



Vysoká škola
polytechnická
Jihlava



Informatika 2023

Sborník příspěvků z konference

Vysoká škola polytechnická Jihlava

4.–5. září 2023

Jihlava, Česká republika

Konference Informatika 2023 se konala pod záštitou EUNIS.



EUNIS.CZ

Informatika 2023

Sborník příspěvků z konference

Vydavatel: Vysoká škola polytechnická Jihlava
Tolstého 1556/16, 586 01 Jihlava

Editor: Hana Vojáčková

Publikace neprošla jazykovou kontrolou.

Příspěvky jsou řazeny abecedně podle příjmení autora.

Všechny příspěvky prošly recenzním řízením.

Technické zpracování a výroba:

Vysoká škola polytechnická Jihlava
Tolstého 1556/16, 586 01 Jihlava

První vydání

2023

© Vysoká škola polytechnická Jihlava

© Autoři příspěvků

ISBN 978-80-88064-69-5 (online ; pdf)

Vědecký výbor

prof. Ing. Tomáš Dostál, DrSc., Vysoká škola polytechnická Jihlava

doc. Ing. Ladislav Beránek CSc., MBA, Jihočeská univerzita v Českých Budějovicích

doc. Ing. Zbyněk Bureš, Ph.D., České vysoké učení technické v Praze

doc. Ing. Karel Richta, CSc., Vysoká škola polytechnická Jihlava

Ing. Radek Holý, Ph. D, ČVUT, EUNIS-CZ

PaedDr. František Smrčka. Ph.D., Vysoká škola polytechnická Jihlava

Mgr. Hana Vojáčková, Ph.D., Vysoká škola polytechnická Jihlava

Ing. Lenka Kuklišová Pavelková, Ph.D., Vysoká škola polytechnická Jihlava

Ing. Bc. Karel Dvořák, Ph.D., Vysoká škola polytechnická Jihlava

Organizační a programový výbor

PaedDr. František Smrčka. Ph.D.

Ing. Lenka Kuklišová Pavelková, Ph.D.,

Mgr. Hana Vojáčková, Ph.D.

Ing. Bc. Karel Dvořák, Ph.D.

Mgr. Zdeňka Dostálová

Ing. Marek Musil

Mgr. Antonín Příbyl

Michaela Machovcová

Webové stránky konference: <https://informatika2023.vspj.cz/>

Obsah

NESTANDARDNÍ VIZUALIZACE V PROSTŘEDÍ MS EXCEL	7
Josef Botlík, Milena Botlíková	7
VYUŽITÍ DAT DATABÁZE EUROSTAT A URBAN ATLAS VE VÝUCE PŘEDMĚTU DATA MINING.....	14
Zdena Dobešová	14
CAX/PLM SYSTÉMY NÁSTROJEM KOMUNIKACE.....	19
Karel Dvořák	19
KORPUS STUDENTSKÝCH ZDROJOVÝCH KÓDŮ – SYNTAKTICKÁ EKVIVALENCE	25
Jiří Fišer, Pavel Beránek, Jiří Škvor.....	25
ÚVOD DO IT NA TECHNICKÉ FAKULTĚ	31
Miroslav Hrubý.....	31
ELEARNINGOVÉ ŘEŠENÍ NA TUL ZALOŽENÉ NA MOODLE	37
Igor Kopetschke.....	37
OPERAČNÍ SYSTÉMY – STARÉ TEORIE, NEBO NOVÉ VÝZVY?.....	43
Lenka Kosková Trísková.....	43
DIDAKTICKÉ METODY A EXPERIMENTÁLNÍ PŘÍSTUPY VE VÝUCE PROGRAMOVÁNÍ...49	
Vojtěch Lapuník	49
ZÁKULISÍ VÝUKY PŘEDMĚTU DIGITÁLNÍ TRANSFORMACE PODNIKU – PODNIKOVÁ ARCHITEKTURA (MEZI TEORIÍ A PRAXÍ).....	55
Martin Lukáš	55
VÝUKA PŘEDMĚTU STROJOVÉ UČENÍ.....	59
Jan Mittner.....	59
APLIKACE PRO AUTOMATIZOVANÉ HODNOCENÍ ÚKOLŮ.....	65
Marek Musil	65
PROBLEMATIKA E-LEARNINGU A DIGITÁLNÍCH ZDROJŮ VE VÝUCE ČEŠTINY PRO CIZINCE.....	70
Dean Posavec-Malok	70
ZKUŠENOSTI SE SOFTWAREM DVWA VE VÝUCE PŘEDMĚTU BEZPEČNOST INFORMAČNÍCH SYSTÉMŮ.....	79
Radim Remeš, Ladislav Beránek, Jan Fiala	79
APLIKACE PRO ZRYCHLENÉ VYTVOŘENÍ TESTŮ PRO LMS MOODLE	85
František Smrčka	85
VÝUKA PŘEDMĚTU BUSINESS INTELLIGENCE NA SU OPF V KARVINĚ	90
Petr Suchánek	90

AUTOMATICKÁ KONTROLA ÚLOH Z PROGRAMOVÁNÍ.....	96
Martin Vitek	96
INOVATIVNÍ AKTIVITY VE VÝUCE IT NA TECHNICKÉ UNIVERZITĚ V LIBERCI	101
Jana Vitvarová	101
ROZŠÍŘENÉ POJETÍ VÝUKY DEVOPS NA FIT ČVUT.....	107
Tomáš Vondra	107

Vážené čtenářky, vážení čtenáři,

po roční odmlce se vám do rukou dostává další sborník konference Informatika, nyní tedy Informatika 2023, pořádané Vysokou školou polytechnickou Jihlava. Obsah tohoto sborníku je zaměřen na sdílení pedagogických zkušeností akademických pracovníků především v ICT předmětech. Autoři také prezentují zaměření a výsledky své výzkumné práce, případně svoji spolupráci se soukromou sférou.

Jsem rád, že tento sborník obsahuje zase o něco více příspěvků než ten minulý. Považuji to za signál, že náš záměr obnovit informaticko-pedagogickou platformu pro vzájemné setkávání, sdílení dobré praxe a budování kontaktů je na dobré cestě. Též na obsahu příspěvků je vidět, že si konference našla svoje témata. Letos se navíc objevují některá nová zajímavá témata, jako je strojové učení nebo Data Mining. Je vidět, že konference Informatika se stává příležitostí nejen pro zmíněné setkávání mezi akademiky, zaměřenými především na informatické předměty, ale i pro navazování spolupráce mezi kolegy, jednotlivými školami či výzkumnými kolektivy.

Jsem přesvědčen, že i tento sborník konference Informatika 2023 přinese poučení a inspiraci pro vaši výzkumnou a pedagogickou práci.

Za organizační a programový výbor

PaedDr. František Smrčka, Ph.D.

NESTANDARDNÍ VIZUALIZACE V PROSTŘEDÍ MS EXCEL

NON-STANDARD VISUALIZATION IN THE MS EXCEL ENVIRONMENT

Josef Botlík, Milena Botlíková

Abstrakt

Inovativnost a lukrativitu výukového procesu lze podpořit nestandardním využíváním nástrojů, zdánlivě nesouvisejících s vyučovaným tématem. Produkty MS Office mají značný skrytý edukační potenciál vycházející z nástrojů a vlastností jednotlivých objektů. Příspěvek se zabývá možnostmi vizualizace na základě využití modifikace vlastností buněk, které mohou doplnit a podpořit výuku, například při zpracování obrazu. Efektivita a přidaná hodnota je dána vytvořením procesu, ve kterém studenti nejen sledují změny obrazu související se změnou atributů a obrazových funkcí, ale mohou sledovat podstatu tohoto procesu na konkrétních parametrech obrazových bodů.

Klíčová slova: MS Excel, výuka, inovace, barvy, obraz

Abstract

The innovativeness and lucrativeness of the teaching process can be supported by the non-standard use of tools seemingly unrelated to the taught topic. MS Office products have considerable hidden educational potential based on the tools and properties of individual objects. The paper deals with the possibilities of visualization based on the use of a modification of cell properties, which can supplement and support teaching, for example during image processing. Efficiency and added value is given by the creation of a process in which students not only observe image changes related to the change of image attributes and functions but can observe the essence of this process on specific pixel parameters.

Keywords: MS Excel, education, innovation, colors, picture

Úvod

V dnešní době je trendem do výukového procesu implementovat inovativní metody, hovoříme o inovativním vzdělávání, resp. o inovacích ve vzdělávání. Zejména je patrný nárůst efektivity vzdělávacího procesu podmíněný inovacemi v ICT, nedávná pandemická doba nastartovala vlnu inovativních procesů, počínaje online formami výuky, zvýšeným zájmem o ICT a konče například novými formami týmové práce. Prostředky ICT tedy působily jako multiplikátory, lze konstatovat, že v některých případech lze vysledovat prvky pozitivní externality (zvyšování kompetencí v různých oblastech IT podmíněné inovacemi v komunikačním procesu). Velmi často je však v tomto procesu opomíjený fakt, že "svět se vyvíjí ve spirále", většina "pseudo" inovací již v jistých podobách existovala a "nově" objevené metody a postupy lze aplikovat stejně dobře jak "novými" prostředky, tak i "starými" nástroji, zapomínáme na zkušenosti z minula a "s velkou slávou" tyto zkušenosti později opět objevujeme a "inovativně" implementujeme. V procesu výuky velmi často zapomínáme na skutečnost, že vyučovat bychom měli "podstatu", tj "příčinu" a nikoli "důsledky". Tento fakt je stále patrný například ve fyzice či v matematice, kdy výuka příslušných vzorců sice studentům vysvětlí "co" se změní a "jak", málokdy ale následuje vysvětlení "proč". Informační technologie se potýkají

s obdobným problémem, mnoho procesů je založeno na matematických výpočtech, které jsou reprodukovány pomocí programů a aplikací. Existují sofistikované nástroje, které umožňují provádět náročné výpočty, výuka však může přecházet do formy "práce s nástroji" v příslušné aplikaci bez možnosti sledovat konkrétní "procesy" vytvářené a prováděné těmito nástroji. Tento příspěvek si klade za cíl seznámit s možnostmi standardně používaného prostředí používaného při výuce i v praxi pro výuky základních poznatků vztažených k elementárním poznatkům týkajících se práce s obrazem. Není zde uváděná sofistikovaná práce s grafikou, v rámci rozsahu se budeme věnovat pouze ukázce práce s bitovou grafikou.

1 Práce s grafikou a její výuka

Existuje nepřehledné množství kurzů a návodů sloužících k výuce grafiky a mnoho více či méně profesionálních aplikací. V podstatě práci s grafikou můžeme rozdělit do několika obsahových skupin (podle různých kritérií), v zásadě se ale bude jednat v jedné rovině o praktickou část práce s obrazem prostřednictvím existujícího software (namátkou Adobe Illustrator, Adobe Photoshop, CorelDRAW, Corel Paint, Clip Studio Paint, Artrage, Krita, Sketch, Affinity Designer, Inkscape, či GIMP). V jiné rovně se pak jedná o sofistikované, více méně matematické nástroje, umožňující programování a implementaci matematických modelů pro reprezentaci dat, úpravu obrazu, práci s barevnými modely, převody mezi barevnými prostory apod.

V prvním případě se jedná o rutinní činnosti, které jsou sice vysoce odborné a vztažené ke konkrétnímu software, nicméně, uživatel však není seznámen s podstatou procesu. Jako příklad lze uvést návod pro převod barevných režimů v aplikaci Photoshop (helpx.adobe.com, 2023), kde je postup: "Zvolte Obraz > Režim a vyberte z podnabídky požadovaný režim. Režimy, které nejsou pro aktivní obraz použitelné, jsou v nabídce ztlumené", popřípadě, pro "převod obrazu do režimu bitová mapa" je uváděno: "Chcete-li převést barevný obraz do režimu bitová mapa, převedte ho nejdříve do režimu stupně šedi. Tím se odstraní z obrazových bodů informace o odstínu a sytosti a ponechají se pouze hodnoty jasu. Protože pro obrazy v režimu bitová mapa je k dispozici jen málo příkazů pro úpravy, je obvykle nejlepší upravovat obraz v režimu stupňů šedi a pak ho převést do režimu bitová mapa". Je patrné, že uživatel daný efekt dosáhne, neví však "co se děje v pozadí". Naopak, více sofistikované postupy volí matematické metody, resp. algoritmizaci a výpočty např. v prostředí Pythonu, resp. jiného programovacího jazyka s podporou knihoven pro práci s grafikou, popř. prostředí jako MATLAB. V tomto případě se dostáváme do situace, že zejména na netechnických studiích (a s přihlédnutím k současnému preferovanému "odbourávání" matematiky z výuky), budou pro "nematematiky" (pro studenty bez matematického chápání a studenty humanitních oborů) probíhající procesy nesrozumitelné. Například implementace vzorce pro konvoluci (Ličev, 2023) bude pro většinu "nematematiků" nepochopitelná (1).

$$f(x, y) * h(x, y) = \sum_{i=-k}^k \sum_{j=-k}^k f(x - i, y - j) \cdot h(i, j) \quad (1)$$

Na hranici chápání bude pro některé např. vzorec pro převod mezi barevnými modely (2) z HSL do RGB (převzato z www.rapidtables.com, 2023, uvedena část vzorce). Takových příkladů je nespočet.

$$0 \leq H < 360, 0 \leq S \leq 1 \text{ and } 0 \leq L \leq 1: \begin{cases} C = (1 - |2L - 1|) \times S \\ X = C \times (1 - |(H/60^\circ) \bmod 2 - 1|) \\ m = L - C/2 \end{cases} \quad (2)$$

Faktem zůstává, že je enormní (pro mnohé nepřekonatelná) mezera mezi chápáním funkčnosti v daném programu, jeho matematickou podstatou a vlastní implementací matematického modelu.

