trhu práce približne 63 % stredoškolských a 53 % vysokoškolských absolventov nepracuje v odbore, ktorý vyštudovali. IT odvetvie predstavuje takmer 7,5 % zamestnanej pracovnej sily v košickom regióne, čo je druhé najvýznamnejšie centrum po Bratislave. IT odvetvie v Košickom kraji vykazuje rast z hľadiska tvorby pracovných miest už od roku 2013. [2]

Požiadavky zamestnávateľov pokrývajú celú škálu zručností, od programátorských, cez analytické až po manažérske. Trend zároveň smeruje k integrovaniu hardvérových a softvérových pozícií, vyžaduje od vývojárskych a prevádzkových tímov schopnosť spolupracovať s cieľom naplnenia potrieb zákazníka (DevOps). Dochádza k posunu od striktnej špecializácie k väčšej generalizácii.

Štúdium by malo pripraviť absolventa na širokú paletu povolání v IT odvetví. Stredné školy potrebujú dramaticky zvýšiť svoju orientáciu na prax. Za hlavné problémy a výzvy vo vzdelávaní považujeme: podfinancované školy, nedostatok kvalifikovaných učiteľov STEM predmetov, neefektívny systém vzdelávania učiteľov, zaostávajúci a nereformovaný vzdelávací systém všeobecne, učitelia a výchovní poradcovia nemajú dostatočné informácie o IT odvetví, nedostatočne zadané národné štandardy kvalifikácií, nutnosť zmeny prístupu vo vzdelávaní prichádzajúcich generácií, nedostatok mäkkých zručností absolventov, potreba celoživotného vzdelávania, nedostatočná komunikácia medzi zamestnávateľmi a školami a v smere od vyšších stupňov vzdelávania k nižším, legislatívne prekážky pre duálne vzdelávanie SOŠ v IKT odbore.

Školský vzdelávací program na základných školách by mal zohľadniť reálne potreby detí ako budúcich používateľov IKT (digitálne zručnosti, vyhľadávanie informácií, prácu s dátami a dokumentami, algoritmické riešenie a bezpečnosť. [3] Predmet Informatika sa priradzuje k matematike a ostatným prírodným vedám, aj napriek tomu, že digitálna gramotnosť má viac technickú povahu – táto systematická chyba je najviditeľnejšia na gymnáziu. Informatika by mala rovnomerne pokrývať 3 hlavné smery: počítačové inžinierstvo, počítačovú vedu, dátovú vedu.

3 ZÁVER

V článku sme popísali východiská, prečo Košický kraj potrebuje nevyhnutne komplexnú a koherentnú stratégiu pre IKT vzdelávanie. Bez inštitucionalizácie verejno-súkromných partnerstiev a ich viac-zdrojového financovania a bez synergie zainteresovaných hráčov sa potenciál, ktorý v sebe má IT odvetvie v Košickom kraji, nenaplní, pretože nebude mať kvalitne pripravené ľudské zdroje.

Tento článok bol pripravený v rámci riešenia projektu „Tvorba verejnej stratégie vzdelávania a prípravy pracovnej sily pre potreby IT sektora v regióne Košického kraja“ (akronym: LearnIT.ke, operačný program: Efektívna verejná správa, kód projektu: 314011L929)“ implementovaného Občianskym združením Sapiente. Tento projekt sa realizuje vďaka podpore z ESF.

4 BIBLIOGRAFICKÉ ODKAZY

- [1] NOVAK, J. et al. *The Rise of Digital Challenges*. New York, USA: Digital/McKinsey, 2018.
- [2] HROBÁK, P. *Analýza požiadaviek zamestnávateľov z IT sektora od trhu práce v regióne Košického kraja*. Košice: občianke združenie Sapiente, 2019. <https://www.typlusit.sk/wp-content/uploads/2020/02/2-ANALYZA-ZAMESTNAVATELOV-plna-verzia.pdf>.
- [3] VELŠIC, M. *Digitálna gramotnosť na Slovenku 2018. Inštitút pre verejné otázky*. IBSN 978-80-89345-69-4. http://www.ivo.sk/buxus/docs//publikacie/subory/Digitalna_gramotnost_2018.pdf.