Tímto jsme chtěli ukázat, že zpracování obrazu je sice v současnosti velmi rozšířené, ale výuka "podstaty" může být problematická.

2 Alternativní metody výuky a vizualizace pomocí MS Excel

Na skutečnost, že *"využití technologií ke zlepšení vzdělávání ... je velmi pozitivní"* upozorňují např. Banegas, et al. (2021), kteří zmiňují *"nový přístup, vyvíjející různé virtuální nástroje v Excelu, ... umožňuje snadné použití jak pro učitele, tak pro studenty"*. Z obecného pohledu lze zmínit i studii (Tarkhova et al. 2020) jejímž účelem je *"zvýšit efektivitu vzdělávacího procesu pomocí infografického obsahu, který umožňuje vytvářet interaktivní infografiky..."* a pro práci s obsahem používají Excel, ten jako nástroj pro vizualizaci zmiňují i další autoři, například Derzon et al. (2013) popisuje grafické možnosti. Ve většině případů se jedná v obdobných studiích o využívání standardních nástrojů MS Excel, jako grafy a grafické objekty. V některých případech se jedná o využití kombinací nástrojů, například Excel je využíván jako prostor pro uchování dat, která jsou importována do bitové mapy (Rougier, 2018). Univerzálnost nástrojů MS Office uvádí mnoho autorů, již v roce 2004 lze nalézt mnoho inspirací (Higgins, 2004) ve výuce, například využívání interaktivních spreadsheetů vytvářených v Excelu. Konkrétní využití pro zpracování kanálů HSV a RGB prostřednictvím statistických funkcí MS Excel uvádějí Sharma et al. (2021), vlastní zpracování obrazu je však prováděno pomocí Pythonu. V souvislosti s algoritmizací pak Safitri et al. (2020) z jiného úhlu zkoumali *"získání... efektivních kritérií pro efektivitu učebních médií pomocí VBA"*, čímž poukazují na možnost propojení algoritmizace a uložení dat v MS Excel při výuce. Použití Excelu a VBA při vizualizaci ve vzdělávání uvádějí Villalobos et al (2011) jako práci s nástroji *"pro lidi nebo společnosti, které mají omezený přístup k sofistikovaným nástrojům informačních technologií"*.

Při výuce grafiky v Excelu je důležité klást důraz na to, že Excel je nástroj, který může být použitelný i v profesionálním prostředí pro základní vizualizace, ale pro složitější grafiku by bylo vhodné studenty seznámit s nástroji specificky určenými pro grafický design. Obecně lze MS Excel ve výuce grafiky pro vizualizaci využít na grafy, včetně kombinace s histogramy, kontingenčními grafy či dynamickými grafy propojenými s ovládacími prvky. Při propojení s formuláři a funkcemi lze dále efektivně využívat barvy a efekty barev.

Pokud bychom chtěli pracovat s grafikou v Excelu tak, že vytvoříme obrázek z buněk (support.microsoft.com, 2021), je nutné si uvědomit, že vytvoření obrázku použitím grafu nebo objektu v Excelu, resp. vložení obrázku přes schránku není skutečné vytvoření pracovního prostředí pro práci s obrázkem. Chceme-li skutečně fyzicky pracovat s obrázkem, je vhodné pracovat s bitovou mapou a jako jednotlivé pixely použít buňky Excelu. Excel umožňuje práci s RGB a HSL barevným modelem, umožňuje tak seznámit studenty s chováním barev při míchání barevných složek, konverzí barevného obrazu na negativ, na stupně šedi apod., popřípadě pracovat s jasnou, sytostí či světlostí obrazu. I když Excel neumožňuje přímý export parametrů těchto barevných modelů jako hodnoty do buněk, pomocí jednoduchých algoritmů či s pomocí externích nástrojů lze tyto parametry přenést jako číselné hodnoty buněk. Uspořádáme-li tyto hodnoty do příslušné matice lze je využívat pro vytváření a modifikaci obrázků.

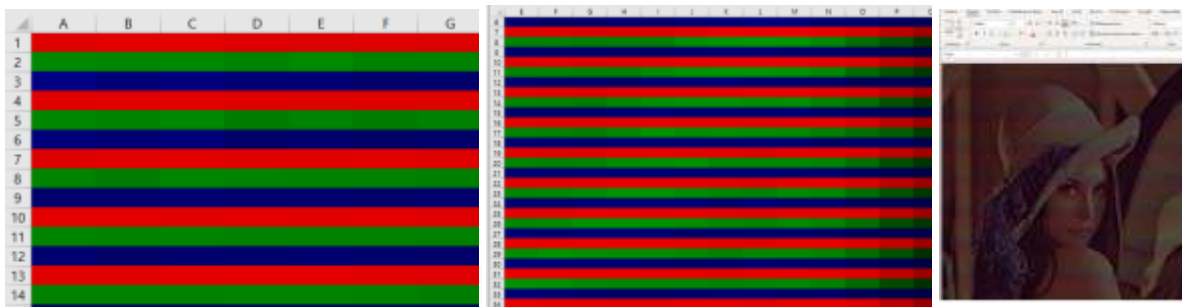
3 Praktické příklady nestandardních metod využití grafických nástrojů MS Excel

Pro základní demonstraci byl zvolený standardní testovací obrázek Lenna (viz obrázky 2 a 4) v rozlišení 128x128 pixelů. V první fázi byly barevné složky obrazu převedeny do buněk jako numerické hodnoty, vlastní zdrojový kód VBA má 5 deklarovaných funkcí a přibližně 50 řádků bez optimalizace kódu (obrázek 1).



Obrázek 1: Část kódu VBA. Zdroj: vlastní zpracování

Uvedené načtené hodnoty byly následně využity dvěma základními způsoby demonstrujícími skládání barevných složek. V prvním případě byly vytvořeny trojice buněk pod sebou, nesoucí R, G a B hodnotu a na základě této hodnoty byla upravena barva výplně buněk (obrázek 2). Takto lze demonstrovat jednoduše reálné vytváření obrazu pomocí barevných složek.



Obrázek 2: Skládání barevných složek. Zdroj: vlastní zpracování

Celý proces je velmi jednoduchý a spočívá v podstatě z několika řádků VBA: přečtení hodnoty v buňce a nastavení příslušné barvy výplně a barvy písma na tuto barvu (samostatně pro buňky s R, G a B složkou). Pro modrou barvu tedy:

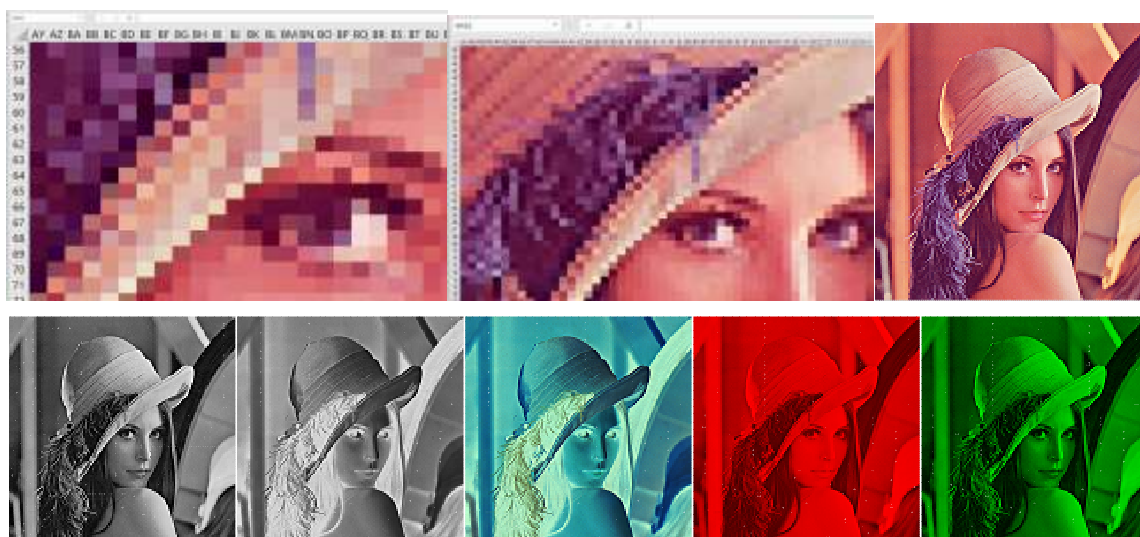
```
barva = ActiveSheet.Cells(i + k, j).Value
ActiveSheet.Cells(i + k, j).Interior.color = rgb(0, 0, barva)
ActiveSheet.Cells(i + k, j).Font.color = rgb(0, 0, barva)
```

Uvedené barevné složky lze následně libovolně upravovat, prahovat, invertovat apod. Ve druhém případě byly barevné složky rozloženy na 3 samostatné listy, což umožnilo jednodušší zpracování jednotlivých složek, dále byl vytvořen podle standardního postupu list s odstíny šedi. Možností převodů mezi barevnými modely je více, podle požadovaných příkladů pro demonstraci, studenti pracovali například s YCbCr modelem, snadno z RGB složek vytvořili listy s Cb a Cr složkou a mohli tak vidět reálné rozdíly jak v hodnotách, tak v interpretaci modelu. Studenti se tak prakticky seznamují s modely, nevidí pouze vzorce nebo obrázek, vidí reálný proces proměny obrázku při práci, resp. transformaci. Studenti si vytvářejí s pomocí pedagoga prostředí podle potřeby a podle případných požadavků na transformaci, oproti jiným nástrojům však vidí implementaci příslušných algoritmů v listech s daty a v kódu VBA. Vzhledem k všeobecným znalostem prostředí MS Excel studenti tuto práci zvládají i u netechnických oborů. S využitím standardních nástrojů (grafy, statistické funkce) tak studenti mohou snadno vizualizovat hodnoty například pomocí histogramu (obrázek 3).



Obrázek 3 Rozhraní a možnosti práce. Zdroj: vlastní zpracování

Ve výsledku tak studenti mohou nejen vytvářet různé modifikace obrázku, ale i srovnávat výstupy, počínaje vlastními hodnotami dat, statistickými parametry i finální podobou obrázku (Obrázek 4).



Obrázek 4 Vizualizace pomocí formátu buněk v Excelu. Zdroj: vlastní zpracování

Závěr

Je na místě zdůraznit, že jsme si vědomi omezení, které nabízí Excel a VBA. Vycházíme z poznatků, že jednoduché praktické ukázky principů lze však v tomto prostředí velmi zajímavým způsobem demonstrovat. Nejsilnějším argumentem je fakt, že s prostředím MS Office se dnes standardně žáci a studenti setkávají ve školách a znalosti práce v tomto prostředí jsou považovány za základní vědomosti. Rovněž ve firemním prostředí je tento softwarový balík hojně využíván, takže je vysoká pravděpodobnost obecných znalostí tohoto prostředí. Kombinace práce s formáty objektů a možností algoritmizace vytváří předpoklady pro praktické aplikace matematických modelů do reálného prostředí. Velmi zajímavá je rovněž možnost využít schopností práce s 3D datovými oblastmi a vytvářet jednoduché animační sekvence, popřípadě "průchozí 3D objekty. Tato myšlenka není nová, už v roce 2008 na tyto možnosti upozornili redaktori aktuálně.cz (Pergl, 2008) s odvoláním na Gamasutra.com. Je tedy více než patrné, že je jen otázkou zkušeností pedagogů, jak skloubí inovativní učení s nestandardním využíváním obecně známých znalostí.

Poděkování

Tento článek byl podpořen Ministerstvem školství, mládeže a tělovýchovy ČR v rámci Institucionální podpory dlouhodobého rozvoje výzkumné organizace v roce 2023.

Literatura

- Převádění mezi barevnými režimy (2023). [online], [2023-08-01]. *Obrazová konvoluce*. Dostupné z: <https://helpx.adobe.com/cz/photoshop/using/converting-color-modes.html>
- Banegas, J. M., Cámara, E., Orué, M. (2021). Developing virtual instruments in excel for metrology teaching and training. *Journal of Physics: Conference Series*, 1826(1)
- Derzon, J. H., Alford, A. A. (2013). Forest plots in excel: Moving beyond a clump of trees to a forest of visual information. *Practical Assessment, Research & Evaluation*, 18, 7.
- Higgins, S., Packard, N. (2004). *Meeting the Standards in Primary ICT : A Guide to the ITTNC*, Taylor & Francis Group
- HSL to RGB color conversion* (2023). [online], [2023-08-01]. Dostupné z: <https://www.rapidtables.com/convert/color/hsl-to-rgb.html>
- Ličev, L. (2023). [online], [2023-08-01]. Dostupné z: (https://www.cs.vsb.cz/licev/lzs%20I_cviceni/Cv_5_konvoluce.pdf)
- Pergl, R. (2008). *Trojrozměrná grafika v Excelu*, [online], [2023-08-01]. Dostupné z: <https://zpravy.aktualne.cz/ekonomika/technika/trojrozmerna-grafika-v-excelu/r~i:article:523489/>
- Rougier, N. P. (2018). *A density-driven method for the placement of biological cells over two-dimensional manifolds*. *Frontiers in Neuroinformatics*
- Safitri, A. D., Lesmono, A. D., Maryani, Wardoyo, A. A. (2020). Development of learning media using VBA excel in physical learning in senior high school. *Journal of Physics: Conference Series*, 1563(1)
- Sharma, R., Kumar, M., Alam, M. S. (2021). Image processing techniques to estimate weight and morphological parameters for selected wheat refractions. *Scientific Reports (Nature Publisher Group)*, 11(1)
- Tarkhova, L., Tarkhov, S., Nafikov, M., Akhmetyanov, I., Gusev, D., Akhmarov, R. (2020). Infographics and their application in the educational process. *International Journal of Emerging Technologies in Learning*, 15(13), 63-80.
- Villalobos, J. P. C., Sanabria, S. P. J., Cáceres, R. G. G. (2011). Activity scheduling through gantt charts in an ms excel spreadsheet/Diagramas gantt para programación de actividades en el tiempo en diferentes recursos usando ms excel. *Revista Facultad De Ingeniería Universidad De Antioquia*, (61), 132-145.
- Vytvoření obrázku z buněk, grafu nebo objektu v Excel*. (2021). [online], [2023-08-01]. Dostupné z: <https://support.microsoft.com/cs-cz/office/vytvo%C5%99en%C3%AD-objekt%C3%A1-z-bun%C4%9Bk-grafu-nebo-objektu-v-excel-5545100b-65f7-4caf-ac12-7a56f4a4e7b6>

Kontaktní údaje

Ing. Josef Botlík

Katedra Informatiky a matematiky, Slezská univerzita v Opavě, OPF v Karviné

Univerzitní náměstí 1934/3, 733 40 Karviná

e-mail: botlik@opf.slu.cz

Ing. Milena Botlíková, Ph.D.

Slezská univerzita v Opavě, FPF v Opavě

Masarykova třída 343/37, 746 01 Opava

e-mail: milena.botlikova@fpf.slu.cz

VYUŽITÍ DAT DATABÁZE EUROSTAT A URBAN ATLAS VE VÝUCE PŘEDMĚTU DATA MINING

USE OF EUROSTAT AND URBAN ATLAS DATA IN DATA MINING COURSE

Zdena Dobešová

Abstrakt

Magisterské studium Geoinformatika a kartografie má zařazen předmět Data Mining. V rámci praktického procvičování témat byla využita data statistického úřadu Evropské unie Eurostat a dále data programu Copernicus Land Monitoring Service – Urban Atlas. Jedná se o volně dostupná data, která jsou průběžně aktualizována. Zpracování dat přináší studentům znalost o evropských geografických tématech, která souvisí se zaměřením jejich studia. Pro výuku shlukové analýzy byla použita data průměrné týdenní pracovní doby a podíl pracovníků v jednotlivých odvětvích. Analýza časových řad je procvičována na datech železniční dopravy. Neuronové sítě nebo frekventované sady lze aplikovat na data Urban Atlas o využití území Evropských měst.

Klíčová slova: Eurostat, Urban Atlas, Data Mining, geoinformatika

Abstract

The Master's degree in Geoinformatics and Cartography includes Data Mining course. The data of the Statistical Office of the European Union Eurostat and the data of the Copernicus Land Monitoring Service - Urban Atlas were used in practical lectures. Mentioned data are freely available. Data processing provides students with knowledge of European geographical topics related to their study. The data on average weekly working hours and the share of workers in each industry were used for learning cluster analyses. Time series analysis is practiced on rail transport data. Neural networks or frequent datasets can be applied to Urban Atlas data on land use of European cities.

Keywords: Eurostat, Urban Atlas, Data Mining, Geoinformatics

Úvod

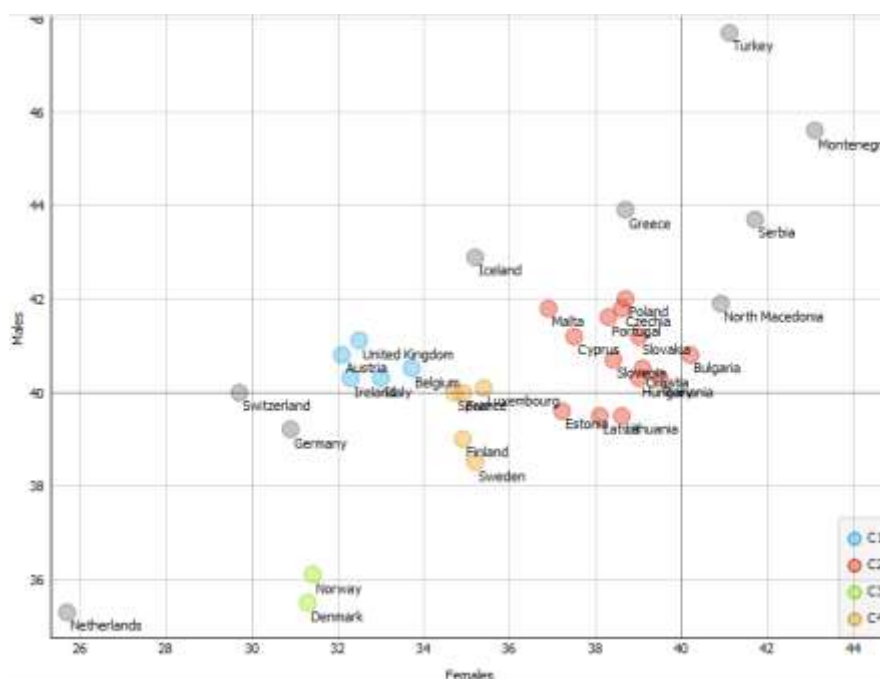
Praktická výuka na konkrétních datech je nezbytnou součástí výuky a napomáhá k pochopení teoretického výkladu jednotlivých metod. Při hledání vhodných aktuálních dat pro předmět Data Mining byly vytipována statistická databáze Evropské Unie Eurostat a její jednotlivé statistiky jako např. přeprava cestujících (Eurostat 2022). Druhým vhodným zdrojem z hlediska oboru Geoinformatika a kartografie, kam je předmět Data Mining zařazen, se jevila databáze projektu Copernicus – Urban Atlas (UA) (Copernicus Programme 2020), kde jsou dostupná data o využití území v městských funkčních oblastech. Vzhledem k tomu, že výuka předmětu Data Mining je převážně realizována v softwaru Orange, tak byla pro praktickou část cvičení nachystána učebnice s ukázkou praktických příkladů (Dobešová 2022). Některé příklady v učebnici právě čerpají data z výše uvedených zdrojů. Navíc je dostupná i cvičebnice pro samostatné procvičování úloh, která i vysvětluje postup a získávání dat v uvedených dvou databázích. Obě učebnice jsou ke stažení na webu projektu <http://UrbanDM.upol.cz>. V textu

tohoto příspěvku budou nastíněny některé vybrané úlohy, které jsou procvičovány ve výuce a vychází z dat ze zmíněných databází.

1 Použití dat z databáze Eurostat

Pro základní úlohy inspekce dat, jako je zjišťování minimálních, maximálních, průměrných, mediánových hodnot, četností, chybějících hodnot, odlehlých hodnot lze použít řadu dat z databáze Eurostat. Každá datová sada má unikátní označení kódem a tak lze data aktualizovaná data opakovaně vyhledávat, odkazovat a stahovat, popř. upravovat výběry.

Jednou z vhodných datových sad je sada s kódem LFSA_EWHUN2, která obsahuje *průměrný počet odpracovaných hodin v týdnu* v každé zemi. Počty pracovních hodin jsou uvedeny samostatně pro ženy, muže a celkem. Na těchto datech lze dobře demonstrovat odlehlé hodnoty, kdy v Turecku mají muži nejdelší pracovní týden a to skoro 48 hod. Pomocí krabicového grafu lze zobrazit rozpětí hodnot, zajímavé je i použití houslového grafu, pro základní představu o datech. V rámci předmětu Data mining se jako typická úloha procvičuje shlukování. Na těchto datech lze dobře vyšetřit shluky podobných států v Evropě, které se podobají počtem pracovních hodin jak pro muže, tak pro ženy. Prakticky byla použita shlukovací metoda k-Means a DBSCAN (Obr. 1). Studenti nejen procvičují postupy zpracování dat ale hlavně i interpretaci výsledků a vhodnost použitých metod a nastavení jejich parametrů.



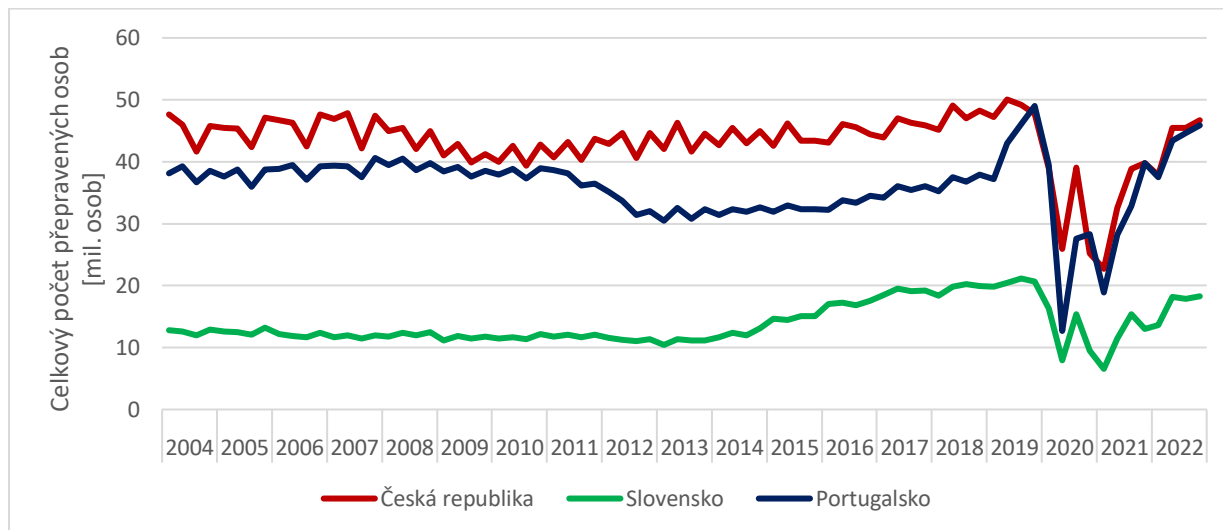
Obrázek 1: Shlukování metodou DBSCAN států EU podle průměrného počtu pracovních hodin v týdnu mužů a žen (parametry Core point Neighbors=2, Distance=1)

Druhý příklad dobrého zdroje jsou data Eurostatu o *počtu pracovníků v jednotlivých odvětvích* (NACE) s kódem NAMA_10_A64_E. Státy Evropy se výrazně liší počtem pracovníků v jednotlivých odvětvích. Zde už se jedná o multidimenzionální data, kdy na první úrovni členění je každý stát popsán počtem pracovníků v 21 typech odvětví (Masopust et al. 2021). Zde si studenti musí nejprve absolutní počty pracovníků přepočítat na relativní hodnoty podle počtu obyvatel v produktivním věku. Tato úprava spadá pro předpřípravu dat, která je také teoreticky přednášena v úvodních hodinách předmětu. Na tato data lze dobře aplikovat analýzu hlavních komponent. Zajímavé jsou korelace, kdy např. pozitivně korelují počty zaměstnanců v odvětví „školství“ a odvětví „umění, zábava, rekreace“. Díky hierarchickému shlukování lze

určit státy, které jsou si podobné počtem pracovníků ve stejných odvětvích. Naopak interpretačně zajímavé jsou odlehlé státy jako je Lucembursko nebo Rumunsko (Dobešová et al. 2022).

Databáze Eurostat poskytuje statistiku *počtu cizinců žijících v jednotlivých zemích EU*. Statistika je z roku 2011. Pro každý stát jsou uvedeny počty cizinců podle kontinentu, odkud do konkrétní země přišli. Na těchto datech lze vyšetřovat podobné země podle složení cizinců.

Pro téma zpracování časových řad lze v databázi Eurostat nalézt řadu vhodných dat. Ve výuce byla použita data o *počtu přepravených osob po železnici*, která je uváděna po čtvrtletích. V datech je dobře viditelný propad přepravy v pandemickém roce 2020 a 2021 (Obr. 2).



Obrázek 2: Vývoj počtu přepravených osob od roku 2004 do roku 2022 v České republice, na Slovensku a v Portugalsku (Eurostat 2022)

2 Použití dat Urban Atlas

Projekt Urban Atlas poskytuje data u využití území ve skoro 800 městských funkčních oblastech (tzv. FUA) v rámci celé Evropské Unie. Jedná se o prostorová data, kde je pro verzi 2018 vymezeno 27 kategorií využití území, jako jsou různé plochy s různou hustotou zástavby, silnice, plochy zeleně, vodní plochy, komerční průmyslové a vojenské plochy, různé druhy ploch zemědělsky využívaných až po plochy letišť (Obr. 4).

Na těchto datech lze nachystat řada úloh. Od statistických ukazatelů o počtu, rozloze jednotlivých ploch až po využití neuronových sítí pro zjištění prostorových vzorů. Data byla použita pro demonstraci nalezení podobných měst podle prostorového uspořádání ploch. Jako vstup byly použity kruhové mapové výřezy center měst předchystané v ArcGIS softwaru (Janoušek 2019). Software Orange disponuje doplňkem, který nabízí natrénované neuronové sítě, kde nejlepší výsledky pro tato data přinášela síť Painters. Studenti jsou tímto příkladem uvedeným v učebnici (Dobešová 2022) seznámeni s výsledky zkoumání prezentovaném v publikovaných vědeckých článcích (Dobesova 2019) (Dobesova 2020).