Diskusní příspěvek¹ k porovnání požadavků na informatiku a učitele informatiky

Libor Dostálek

Přírodovědecká fakulta Jihočeské univerzity v Českých Budějovicích, Branišovská 1760, 370 05 České Budějovice, Česká Republika, dostalek@jcu.cz

Na obrázku je zobrazen rozsah výuky v bakalářském a navazujícím magisterském stupni v informatice a v učitelství informatiky pro střední školy na některých univerzitách. Údaje je třeba brát jen orientačně, protože jednotlivé vysoké školy mají různou metodiku pro stanovení kreditů zejména za pedagogickou praxi (např. MFF UK). Jiné zase předpokládají, že se jedná o dvouoborové studium a kurzy společné pro oba obory jsou všechny vykázány v obou oborech, tj. ve své podstatě dvakrát (např. Jihočeská univerzita).

Z obrázku je ale patrné, že středoškolský učitel absolvuje v rámci bakalářského a studia přibližně 1/3 výuky informatiky oproti studentům studujícím informatiku jako takovou. Přičemž navíc ještě některé odborné kurzy mohou mít uživatelský charakter, tj. v případě takových kurzů se spíše jedná o zaškolení.

Ještě markantnější rozdíl je v magisterském studiu, Proti tomu středoškolský učitel získá na některých školách jen 10 kreditů z odborného předmětu (nepočítaje v to didaktiku a uživatelsky orientované kurzy).

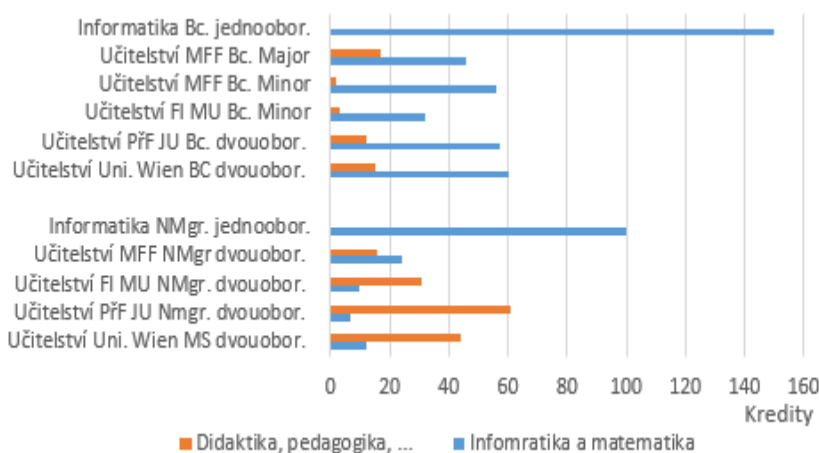
Skutečnost, že absolvent učitelství pro střední školy absolvuje přibližně třetinu kurzů z odborného předmětu, je vcelku přirozená a odpovídá modelu dvouoborového studia, kdy studium je vcelku rozděleno na třetiny: dva obory a didakticko-pedagogický blok. S příchodem major/minor modelu se pak rozložení kurzů může vychýlit ve prospěch jednoho oboru. Otázkou je, jestli výsledné znalosti jsou dostatečné. Je výuka v takovém rozsahu vůbec schopna přiblížit posluchače k současnému stavu poznání v oboru, který bude vyučovat? Nemohu odpovědět za všechny obory, ale v informatice nebo matematice dle mého názoru rozhodně nikoliv.