Data byla využita i pro semestrální úlohu, kdy každý student zpracoval několik vybraných měst.



Obrázek 3: Zdrojová data Urban Atlas pro město Girona ve Španělsku

Data Urban Atlas byla použita i na zjišťování frekventovaných sad sousednosti ploch s různým využitím. Data mining metoda asociačních pravidel a frekventovaných sad je v rámci předmětu také přednášena. Frekventované sady, které vycházejí z prostorové sousednosti byly řešeny v rámci diplomové práce (Novák 2023). Jednalo se o aplikaci analýzy nákupního košíku (MBA) a generování transakcí sousednosti polygonů. Předpokládám zařadit postup pro zjišťování frekventovaných sad navrženého v diplomové do výuky v nadcházejícím akademickém roce.

Závěr

Použití dat Eurostat a Urban Atlas ve výuce přináší několik výhod. Data jsou volně dostupná ke stažení. Databáze jsou průběžně aktualizované a jsou tak dostupná aktuální data. V případě časových řad dopravy se jedná např. o čtvrtletní data od roku 2004. Další výhodou je, že se jedná o reálná data na rozdíl od „umělých“ trénovacích datasetů dodávaných často k učebnicím či softwarům. Výhodou je velký rozsah dat, kdy lze zadávat samostatné práce jednotlivým studentům ze cvičení, či ve formě semestrálních úloh, kdy každý student pracuje se svým zadaným výběrem dat. Stažená data obsahují metadatový popis o významu dat, časovém rozsahu atd. Což nepřímo edukuje studenty mimo náplň předmětu, kdy pracují se správným metadatovým popisem dat (např. souřadnicový systém u dat Urban Atlas). Použití těchto dat přináší povědomí o Evropském regionu a v řadě socioekonomických témat. Přínosná pro rozhled studentů je i interpretace dat, kdy na základě zjištěných faktů mohou kombinovat různé informace. Studenti tak získávají nejen dovednosti v použití metod data mining, ale rozšiřují se jejich znalosti v geografii a geoinformatice. Další zdroje vhodných dat lze nalézt v knize (Pászto et al. 2020). Doporučit lze YouTube kanál sw Orange (Pretnar 2016).

Poděkování

Výuka je podpořena projektem Erasmus+ Programme of the European Union, Jean Monnet Module, Project No. 620791-EPP-1-2020-1-CZ-EPPJMO-MODULE “Data mining and analyzing of urban structures as contribution to European Union studies“.

Literatura

- Copernicus Programme (2020). Urban Atlas. [5-5-2022] Dostupné z:
<https://land.copernicus.eu/local/urban-atlas>.
- Dobesova Z. (2019). The Similarity of European Cities Based on Image Analysis. In: Silhavy R., Silhavy P. (ed) *Advances in Intelligent Systems and Computing*. Springer, Cham, 341–348.
- Dobesova Z. (2020). Experiment in Finding Look-Alike European Cities Using Urban Atlas Data. *ISPRS Int J Geo-Information* 9:406. <https://doi.org/10.3390/ijgi9060406>
- Dobešová Z. (2022) ORANGE, Praktický návod do cvičení předmětu Data Mining. Univerzita Palackého v Olomouci, Olomouc, Dostupné z: <https://www.vydavatelstviupol.cz/cz/978-80-244-6086-4>
- Dobešová Z., Macků K., Kučera M. (2022) Výuka geoinformatických předmětů na příkladech dat Evropské unie. In: *Symposium GIS Ostrava 2022*. Ostrava
- Eurostat (2022) Passengers transported (detailed reporting only) - (quarterly data). [12-10-2022] Dostupné z: https://ec.europa.eu/eurostat/databrowser/product/page/RAIL_PA_QUARTAL. 1
- Janoušek M. (2019). Porovnání urbánního prostoru pomocí kruhových výsečí. Univerzita Palackého v Olomouci, Diplomová práce, [7-21-2023]. Dostupné z: <https://theses.cz/id/w91ot7/>
- Masopust J., Dobesova Z., Macků K. (2021). Utilisation of EU Employment Data in Lecturing Data Mining Course BT - Artificial Intelligence in Intelligent Systems. In: Silhavy R (ed). *Springer International Publishing*, Cham, 601–616
- Novák P. (2023) Porovnání sousednosti využití území evropských měst pomocí frekventovaných sad. Univerzita Palackého v Olomouci, Diplomová práce. [7-21-2023]. Dostupné z: <https://theses.cz/id/h6kc79/>
- Pászto V., Redecker A., Macků K, et al. (2020). Data Sources. In: Pászto V., Jürgens C., Tominc P., Burian J. (eds) *Spationomy: Spatial Exploration of Economic Data and Methods of Interdisciplinary Analytics*. Springer International Publishing, Cham, 3–38
- Pretnar A (2016). All I See is Silhouette. [11-9-2021] Dostupné z:
<https://orangedatamining.com/blog/2016/03/23/all-i-see-is-silhouette/>.

Kontaktní údaje

doc. Ing. Zdena DOBEŠOVÁ, Ph.D.

Katedra geoinformatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci

17. listopadu 50, 771 46 Olomouc

e-mail: zdena.dobesova@upol.cz

CAX/PLM SYSTÉMY NÁSTROJEM KOMUNIKACE

CAX/PLM SYSTEMS CAX/PLM SYSTEMS AS A COMMUNICATION TOOL

Karel Dvořák

Abstrakt

Nástrojem komunikace v průmyslové technické praxi jsou paralelně s textovými výstupy i grafická znázornění ve formě zobrazení technických řešení, grafů, schémat. Grafická znázornění představují i různé typy a kategorie technických výkresů v papírovém a především digitálním provedení. Technický výkres v současnosti představuje základní komunikační médium ve strojírenské a příbuzné průmyslové praxi. Aktuální standardy konstruování, přípravy výrobních technologií a řízení životního cyklu výrobku dávají k dispozici další informační východiska pro zpřesnění, zrychlení a celkové zefektivnění komunikace. Významným komunikačním objektem je 2D, nebo 3D model, který je nositelem rozsáhlého spektra kvantitativních i kvalitativních informací.

Klíčová slova: 3D model, 2D výkres, Komunikace

Abstract

Along with textual outputs, graphic representations in the form of displaying technical solutions, graphs, and diagrams are tools of communication in industrial technical practice. Graphic representations also represent various types and categories of technical drawings in paper and especially digital form. The technical drawing currently represents the basic communication medium in engineering and related industrial practice. Current design standards, preparation of production technologies and product life cycle management provide additional information sources for making communication more precise, faster and overall more efficient. An important communication object is a 2D or 3D model, which carries a wide range of quantitative and qualitative information.

Keywords: 3D model, 2D drawing, Communication

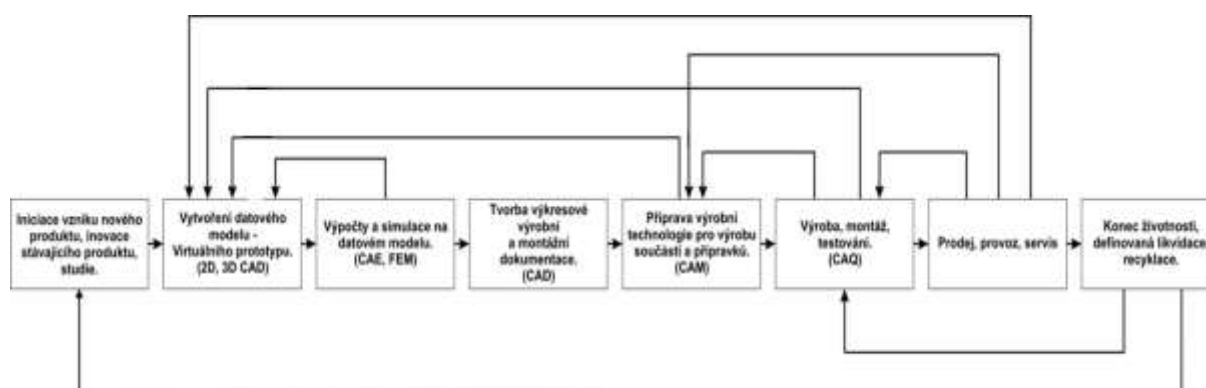
Úvod

Komunikační platformy, využívané v průmyslové praxi nabízejí široké spektrum přenosu digitálních informací v textové, grafické a další multimediální formě. Cílem je zajistit spolehlivý tok příslušných informací v průběhu celého životního cyklu produktu, kterým může být součást, sestava, nebo proces v některé příslušné vztažené formě. Obsažené informace jsou průběžně zpracovány a využívány v relevantních fázích životního cyklu produktu. Informace jsou současně archivovány pro opakované použití v rámci řízení podobných, nebo obecných životních cyklů následných produktů. Předložený text uvádí do problematiky využití informačního potenciálu primárně digitálních modelů virtuálních prototypů výrobků. Problematika je demonstrována na strojírenských produktech, je však uplatnitelná v dalších odvětvích, kde je výhodné pracovat s modelovou geometrií na úrovni virtuálních prototypů. Uvedené postupy jsou uplatňovány a průběžně optimalizovány na aplikační i procesní úrovni v průmyslové praxi. Implementace postupů do procesů technického vzdělávání zvyšuje

kompetenci absolventů technických oborů pro jejich uplatnění v rámci profesní praxe (Dvořáková, 2022). Ze širokého portfolia informačních atributů jsou zdůrazněny především geometrické, vizuální a textové informace, integrované v digitálním modelu příslušného virtuálního prototypu.

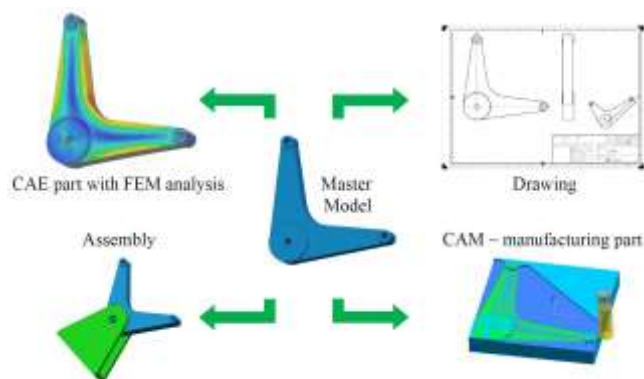
1 CAD model v kontextu životního cyklu výrobku

V kontextu obecného schématu životního cyklu výrobku představuje tvorba 3D modelu počáteční fázi návrhu nového, nebo inovovaného produktu. Schématické znázornění fází celého životního cyklu, s vyznačením informačních toků, je na obr. č. 1. Ze schématu je zřejmý význam nástrojů pro tvorbu digitálního modelu – CAD, Computer Aided Design.



Obrázek 1: Schéma základních fází životního cyklu nového, nebo inovovaného produktu

Na fázi tvorby 3D digitálního modelu navazují fáze provádění výpočtů a simulací pro optimalizaci návrhu výrobku. Východiskem pro provádění výpočtů a simulací prostřednictvím nástroje CAE – Computer Aided Engineering jsou obvykle modelová data a informace z modelů, vytvořených v CAD. Optimalizovaná CAD data jsou východiskem pro přípravu výrobních technologií, především číslíkově řízeného obrábění, případně 3D tisku (Dvořák, 2018) (Dvořák, 2021) s využitím nástrojů CAM – Computer Aided Manufacturing. Další postupy zahrnují např. definování procesů řízení automatických zařízení (Belda, 2019). Jednotlivé digitální modely součástí jsou na digitální virtuální úrovni spojovány do sestav pro definování a analýzu montážních a funkčních charakteristik. Významným produktem virtuálního prototypu sestavy je kusovník – soupis položek, který je významným východiskem následného zpracování především pro účely logistiky ve fázích výroby a servisu. Znázornění využití digitálního modelu pro navazující a vztažené aktivity s využitím informačních atributů je na obr. č. 2. Znázorněný 2D výkres lze uvažovat ve formě klasické koncepce s využitím pravidel zobrazování, kótování a popisování, která jsou určena příslušnými mezinárodními průmyslovými standardy. 2D koncepce výkresů je v současné době částečně nahrazována, nebo doplňována 3D modelem, obsahujícím uvedené informace v odpovídající vizuální formě (Dvořák, 2013). Příklad vizualizace modelu, obsahujícího vybrané kóty je na obr. č. 3. Definování kót je s využitím nástroje PMI – Product Manufacturing Information. Model s vizualizací informací lze využít samostatně, nebo v kombinaci s 2D výkresem.

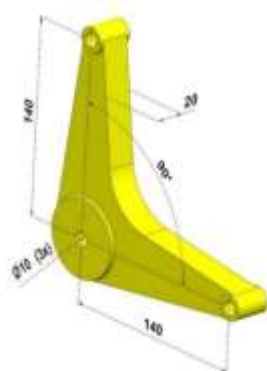


Obrázek 2: Princip společného výchozího modelu – Master Model Concept

2 Informační struktura primárních CAD dat

Primární informací, obsaženou v CAD datech na úrovni křivkového, plošného, nebo objemového modelu jsou rozměry (Dvořák, 2016), reprezentované vektorovým analytickým vyjádřením. Rozměrové charakteristiky lze pro účely komunikace kvantifikovat několika více způsoby. Základem jsou soubory parametrů, obsažené v modelu prostřednictvím identifikátorů a přiřazených hodnot. Další možností je zpracování hodnot příslušných rozměrů změřením prostřednictvím nástrojů diagnostiky. V navazujících fázích životního cyklu lze využít zpracování parametrů kót výkresové dokumentace, případně kvalitativních veličiny tvarových analýz. Obvyklým postupem je kombinování některých, nebo všech uvedených postupů.

Uvedené zdroje informací jsou zahrnuty v master-model konceptu dle vizualizace obr. č. 2. Zvyšování efektivity přístupů je možné optimalizací jednotlivých větví, zejména vazby výchozího 3D modelu na 2D výkres (Dvořák, 2023). Komplexní využití tvarových prvků geometrie modelu, společně s integrovanými rozměrovými parametry a současně vloženými hodnotami příslušných rozměrů prostřednictvím kót dává k dispozici významný soubor informací a vytváří významné komunikační médium mezi fázemi celého životního cyklu výrobku. Vnitřní i vnější parametry mohou být v rámci modelu dále zpracovány prostřednictvím matematických vztahů pro generování dalších souborů informací v textové, nebo tabulkové formě. Příklad 3D modelu s kótami PMI – Product Manufacturing Information je na obr. č. 3.



Obrázek 3: Příklad 3D modelu s výrobními informacemi.

Procesní úroveň řízení vývojových a výrobních dat představují přístupy a systémy PLM – Product Lifecycle Management, řízení životního cyklu produktu – výrobku. Principiálně jde o sled navazujících aktivit a příslušných zpětných vazeb, kde jednotlivé fáze představují konkrétní úrovně pod-procesů. Objekty řízení procesů představují atributy ve formě textových, numerických, případně logických proměnných. Atributy CAX dat na úrovni všech aplikací jsou vytvářené automaticky v průběhu parametrického modelování, tvorby sestav a výkresů. Současně jsou generovány v rámci dalších aktivit na úrovni provádění simulací, analýz, nebo přípravy výrobních technologií. Paralelně s automatickou tvorbou atributů, integrovaných uvnitř jednotlivých CAX objektů lze vytvářet vlastní proměnné na úrovni textových, číselných, nebo logických hodnot, včetně tvorby výrazů mezi jednotlivými proměnnými, kde na základě hodnot, vypočítaných prostřednictvím výrazů mohou být generovány parametry pro řízení následné geometrie, případně pro zpracování a řízení procesů na procesní úrovni systému řízení životního cyklu produktu. Příklad výpisu části seznamu proměnných pro model, znázorněný na obr. č. 3 je na obrázku č. 4. Transfer stejných dat do tabulky vnější aplikace, např. MS Excel je znázorněn na výřezu tabulky obr. č. 5. Metoda zpracování dat v rámci CAX aplikace, automaticky mezi jednotlivými systémy, případně manuálně spočívá v propojení, nebo exportu atributů na úrovni výstupu nejčastěji textových, nebo tabulkových formátů. Na obecné bázi formátů souborů jsou následně definovány formáty pro konkrétní nástroje, nejčastěji formáty *.txt, *.xls, *.html. Uvedená komunikace a zpracování proměnných může probíhat obousměrně, tzn. atributy generované obvyklými postupy vně vnitřní strukturu systému lze zpracovávat v rámci řetězce aktivit a naopak, vnějšími atributy lze řídit kteroukoli fází komunikace v rámci jednotlivých aktivit. Typickým příkladem je řízení parametrické geometrie CAD dat tabulkou MS Excel, což je významné z hlediska komunikace pro zapojené pozice uživatelů bez CAD kompetencí. Systémová podpora postupů může být na úrovni přímého propojení příslušných formátů souborů jednotlivých aplikací, případně je možná tvorba specifických nástrojů pro zvýšení uživatelského komfortu transferu dat. Za nejvyšší úroveň komunikace lze aktuálně považovat nástroje PLM, zahrnující všechny úrovně struktury obecné architektury informačních systémů, tzn. vrstva technologická, aplikační, business-procesní a komunikační.

#	Name	Formula	Value	Units	Dimensionality	Type	Source
1	Default Griid						
2				mm	Length	Number	
3	p3	0	0	mm	Length	Number	(SKETCH_000:Sketch(1) Plane Off...
4	p5	0	0		Unitless	Number	
5	p6	0	0		Unitless	Number	
6	p7	0	0		Unitless	Number	
7	p8	200	200	mm	Length	Number	(SKETCH_000:Sketch(1) Parallel ...
8	p9	200	200	mm	Length	Number	(SKETCH_000:Sketch(1) Parallel ...
9	p10	0	0	mm	Length	Number	(Extrude(2) Start Limit)
10	p11	25	25	mm	Length	Number	(Extrude(2) End Limit)
11	p60	15	15	mm	Length	Number	(Ø15 Hole(4) Diameter)
12	p68	0.0	0	°	Angle	Number	
13	p113	70	70	mm	Length	Number	(Ø15 Hole(4) Perpendicular Dimen...
14	p114	70	70	mm	Length	Number	(Ø15 Hole(4) Perpendicular Dimen...
15	p115	70	70	mm	Length	Number	(Ø15 Hole(4) Perpendicular Dimen...

Obrázek 4: Výpis části seznamu proměnných v rámci aplikace pro zpracování v rámci komunikačních procesů.

	A		C
1	//	Version:	3
2	[MilliMeter]p3		0
3	p5		0
4	p6		0
5	p7		0
6	[MilliMeter]p8		200
7	[MilliMeter]p9		200
8	[MilliMeter]p10		0
9	[MilliMeter]p11		25
10	[MilliMeter]p10		15
11	[Degrees]p8	0.0	
12	[MilliMeter]p113		70

Obrázek 5: Výpis části seznamu proměnných exportovaných do tabulky Excel pro zpracování v rámci komunikačních procesů.

3 Systém řízení životního cyklu výrobku v roli komunikačního nástroje.

Vysokou míru efektivity komunikace v rámci řízení životního cyklu výrobku zajišťují nástroje PLM – Product Lifecycle Management, v předchozích generacích označované PDM – Product Data Management. Obecně lze PLM považovat za databázový systém s vysokou mírou přizpůsobení specifickým procesům a současně podporující definování a řízení příslušných procesů. Základní funkcionality systému PLM spočívají v integrování modelových, výkresových a dalších souvisejících dat dle definice a schématu na obr. č. 2. Kromě modelových, výkresových, analytických a technologických dat je významnou položkou soupis položek – kusovník. Transfer dalších souvisejících informací zajišťují atributy na úrovni metadat. V rámci řízení příslušných procesních work-flow jsou definována řízení odpovědnosti za příslušné položky a podprocesy, podporována práce v týmu a řízení úrovně vyspělosti položek v rámci příslušných fází životního cyklu výrobku. Paralelně s řízením dat a procesů jsou generovány příslušné reporty pro archivaci a další zpracování. Významnou funkcionalitou je možnost propojení s dalšími podnikovými systémy – ERP. Problematika zvyšování efektivity a definování charakteristik propojení systému PLM a ERP je řešena např. v (Semerád, 2020). Nezanedbatelným faktorem je možnost vysoké úrovně zabezpečení dat.

Základní objekty přímých dat v systému představují položky s další definovanou strukturou, znázorněnou na obr. č. 6. Kromě CAD dat jsou zřetelné zejména formuláře atributů a datasey zahrnující obsahově příslušná data konkrétních aplikací.



Obrázek 6: Příklady dvou různých položek v rámci PLM systému, zahrnujících data příslušných aplikací.

Závěr

Komunikační principy představují klíčový faktor řízení životního cyklu výrobku v průmyslových procesech. Integrovaní standardních i pokročilých postupů v rámci technického vzdělávání představuje významný atribut efektivního získávání profesních kompetencí absolventů pro uplatnění v průmyslové praxi. CAD data představují digitální platformu, zahrnující široké spektrum kvalitativních i kvantitativních informací pro efektivní využití při řešení dílčích i komplexních úloh fáze koncepční studie, konstrukce, optimalizace, technické přípravy výroby, provozu, servisu a definovaného ukončení životního cyklu.

Literatura

- Dvořák, K. (2023). Přístupy k tvorbě technické dokumentace 3D tištěných produktů. In *Inovace v aditivních technologiích – INAM 2023 (7-8)*. Jihlava: Vysoká škola polytechnická Jihlava, Tolstého 1556/16, 586 01 Jihlava.
- Dvořáková, J., & Dvořák, K. (2022). CAX/PLM SYSTÉMY V TECHNICKÉM VZDĚLÁVÁNÍ. In *Informatika 2022 (7-12)*. Jihlava: Vysoká škola polytechnická Jihlava.
- Dvořák, K., & Dvořáková, J. (2021). Topological Optimization of a Component Made by the FDM Method. *International Journal of Mechanical Engineering and Robotics Research*, 10(2). doi: 10.18178/ijmerr.10.2. 67-71
- Belda, K., & Dvořák, K. (2019). Path Modeling and 3D Robot Visualization for Model-Based Control of Articulated Robots. In *Preprints of the 15th European Workshop on Advanced Control and Diagnosis (ACD 2019)*. Bologna, IT: Springer International Publishing. doi: 10.1007/978-3-030-85318-1_52, 1-16.
- SEMERÁD, Tomáš. (2020) Propojení systému PLM a ERP. Jihlava, 49 s. Bakalářská práce. Vysoká škola polytechnická Jihlava. Vedoucí práce Karel Dvořák.
- Dvořák, K., & Dvořáková, J. (2018). Přístupy k přípravě dat pro 3D tisk. *LOGOS POLYTECHNIKOS*, 2018(4), 4-15. Dostupné z <https://www.vspj.cz/soubory/download/id/7408>
- Dvořák, K., Bílek, M., & Dvořáková, S. (2016). Free form surface modeling and analysis. In *WSCG 2016 Posters Proceedings*. Plzeň: UNION Agency, 17-20.
- Dvořák, K., (2013). Management of parametric CAD model by external tools. *Applied Mechanics and Materials*. roč. 390, č. 1, s. 616-620. DOI: 10.4028/www.scientific.net/AMM.390.616.

Kontaktní údaje

Ing. Bc. Karel Dvořák, Ph.D.
Katedra technických studií
Vysoká škola polytechnická Jihlava
Tolstého 16, 586 01, Jihlava
e-mail: karel.dvorak@vspj.cz

KORPUS STUDENTSKÝCH ZDROJOVÝCH KÓDŮ – SYNTAKTICKÁ EKVIVALENCE

CORPUS OF STUDENTS' SOURCE CODES - SYNTACTIC EQUIVALENCE

Jiří Fišer, Pavel Beránek, Jiří Škvor

Abstrakt

V rámci budování korpusu studentských zdrojových kódů (Q-Corpus) je aktuálně řešena i problematika kvantifikace podobnosti mezi zdrojovými kódy (resp. jejich fragmenty). Motivací je kromě nalezení vztahů mezi výstupy různých studentů i případná identifikace kódu převzatého z generativních jazykových modelů (AI chatbots), což by mělo umožnit mapovat jejich využití. Pozornost je zaměřena na syntaktickou podobnost, tj. použití stejných jazykových konstrukcí na odpovídající úrovni hierarchie.

Klíčová slova: korpus, syntaktická ekvivalence, generativní jazykové modely

Abstract

Within the framework of building a corpus of students' source code (Q-Corpus), the problem of quantifying similarity between source codes (or fragments thereof) is currently being addressed. The motivation, besides finding relationships between the outcomes of different students, is the possible identification of code taken from language generative models (AI chatbots), which should allow to map their usage. The focus is on syntactic similarity i.e. the use of the same language constructs at the appropriate level of the hierarchy.

Keywords: corpus, syntactic equivalence, generative language models

Úvod

Již od samého začátku projektu studentského korpusu byla zvažována otázka testování shody zdrojových textů. Cílem není odhalení plagiátorství, ale využití této klíčové informace pro statistické zpracování, a to jak ve fázi předzpracování (odstranění duplikátů), tak i v jeho rámci, např. pro shlukovou analýzu.

V současnosti se však nabízí i využití pro testování vlivu generativních velkých jazykových modelů typu generative pre-trained transformers (GPT). Přestože jsou známy především díky schopnosti generování textů v přirozených jazycích, jsou využívány i pro generování zdrojových kódů. I když v oblasti zdrojových kódů se tyto modely používaly již dříve, neboť komplexita krátkých zdrojových kódů nedosahuje složitosti přirozených textů (Github Copilot, Codex model, 2021). Dosah těchto technologií se však výrazně zvýšil jejich integrací do univerzálních chatbotů, jako je ChatGPT (Radford, 2018).

Ačkoli jsou možnosti těchto modelů v praxi ještě relativně omezené, neboť slouží primárně jako generátory krátkých fragmentů kódu či triviálních programů, generovaný kód může být navíc suboptimální či chybný, mohou už nyní výrazně ovlivnit výuku základů programovacích jazyků (Finnie-Ansley, 2022). Zvládají totiž generovat kód na úrovni většiny absolventů úvodních kurzů programování (či v jistých oblastech i na úrovni výrazně vyšší), a to na základě zadání v přirozené řeči, včetně příkladů jak abstraktních, tak konkrétních (slovní úlohy). Navíc

jejich schopnosti budou rychle růst a v oblasti generování zdrojových kódů nelze očekávat regulaci ze strany státních orgánů.