Absolvent oboru středoškolský učitel má být připraven pro výuku na nejrozličnějších typech středních škol: gymnázia, odborné střední školy s maturitou, střední odborné školy s výučním listem či střední odborné školy bez výučního listu (s vysvědčením). Předpokládá se, že absolvent učitelství pro střední školy je univerzálním učitelem pro všechny typy uvedených středních škol. Já se domnívám, že takový univerzální učitel nemůže existovat. Pro každý typ školy potřebuje učitel odlišné znalosti nejen vyučovaného oboru, ale zejména z oblasti didaktiky a psychologie.

Dle mého názoru není správné dělit učitele podle středních škol, na kterých budou učit, ale podle studentů, které budou učit.

Středoškolští učitelé budou učit dva typy studentů:

- Studenty, pro které je daný obor (v našem případě informatika) profilovým oborem. Tj. obor, kterému se budou dále věnovat studiem na vysoké škole.
- Studenty, pro které je daný obor součástí jeho všeobecného středoškolského vzdělání a profesně se mu již dále věnovat nebudou.



Pro studenty první skupiny by měl být středoškolský profesor vzorem, pro který se budou danému oboru věnovat. Avšak aby učitel mohl být vzorem, pak by měl mít alespoň špetku znalostí současného stavu poznání daného oboru. Opět vystává otázka, jestli během vysokotlakého studia učitelství pro střední školy má šanci takové znalosti vůbec získat.

Domnívám, že současní absolventi jsou vychováni pro studenty druhého typu. Tím pádem středoškolští studenti se nesetkávají se vzory, které by je vedly ke studiu nejenom informatiky, ale i matematiky, fyziky, chemie atd.

První polovina dvacátého století byla bojem o to, aby se studium učitelství přeneslo z odborných fakult na pedagogické fakulty, aby absolventi získali hlubší znalosti z didakticko-pedagogického bloku. Dnes jsme u opačného extrému. Dnešní budoucí středoškolští učitelé upřednostňují tento blok nad vlastní obor. Na vysokých školách s nimi velice tvrdě bojujeme s jejich argumentací, že nepotřebují o moc víc odborných znalostí, než sami budou vyučovat na střední škole. V konečném důsledku očekávají, že budou v oboru zaškoleni nikoliv, že ho vystudují.

Realita je dokonce taková, že pokud mají společný kurz se studenty „ne-učitelství“, tak často dosahují horších výsledků než studenti „ne-učitelství“, kteří mají podstatně horší výchozí předpoklady pro studium (např. absolvovali střední školu s menším rozsahem výuky matematiky). V podstatě se brání hlubšímu vzdělání v oboru, protože nejsou dostatečně motivováni skutečností, že by tyto znalosti využili. Je to důsledkem toho, že není společenská poptávka po tom, aby středoškolští učitelé měli tyto znalosti.

V informatice máme oproti jiným oborům ještě jeden velký problém. Tím je výchova oboru učitelství informatiky v kombinaci se správcem sítě. To u středoškolských studentů vyvolává falešný dojem, že vzorem informatika je správce sítě. Avšak v IT jsou tester a správce sítě (pokud nespravuje síť o stovkách nebo tisících počítačů) považováni za nejméně kvalifikovanou pozici.

Před tím, než budeme řešit, jaké požadavky budeme klást na uchazeče o práci středoškolského učitele, je třeba si položit otázku, jaké studenty bude tento učitel učit. Přesněji řečeno. Budou mezi jeho studenty i studenti prvního druhu? Pokud nikoliv, pak takovému uchazeči dostačují znalosti na úrovni současného vysokoškolského studia středoškolského učitelství.

Ovšem pokud se mezi jeho studenty budou vyskytovat i studenti prvního typu, pak bychom od takového uchazeče měli vyžadovat i něco navíc. Otázkou je, co to „něco navíc“ je.