Vliv těchto technologií na oblast výuky programovacích jazyků není možno ignorovat, byť vědecký výzkum této problematiky je teprve v počátcích. Vliv řešení založených na použití ChatGPT či dalších prediktivních modelů musí zohlednit i náš korpus.

V prvé řadě chceme přispěvatelům umožnit, aby mohli v rámci metadat uvádět i využití umělé inteligence. Studenti však samozřejmě mohou využívat řešení poskytovaná generativními modely i v případech, kdy to explicitně neuvedou. I v tomto případě by bylo vhodné určit ta řešení, která jsou obdobou řešení poskytovaných AI nástroji (s vědomím, že nástroje mohou poskytovat různá řešení, a to i při stejném zadání), aby mohla být tato skutečnost zohledněna při statistickém vytěžování korpusu (přestože nelze nikdy s absolutní jistotou potvrdit, že student převzal řešení umělé inteligence).

1 Syntaktická ekvivalence zdrojových kódů

Testování ekvivalence dvou zdrojových kódů není triviální úkol (Ain, 2019). Za prvé je nutné odlišit porovnání na sémantické a syntaktické úrovni. Dva zdrojové soubory lze považovat za sémanticky ekvivalentní, pokud pro všechny přípustné vstupy programy vyprodukují stejný výstup a skončí ve stejném stavu (normální ukončení, stejná nezachycená výjimka, nekonečný cyklus).

V případě zdrojových kódů uložených v korpusu je sémantická ekvivalence irelevantní, resp. je ověřována technikami určenými pro testování validity kódu (např. různými typy jednotkového testování).

Definice syntaktické ekvivalence je obtížnější a především neexistuje žádný přístup, který by byl univerzálně aplikovatelný.

Z důvodu zjednodušení jsme prvotně vycházeli z následujících východisek (omezení):

- syntaktická ekvivalence je běžná relace (tj. nikoliv například fuzzy relace či metrika),
- pokud jsou dva kódy syntakticky ekvivalentní, pak jsou i sémanticky ekvivalentní,
- syntaktická ekvivalence je definována jen pro zdrojové kódy ve stejném programovacím jazyce (striktně pro konkrétní překladač, reálně lze ekvivalenci uplatnit i pro skupinu dostatečně blízkých jazyků).

Nejjednodušším přístupem umožňujícím porovnání dvou zdrojových kódů je **znakové porovnání kanonického tvaru** dvou souborů, tj. souborů, ze kterých jsou odstraněny nevýznamné mezerové znaky a zbývající jsou převedeny do jednotné podoby (Whistler, 2022). Zásadním nedostatkem tohoto přístupu je skutečnost, že ani odstranění mezer a znaková normalizace neumožní definovat dostatečně univerzální ekvivalenci (tzv. klon 1 druhu, viz např. Saini 2019). Problémem je skutečnost, že i v kanonickém tvaru jsou konstrukce či data, které vůbec či jen velmi omezeně ovlivňují sémantiku programů:

- konkrétní tvary některých identifikátorů
- konkrétní hodnoty některých literálů (především řetězců)
- pořadí některých konstrukcí (například definicí funkcí či tříd)

V případě identifikátorů lze relativně snadno odlišit názvy, jejichž konkrétní podoba nese klíčovou informaci, od těch, jejichž konkrétní podobu si mohou studenti zvolit. Fixní identifikátory jsou totiž definovány v jiných souborech, resp. v případě rozsáhlejších projektů

s více soubory mimo projektový adresář (typicky jsou to například jména knihovnických tříd či metod). Naopak identifikátory lokálních proměnných či proměnných viditelných v rámci jediného modulu by měly být v procesu testování ekvivalence plně transparentní (studenti by naopak měli být schopni tyto identifikátory substituovat).

V následující ukázce kódu (v sekci mé řešení) jsou použity čtyři externí (a tudíž sémanticky významné) identifikátory: *sum*, *int*, *input*, *print*. Zbývající identifikátory (*vowel_count*, *str*, *vowels*, *c*, *s*) jsou lokálně definované a měly by být transparentní (přestože *str* je běžně externí identifikátor vestavěné funkce, zde je zastíněn lokálním identifikátorem).

mé řešení

```
def vowel_count(str):
    vowels = "aeiouAEIOUY"
    return sum(int(c in vowels) for c in str)

s = input("Zadej ASCII řetězec: ")
print(f"Počet samohlásek je {vowel_count(s)}")
```

základní řešení ChatGPT

```
def pocet_samohlasek(text):
    samohlasky = set("aeiouAEIOU")
    pocet = 0
    for znak in text:
        if znak in samohlasky:
            pocet += 1
    return pocet
```

úprava po námitce, že „y“ je v češtině také samohláska

a požadavku na použití funkce `sum`

```
def pocet_samohlasek(text):
    samohlasky = set("aeiouAEIOUY")
    return sum(1 for znak in text if znak in samohlasky)
```

Celou konverzaci s ChatGPTsi můžete prohlédnout na https://1url.cz/@GPT_1.

Situace u ekvivalence literálů je komplikovanější. U řetězcových literálů lze rozlišit tzv. interní řetězce (užívané algoritmem včetně identifikátorů externích protokolů, např. HTML značky) od externích (souvisejících typicky s interakcí s lidskými uživateli). V našem příkladu je interním literálem seznam řetězců (jeho změnou se může podstatně změnit chování programu), zatímco ukázkové řetězce, výzvy a výstupy mohou být změněny bez podstatného vlivu na funkci programu (a měly by být proto při testování ekvivalence transparentní). Externí řetězce typicky obsahují texty v přirozeném jazyce, a proto bývají při profesionální tvorbě softwaru překládány, protože jsou často v kódu explicitně označovány (mechanismy se liší podle jazyka a běžně i podle využití konkrétního vývojového rámce).

Kromě řetězcových literálů tento rozdíl existuje u číselných literálů, i když u nich převažují skutečné konstanty počínaje elementárními celými čísly (0-2) a především matematické, fyzikální a měrné konstanty. Navíc se u těchto konstant preferuje využití symbolických identifikátorů (s výjimkou elementárních celých čísel).

Aby bylo možno při testování ekvivalence zohlednit i konstrukce s nulovou či nízkou sémantickou relevancí, je nutné při porovnání vycházet ze strukturovanější reprezentace

zdrojového kódu, tj. reprezentace v níž lze odlišit jednotlivé typy tokenů, tj. atomických částí zdrojového kódu (literály, identifikátory, separátory) popřípadě určit jejich vzájemný vztah.

V procesu syntaktické analýzy (parsování) zdrojového kódu uvnitř kompilátoru/interpretu se postupně vytvářejí tři základní reprezentace programu

- tokenizovaný text (proud)
- derivační strom
- abstraktní syntaktický strom

Tokenizovaný text je lineárním seznamem tokenů, který však nenesé příliš mnoho informací o jejich kontextu. Na této úrovni nelze určit oblast viditelnosti proměnných, a tudíž ani odlišit jednotlivé proměnné. Stejně tak nelze určit sekce, na jejich vzájemném pořadí nezáleží.

Derivační strom je tvořen stejnými tokeny jako tokenizovaný text, avšak organizovanými do hierarchické struktury podle pravidel analytické gramatiky daného jazyka. Derivační strom tudíž zachovává všechny rysy původního programu (až na bílé znaky a poznámky) a zároveň umožňuje pracovat s programovými strukturami. Bohužel Python jako hlavní podporovaný jazyk nenabízí žádný standardní nástroj pro export či programový přístup k derivačnímu stromu.

Zbývá tak jen abstraktní syntaktický strom (AST), který eliminuje všechny tokeny, jež jsou po zavedení hierarchie nadbytečné (což se týká především různých typů separátorů, např. závorek). Použití AST umožňuje identifikovat klonování 2. druhu (parametrizované klony).

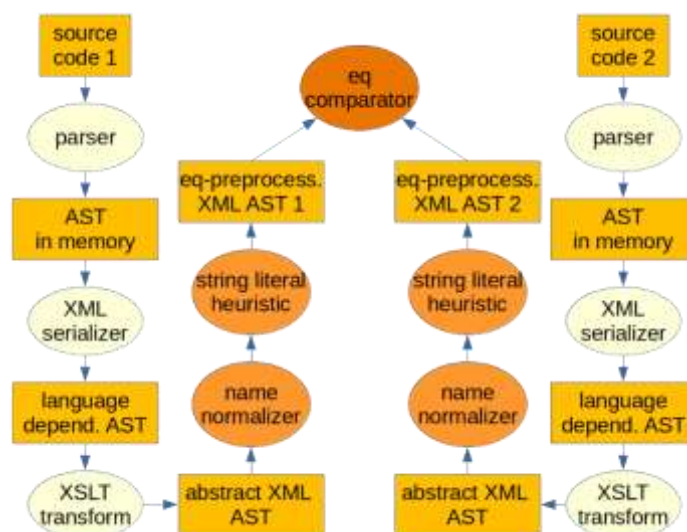
nevýhody: může snižovat citlivost při vyhledávání přímých plagiátů (větší počet falešně pozitivních identifikací, výsledek může záležet na konkrétní reprezentaci abstraktního syntaktického stromu)

výhody: lze snadněji odlišit skutečně originální řešení (tj. nikoliv jen jednoduché substituce), menší počet tříd ekvivalence (podpora statistických zpracování založených na shlukování), lze porovnávat i řešení v různých programovacích jazycích (za předpokladu, že sdílejí stejné programovací paradigma, a tím i základní programové konstrukce)

2 Implementace

Jedním z požadavků na nástroj pro porovnání zdrojových kódů je flexibilita. Nástroj by měl umožňovat volbu různých typů ekvivalence a měl by být snadno přenositelný mezi různými programovacími jazyky. Z tohoto důvodu byla zvolena obecná modulární struktura, v níž jsou jednotlivé akce realizovány pomocí modifikovatelné kolony (pipeline) filtrů nad XML reprezentací abstraktního syntaktického stromu (viz obrázek 1).

V první fázi transformace je zdrojový soubor načten a pomocí již existujícího kódu syntakticky analyzován a převeden do abstraktního syntaktického stromu, jenž je reprezentován stromem objektů v operační paměti. Tento strom je serializován do XML souboru v přímočarém mapování (objekty do stejnojmenných XML prvků, jednoduché atributy do XML atributů). Tento soubor tak přímo odráží model dané knihovny, jenž se mezi různými parsovacími knihovnami a především programovacími jazyky může výrazně lišit (běžné jsou různé idiosynkracie vzešlé z ne vždy přímočarého vývoje jazyka). Z tohoto důvodu je tato reprezentace pomocí XSLT transformována do jednodušší podoby, která je méně závislá na konkrétním jazyce (je navržen tak, aby v maximální míře podporoval procedurální jazyky hlavního proudu). Tyto transformační kroky jsou společné pro většinu nástrojů, jež jsou nad korpusem vytvářeny (tj. například výpočet různých metrik).



Obrázek 1: Syntaktické porovnání dvou souborů - fáze

Následující transformační filtry už jsou vytvářeny speciálně pro porovnání souborů. V současnosti jsou navrženy a ověřovány tři nástroje.

Normalizátor jmen upravuje jména interních proměnných. V první fázi normalizace jsou identifikovány externí identifikátory. Jsou to v první řadě všechny explicitně importované identifikátory. Identifikátory však mohou být importovány i hromadně (což není doporučeno, ale nikoliv zakázáno). Z tohoto důvodu jsou za externí identifikátory považovány i ta jména, která nejsou deklarována (funkce, třídy, formální parametry), resp. není do nich nikdy přiřazeno (proměnné).

V druhé fázi jsou nahrazena normalizovanými jmény, a to podle rozsahu oblasti viditelnosti a pořadí použití (nikoliv podle pořadí případné deklarace). Problematické jsou výrazy, u nichž AST nezachovává původní lineární pořadí, přičemž ani původní, ani nové pořadí nemusí být sémanticky relevantní. Z tohoto důvodu pracujeme na převodu běžných výrazů do kanonického tvaru (vzorem je kanonické uspořádání výrazů v jazyce *Wolfram Mathematica*, viz Expressions, c2023).

Filtr označený jako **string literal heuristic** se snaží identifikovat externí (tj. sémanticky nevýznamné řetězce), přičemž využívá dvou heuristických přístupů:

- *externí řetězce* jsou tvořeny texty přirozeného jazyka (použitelné jen u delších řetězců, robustnější řešení vyžaduje kvalitní nástroj pro strojové zpracování jazyků. Testujeme NLTK, který bohužel příliš nepodporuje češtinu.)
- *typický vnější kontext* (literál uvnitř několika běžných funkcí pro vstup je označen jako externí, literál, který je přiřazován do proměnné nebo na nějž jsou přímo aplikovány řetězcové operace, nerozhodnutelné jsou případy, kdy je řetězcový literál parametrem jiné než výše uvedené funkce)

Poslední funkcí je vlastní komparátor, který porovnává dva XML soubory a vrací boolovský výsledek. Nejjednodušším řešením je použití existujících nástrojů pro porovnání dvou XML souborů. Nejjednodušší transformace obou souborů do kanonického tvaru (nejlépe podle standardu, viz Boyer, 2002) a pak stačí přímé porovnání. Toto porovnání však vždy zohledňuje pořadí XML elementů (tj. i AST uzlů), přestože je v některých případech sémanticky nevýznamné (definice funkcí apod.). Protože počet takto zaměnitelných uzlů nebude velký, lze využít řešení testující všechny možné rozumné permutace (časová složitost je $O(n^2)$), naším

cílem je však najít obecnější model porovnávání XML souborů či prohledávání zefektivnit (např. použitím hashování).

3 Závěr

Aktuální implementace je prozatím využívána pro relativně krátké soubory s omezeným repertoárem programových konstrukcí (jež jsou typické pro začátečníky). I na těchto jednoduchých příkladech se ukazuje, že je stále, co řešit. Jmenovat lze například aliasy identifikátorů (což je snadné, ale vyžaduje přesněji definovat ekvivalenci identifikátorů) či možnost permutací jednoduchých příkazů s přesně definovaným postranním efektem (jednoduchá přiřazení). Statisticky významné výsledky však může přinést až zpracování většího počtu souborů.

Literatura

- Ain Q. U. , Butt W. H. , Anwar M. W., Azam F. and B. Maqbool, A Systematic Review on Code Clone (2019). In: IEEE Access, vol. 7, pp. 86121-86144, 2019, doi: 10.1109/ACCESS.2019.2918202. Boyer, John, Donald E. Eastlake a Joseph Reagle (2002). Exclusive XML Canonicalization Version 1.0: W3C Recommendation [online]. [2023-08-04]. Dostupné z: <https://www.w3.org/TR/xml-exc-c14n/>
- Finnie-Ansley, James; Denny, Paul; Becker, Brett A.; Luxton-Reilly, Andrew; Prather, James (2022). The Robots Are Coming: Exploring the Implications of OpenAI Codex on Introductory Programming. Australasian Computing Education Conference. ACE '22. New York, NY, USA: Association for Computing Machinery: 10–19. doi:10.1145/3511861.3511863.
- Radford, Alec; Narasimhan, Karthik; Salimans, Tim; Sutskever, Ilya (2018). Improving Language Understanding by Generative Pre-Training. OpenAI. p. 12.
- Saini Neha, Singh Sukhdip and Suman (2018) Code Clones: Detection and Management, Procedia Computer Science, Volume 132, 2018, Pages 718-727, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2018.05.080>
- Whistler, Ken, ed. (2022) Unicode Normalization Forms: Unicode Standard Annex #15 [online]. [2023-08-03]. Dostupné z: <http://unicode.org/reports/tr15/>
- Expressions, c2023. Wolfram Language & System: Documentation Center [online]. Wolfram [2023-08-04]. Dostupné z: <https://reference.wolfram.com/language/tutorial/Expressions.html>

Kontaktní údaje

Mgr. Jiří Fišer, Ph.D.

katedra informatiky Přírodovědecké fakulty Univerzity J. E. Purkyně

Pasteurova 3632/15, 400 96 Ústí nad Labem

e-mail: jiri.fiser@ujep.cz

ÚVOD DO IT NA TECHNICKÉ FAKULTĚ

INTRODUCTION TO IT AT THE TECHNICAL FACULTY

Miroslav Hrubý

Abstrakt

V příspěvku je popsán přístup k úvodním předmětům do oblasti IT na technicky zaměřené fakultě vojenské univerzity. Text vychází z aktuálního obsahu příslušných předmětů pro vojenskou a civilní studenty, které absolvují noví studenti v zimním semestru 1. ročníku studia. Příspěvek shrnuje vlastní zkušenosti autora získané při podílu na realizaci tří akreditovaných studijních programů na Univerzitě obrany.

Klíčová slova: IT, technická fakulta, programování, výuka

Abstract

The contribution describes the approach to the introductory subjects in the field of IT at the technically oriented faculty of the military university. The text is based on the current content of the relevant subjects for military and civilian students, which new students take in the winter semester of the 1st year of study. The paper summarizes the author's own experience gained while participating in the implementation of three accredited study programs at the University of Defence.

Keywords: IT, Technical faculty, Programming, Teaching

Úvod

Fakulta vojenských technologií (FVT) je jediná technicky zaměřená fakulta ze stávajících tří fakult Univerzity obrany (UO). Na FVT studují jak vojenští studenti pětiletých souvislých magisterských studijních programů, tak civilní studenti bakalářského a navazujícího magisterského studijního programu.

Příspěvek je zaměřen na předmět Informační technologie v ozbrojených silách (ITOS), který studují vojenští studenti studijních programů Vojenské technologie – elektrotechnické (VT-E) a Vojenské technologie – strojní (VT-S), a předmět Informační technologie, který studují civilní studenti bakalářského studijního programu Technologie pro obranu a bezpečnost – strojní (TOB-S). Oba předměty mají časovou dotaci 56 hodin. Studium předmětu ITOS i IT je zakončeno zápočtem.

1 Shrnutí dosavadních poznatků

Hlavním cílem práce studentů při studiu předmětu ITOS, resp. IT je získání základní báze znalostí a dovedností pro využití při dalším technicky zaměřeném studiu a budoucí praxi. Pro rychlý vývoj v oblasti IT, kdy dosažené vzdělání je nutno udržovat stálým vkladem vlastního času a energie, mají smysl zejména pochopení a akceptace odborné terminologie, získání odborného rozhledu v dané oblasti a efektivní využití informačních zdrojů, ať už tištěných nebo elektronických. Schopnost nejnovější informace získávat, analyzovat a hodnotit aktuální trendy

vývoje včetně formulace vlastních závěrů je významnou součástí profilu absolventů jak vojenského, tak civilního studia.

Tabulka 1: Struktura předmětu ITOS.

1.	P	4	Hrubý	Úvod do studia předmětu, základní pojmy
2.	P	4	Halouzka	Kybernetická bezpečnost
3.	P	2	Halouzka	Informační systémy AČR
4.	P	2	Halouzka	Viry v IT – malware
5.	P	2	Halouzka	Druhy zajištění OUI
6.	P	4	Kozak	Systémy velení a řízení
7.	P	2	Hrubý	Úvod do algoritmizace
8.	P	4	Hrubý	Datové typy a datové struktury
9.	P	4	Hrubý	Základní příkazy vyššího programovacího jazyka
10.	L	2	Hrubý	Základní programovací techniky
11.	P	4	Hopjan	Úvod do počítačových sítí, historie, modely síťové architektury.
12.	P	6	Hopjan	Protokoly linkové a síťové vrstvy
13.	P	6	Hopjan	Směrování, protokoly aplikační vrstvy, kybernetická bezpečnost
14.	P	6	Hopjan	Základní konfigurace síťového prostředí v poč. s ohledem na bezpečnost
15.	L	4	Hopjan	Návrh a realizace jednoduchých sítí

Zdroj: vlastní

Tabulka 2: Struktura předmětu IT.

1.	P	4	Hrubý	Úvod do studia předmětu, základní pojmy
2.	L	4	Hrubý	V/V programu a práce s polem
3.	L	2	Hrubý	Větvení a cykly
4.	L	2	Hrubý	Uživatelsky definované funkce
5.	L	2	Hrubý	Práce s řetězci
6.	L	4	Hrubý	Uživatelské rozhraní (tlačítka, vstupní textová pole, obrázky)
7.	P	2	Hrubý	Úvod do algoritmizace
8.	P	4	Hrubý	Datové typy a datové struktury
9.	P	4	Hrubý	Základní příkazy vyššího programovacího jazyka
10.	L	2	Hrubý	Základní programovací techniky
11.	P	4	Hopjan	Úvod do počítačových sítí, historie, modely síťové architektury.
12.	P	6	Hopjan	Protokoly linkové a síťové vrstvy
13.	P	6	Hopjan	Směrování, protokoly aplikační vrstvy, kybernetická bezpečnost
14.	P	6	Hopjan	Základní konfigurace síťového prostředí v poč. s ohledem na bezpečnost
15.	L	4	Hopjan	Návrh a realizace jednoduchých sítí

Zdroj: vlastní

2 Realizace zápočtu z předmětů ITOS a IT

Zápočet z předmětů ITOS a IT je udělen garantem předmětu na základě úspěšného zvládnutí písemného přezkoušení z první části předmětu (témata 1 až 10) a elektronického přezkoušení z druhé části předmětu (témata 11 až 15). Přezkoušení absolvují studenti v průběhu semestru ihned pro probrání příslušné části předmětu. U obou přezkoušení nesmějí studenti používat žádné pomůcky. Dále jsou pro ilustraci uvedeny příklady konkrétních zadání pro přezkoušení studentů z první části předmětu ITOS a IT.

Písemné přezkoušení z první části předmětu ITOS (témata 1 až 10) může mít např. zadání znázorněné na Obrázku 1. Platí pravidlo, že první dvě otázky se týkají ověření pochopení jednoduchých zápisů v programovacím jazyce JavaScript, další dvě otázky jsou zvoleny na pojmy a teoretický aparát z oblasti datových struktur, algoritmizace a programování a poslední dvě otázky jsou zvoleny z problematiky témat 2 až 6 uvedených v Tabulce 1.

1. Nakreslete situaci v paměti a zobrazení na displeji počítače po provedení skriptu uvedeného níže:

```
var Messi=5,Mbappe=-1,Ronaldo=-3;
Ronaldo=3*Mbappe;
Messi++;
document.write("Qatar: "+Messi*Ronaldo);
```

2. Nakreslete situaci v paměti a zobrazení na displeji počítače po provedení skriptu uvedeného níže:

```
var Kane=[2,6,3,-5],Giroud=[5,3,7,3,-2],i,Luka=0;
for (i=0;i<=2;i=i+2) {
    Luka=Luka+Kane[i]+Giroud[i];
}
alert(Luka);
```

3. Vysvětlete pojem **cyklus** a vyjádřete jednotlivé varianty **vývojovým diagramem**.
4. Převed'te dekadické číslo **18,125** do **dvojkové číselné soustavy**, a pak je vyjádřete v **normalizovaném tvaru**.
5. Jaké jsou v **NATO** uznávané „**válečné domény**“?
6. Co tvoří **systém velení a řízení**?

Obrázek 1: Příklad zadání písemného přezkoušení z první části předmětu ITOS. Zdroj: vlastní

Písemné přezkoušení z první části předmětu IT (témata 1 až 10) může mít např. zadání znázorněné na Obrázku 2. Na rozdíl od předmětu ITOS se přezkoušuje pouze problematika algoritmizace a programování. První dvě otázky jsou zaměřeny na ověření pochopení zápisů v jazyce JavaScript a další čtyři otázky ověřují pochopení pojmů a teoretického aparátu z oblasti datových struktur, algoritmizace a programování.

1. Co způsobí na displeji provedení skriptu uvedeného níže?

```
function alex(fcis1, fcis2, fcis3) {
    document.write("Číslo1: "+fcis1+"<br>");
    document.write("Číslo2: "+fcis2+"<br>");
    document.write("Číslo3: "+fcis3+"<br>");
    return fcis3-fcis1*fcis2;
}
alert(alex(-1, 2-3, 3));
```

2. Nakreslete situaci v paměti a zobrazení na displeji počítače po provedení skriptu uvedeného níže?

```
var i, krhounek=[15, 25, 31, 13, 11, 12];
for(i=0; i<=4; i=i+2) {
    krhounek[i]=0;
}
alert(krhounek[2]-krhounek[5]);
```

3. Co rozumíte pojmem **programovací techniky**?
4. Převed'te do **dvojkové číselné soustavy** dekadické číslo **35,7**. Výsledné dvojkové číslo zapište v **normalizovaném tvaru**.
5. Co rozumíte z pohledu programátora pojmem **výraz**?
6. Vysvětlíte z pohledu programátora pojem **větvění**. Znázorněte **vývojovým diagramem**.

Obrázek 2: Příklad zadání písemného přezkoušení z první části předmětu IT. Zdroj: vlastní

3 Výsledky a diskuze

Jak vyplývá z porovnání dat na Obrázku 1 a Obrázku 2, předměty ITOS a IT se liší v tématech 2 až 6. Vojenští studenti zde mají v rámci první části předmětu ITOS některá převážně vojensky zaměřená témata a civilní studenti programování jednoduchých problémů.

Na základě posledního hodnocení výuky předmětů ITOS a IT ze strany studentů, které poprvé proběhlo anonymně, bude od akademického roku 2023/2024 redukován počet přednášek ve prospěch cvičení a laboratorních cvičení. Rovněž dojde k mírné redukci druhé části předmětů ITOS a IT, která je zaměřena na počítačové sítě, a to ve prospěch algoritmizace a programování. Předmět ITOS bude mít od akademického roku 2023-2024 strukturu znázorněnou v Tabulce 3.

Tabulka 3: Struktura předmětu ITOS od akademického roku 2023-2024.

1.	P C	2 2	Hrubý	Úvod do studia předmětu, základní pojmy
2.	P	4	Buřita	Kybernetická bezpečnost
3.	P	2	Buřita	Informační systémy AČR
4.	P	2	Buřita	Viry v IT – malware
5.	P	2	Halouzka	Druhy zajištění OUI
6.	P	4	Kozak	Systémy velení a řízení
7.	P	2	Hrubý	Úvod do algoritmizace
8.	P L	2 2	Hrubý	Datové typy a datové struktury

9.	P L	2 2	Hrubý	Základní příkazy vyššího programovacího jazyka
10.	L	6	Hrubý	Základní programovací techniky
11.	P	4	Hopjan	Úvod do počítačových sítí, historie, modely síťové architektury.
12.	P	6	Hopjan	Protokoly linkové a síťové vrstvy
13.	P	4	Hopjan	Směrování, protokoly aplikační vrstvy
14.	P L	2 2	Hopjan	Základní konfigurace síťového prostředí v poč. s ohledem na bezpečnost
15.	L	4	Hopjan	Návrh a realizace jednoduchých sítí

Zdroj: vlastní

Struktura předmětu IT, který je určen pro civilní studenty FVT, bude od akademického roku 2023-2024 u témat 1, 7 až 15 shodná s tématy předmětu ITOS. Témata 2 až 6 předmětu IT budou v nezměněné formě tak, jak jsou uvedena v Tabulce 2.