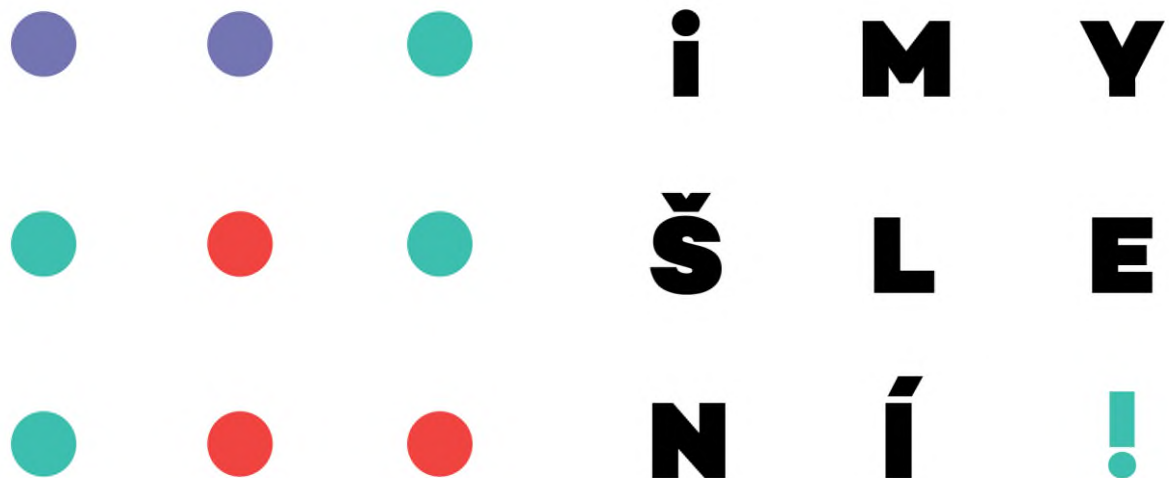
Odpověď je jednoduchá: musí být v daném oboru nějak aktivní. Platí to i pro základní umělecké školy. Pokud hledáme učitele pro nadaného žáka hudby, pak rovněž nehledáme učitele, který pouze učí na ZUŠ, ale takového, který navíc např. hraje v nějakém orchestru.

U vysokoškolských učitelů máme požadavky jasně dány: musí pravidelně publikovat v impaktovaných časopisech. To samozřejmě po středoškolském učiteli požadovat nemůžeme. Můžeme ale po něm vyžadovat, aby články v takových časopisech četl nebo alespoň byl schopen číst. Takovými časopisy se, pochopitelně, nerozumí české populární časopisy.

Dalším problémem na středních školách jsou olympiády. Je to typický příklad skvělé věci, na kterou je nešťastně kladen extrémní důraz. Olympiády jsou perfektní příležitost pro rozvoj některých nadaných studentů, ale nikoliv všech, kteří mohou být úspěšní v daném oboru.

Závěrem bych chtěl vyslovit domněnku, že s tím, jak před cca 70 lety se při vzdělávání budoucích středoškolských učitelů (tehdy středoškolských profesorů) přestal klást důraz na vzdělání v oboru a začal klást důraz na jiné kvality, tak se postupně začala snižovat úroveň vzdělání maturantů. Znalost matematiky ze střední školy mých rodičů jsem dosáhl až na MFF UK. Znalosti dnešních maturantů jsem si odnesl již ze základní školy.

ⁱ Příspěvek byl publikován jako autorovo zamyšlení, polemika. Vychází z autorova názoru, z tohoto důvodu pod ním nenaleznete zdroje a je uveden až po sekci odborných článků a rozšířených abstraktů.



V roce 2020 byla konference Didinfo podpořena z projektu Podpora rozvíjení infromatického myšlení.
reg. č. CZ.02.3.68/0.0/0.0/16_036/0005322

Vydavatel: Technická univerzita v Liberci
Schváleno rektorátem dne 28. 7. 2020, č. j. RE 22/20, č. pub. 55-022-20.

ISBN 978-80-7494-532-8
ISSN 2454-051X