Studentské hodnocení kvality výuky předmětu přináší cenné podněty pro vyučující předmětu, garanta předmětu i garanta studijního programu, nicméně se zdá, že toto hodnocení, které je studenty prováděno až po skončení semestru, je silně ovlivněno dvěma aspekty.

Tím prvním je, že názory studentů výrazně ovlivňuje výuka, kterou studenti absolvují koncem semestru, to je těsně před specifikací pohledu studentů na daný předmět. Druhým aspektem se jeví vliv uzavřenosti, resp. otevřenosti jednotlivých studijních skupin.

Studijní skupiny vojenských studentů jsou zpravidla silně ovlivněné jednou nebo více vůdčími osobnostmi, které se v dané skupině nacházejí. Civilní studenti, kteří netráví většinu dne společně, mají své vlastní pestřejší pohledy na realitu, v níž se vyskytují a mají ji hodnotit. Vojenští studenti a civilní studenti srovnatelných studijních oborů pak posuzují často stejnou problematiku zcela rozdílně.

Závěr

Výchozí znalosti a dovednosti nových studentů FVT UO jsou nejen v oblasti IT velmi rozdílné. Předmět ITOS plní funkci sjednocení znalostí a dovedností v oblasti IT v zinném semestru prvního ročníku studia u vojenských studentů v rámci studijních programů VT-E a VT-S. Předmět IT plní danou funkci u civilních studentů v rámci studijního programu TOB-S.

Autor se domnívá, že studentské hodnocení kvality výuky předmětu by mělo být prováděno navíc alespoň jednou v průběhu semestru. Srovnání získaných dat ke struktuře předmětu a jeho výuce v průběhu semestru a po jeho skončení by mohlo přinést reálnější data nezbytná pro další rozvoj hodnoceného předmětu.

Poděkování

Príspevek vznikl za podpory dílčího záměru rozvoje organizace Kybernetické síly a prostředky (DZRO KYBERSÍLY).

Literatura

Informační systém UO. Akreditace pro NAÚ. Akreditace studijních programů a specializací [online], [2023-07-14]. Dostupné z: <https://apl.unob.cz/Akreditace2017>

Informační systém UO. Studentské hodnocení kvality výuky předmětu [online], [2023-07-14].

Dostupné z: <https://apl.unob.cz/StudentskeHodnoceni>

Kontaktní údaje

Ing. Miroslav Hrubý, CSc.

Univerzita obrany, K-209

Kounicova 65, 66210 Brno

e-mail: miroslav.hruby@unob.cz

ELEARNINGOVÉ ŘEŠENÍ NA TUL ZALOŽENÉ NA MOODLE

ELEARNING SOLUTION AT TUL BASED ON MOODLE

Igor Kopetschke

Abstrakt

V roce 2011 byl na Fakultě mechatroniky, informatiky a mezioborových studií Technické univerzity v Liberci spuštěn elearningový portál založený na Moodle. Jeho popularita úspěšně rostla a v průběhu následujících let se stal hlavním univerzitním nástrojem pro elektronické vzdělávání napříč všemi fakultami. V názvu příspěvku je záměrně zmíněno "založen na Moodle", protože od samého začátku jej vlastními silami rozšiřujeme o nové funkce a možnosti a propojujeme s dalšími zdroji, systémy a službami. Aktuálně nám slouží nejen pro přímou výuku, ale i pro vnitřní kurzy zaměstnanců, výuku v rámci Centra dalšího vzdělávání, distanční přijímací řízení, školení zaměstnanců (BOZP, řidiči aj.), akreditace předmětů a další. Svou nepostradatelnou roli sehrál i v období izolace během pandemie Covid 19.

V rámci tohoto příspěvku bude představeno několik podtémat. V první řadě sada našich rozšíření zaměřené na propojení se studijní agendou (STAG) včetně vytváření kurzů a zápisů do nich, přenášení obsahu mezi verzemi pro jednotlivé akademické roky, vytváření kurzů pro vícejazyčné předměty, propojení s multimediální knihovnou, integrace se službami Google Workspace a další vylepšení. Poté bude pozornost věnována zkušenostmi získanými během pandemie Covid 19 a jak ovlivnily další využívání elearningu až do současnosti. Následující sekce představí řešení pro zaměstnanecká školení včetně propojení výsledků s personální agendou.

Klíčová slova: elearning, Moodle, BOZP, STAG, Covid 19, integrace, programování

Abstract

In 2011, an elearning portal based on Moodle was launched at the Faculty of Mechatronics, Informatics and Interdisciplinary Studies of the Technical University of Liberec. Its popularity grew successfully and over the following years it became the main university tool for e-learning across all faculties. In the title of the paper, "based on Moodle" is deliberately mentioned, because from the very beginning we have been expanding it with new features and possibilities and connecting it with other resources, systems and services. Currently, it is used not only for direct teaching, but also for internal employee courses, teaching within the Continuing Education Centre, distance admissions, employee training (OSH, drivers, etc.), course accreditation and more. It also played an indispensable role during the period of isolation during the Covid 19 pandemic.

Several sub-topics will be introduced in this paper. First and foremost, a set of our extensions aimed at linking to the Student Agenda (STAG), including course creation and enrollment, transferring content between versions for different academic years, creating courses for multilingual courses, linking to the multimedia library, integration with Google Workspace, and other enhancements. Attention will then turn to lessons learned during the Covid 19 pandemic and how they have influenced the continued use of elearning to the present day. The following section will present solutions for employee training, including linking results to the HR agenda.

Keywords: elearning, Moodle, OSH, STAG, Covid 19, integration, programming

Úvod

Výuka s využitím informačních a komunikačních technologií (elearning) je nedílnou součástí portfolia služeb napříč všemi vzdělávacími zařízeními. Tento příspěvek ovšem není zaměřen na doporučené metodiky a didaktické postupy při využívání elearningu. Jsem profesí a svým zaměřením informatik a mj. jsem správce univerzitního elearningového portálu a zároveň vývojář řady jeho modulů a rozšíření. Následující text budu věnovat následujícím tématům.

V rámci prvního tématu stručně shrnu historii a současnost portálu a jeho integraci s dalšími službami univerzity. Následně se zaměřím na některá námi vytvořená rozšíření reflektující specifické požadavky našich uživatelů, případně nahrazující některé standardní Moodle komponenty. Zmíním také rozšíření uživatelské podpory.

Další téma bude věnováno velmi specifickému období výuky na TUL, čímž mám na mysli epidemii Covid 19 a rok a půl trvající absenci prezenční výuky. Zde sehrála existence portálu naprosto zásadní roli při náhradní formě výuky a rád bych zde rozebral některé klady, zápory, slepé uličky a hlavně co nám to přineslo do období, kdy jsme se mohli vrátit do běžného režimu.

Následně představím využívání Moodle na TUL i k jiným účelům, než je běžná výuka, například kurzy určené zaměstnancům, realizaci elearningu pro Centrum dalšího vzdělávání včetně Univerzity 3. věku, přijímací řízení pro cizince distanční formou, vytváření kurzů pro účely akreditace předmětů a jejich přenos do běžné výuky a portál pro školení zaměstnanců v rámci BOZP aj. a jeho propojení s personální agendou.

No a nebudou chybět nějaká ta čísla.

1 ALS není jen nemoc

1.1 Minulost a současnost

V roce 2009 jsme s kolegyní RNDr. Klárou Císařovou, Ph.D. získali finanční podporu na tříletý projekt, jehož účelem bylo vybudovat jednak fakultní elearningové řešení, jednak 2 učebny vybavené technologií pro streaming, ukládání a následné publikování záznamů z přednášek. Díky tomu jsme na Fakultě mechatroniky, informatiky a mezipředmětových studií v roce 2011 s hrdostí představili ALS – Advanced Learning Space. Jednalo se elearningové řešení postavené na jádře Moodle, tehdy ve verzi 3. Záměrně píš *postavené na*, protože od začátku jsme věděli, že budeme Moodle rozšiřovat jednak o integrace s ostatními systémy, jednak o některá funkční rozšíření.

V následujících letech portál expandoval i na další fakulty a aktuálně reprezentuje hlavní elearningové řešení na TUL.

Portál je aktuálně integrován s následujícími službami:

- STAG – studijní agenda
- LDAP + Shibboleth pro ověřování uživatelů a jejich rolí
- Katalog multimediálních záznamů Mediasite
- Turnitin pro prevenci plagiátorství
- Google Workspace, především Drive, Calendar a Meet

1.2 Když dostupné funkce a rozšíření nestačí ...

Jak jsem již zmínil, od začátku bylo jasné, že si z důvodů integrací, absenci některých funkcí anebo díky specifickým požadavkům na TUL nevystačíme se standardní výbavou Moodle včetně existujících rozšíření. Během let tedy vznikla celá řada našich vlastních modulů a některé z nich bych rád představil.

Integrační modul pro propojení se STAG – od počátku bylo základním požadavkem vytvářet kurzy ve verzích pro jednotlivé akademické roky v příslušných kategoriích, vytvářet skupiny podle rozvrhových akcí a zapisovat do nich studenty – a to vše pokud možno s maximální automatizací. První verzi jsme vytvořili v době, kdy ještě ZČÚ neměla k dispozici modul vlastní a používáme jej dodnes. Poslední vylepšení modulu dokáže vytvářet kurzy i v jednotlivých jazykových mutacích, pokud jsou ve STAGu registrované s více vyučovanými jazyky. S tímto úzce souvisí hlavní hierarchie kategorií dle schématu Fakulta – Katedra – Akademický rok. Díky tomuto mechanismu mají uživatelé v daných rolích přístup jen ke kurzům, které jim dle STAGu náleží včetně dané mutace pro AR. Součástí tohoto rozšíření je i naše vlastní Nástěnka.

Rychlá kopírka kurzů – protože od počátku vytváříme nové instance kurzů pro předměty v dalším AR, bylo nutné vyřešit jak jednoduše a rychle přenést obsah starší verze kurzu do nové. Poslední verze Moodle již disponuje funkcí pro rychlý klon kurzu, ale jedná se o relativně novou možnost a dlouhá léta byly jedinými dosti zdoluhavými variantami Import nebo Záloha/Obnovit. Proto jsme si vytvořili vlastní modul pro rychlé přenesení obsahu na dva kliky myši.

Kopírování slovníků – řadu let Moodle trápil nepříjemný neduh. Klasickým scénářem bylo, když kurz obsahoval hlavní a sadu vedlejších slovníků. Při přenosu obsahu kurzu libovolným standardním způsobem docházelo ke ztrátě položek slovníků, případně k porušení vazeb mezi slovníky. Situaci jsme vyřešili vlastním modulem pro kopírování slovníků napříč kurzy.

Hromadné vytváření a zápis do kurzů – standardně nám vytváření kurzů a zápis do nich řídí výše zmíněný modul pro STAG. Nicméně ten do kurzu zapíše pouze ty studenty a pedagogy, kteří jsou u předmětu uvedeni ve STAGu. Museli jsme ovšem řešit situaci (speciálně na FZS), kdy registrovaným učitelem ve STAGu je externí lékař a vytváření kurzů včetně naplnění materiály mají na starosti asistentky na fakultě. Ty ovšem zase nejsou ve STAGu. Pro představu – bývá to přes 350 předmětů v rámci AR, ke kterým je potřeba dozapsat v roli Učitel 6 asistentek. Vytvořili jsme si modul, který dokáže na jedné straně vyfiltrovat seznam předmětů a prostým výběrem uživatelů na straně druhé tyto kurzy v daném AR vytvořit (pokud ještě neexistují) a uživatele do nich v dané roli zapsat. To vše na jeden klik.

Studijní materiál Záznam přednášky – vlastní typ studijního materiálu, který ve spojení s katalogem Mediasite nabídne k vložení do kurzu možnost přehrání pořízeného záznamu přednášky daného předmětu. Zároveň řídí i přístupová práva, takže záznam si může přehrát pouze aktivní účastník kurzu.

Integrace s Google Workspace – kromě propojení se sdílenými materiály na Google Drive máme vlastní modul pro vytváření a vkládání Meet videokonference do kurzu. Jedná se o rozšíření existujícího modulu tak, že studentům zapsaným kurzu či skupině zároveň odešle buď pozvánku, nebo přímo účastníkovi zapíše danou událost do Google Calendar. V období Covid 19 bylo nutné evidovat docházku na videokonference, toho jsme dosáhli analýzou logu videokonference a následným zápisem do tabulek existujícího modulu Attendance.

1.3 Help please ...

Po skončení epidemie Covid 19, kdy na ALS přešla celá univerzita, bylo jasné, že uživatelská podpora ve formě one-man show je neudržitelná, pokud bych měl zároveň ještě učit a pracovat na svých dalších projektech. V roce 2022 byl uveden do života Program na podporu strategického řízení a jeho součástí byla zřízena podpora pro funkci asistenta elearningu. Na každé fakultě máme nyní 2-3 vyškolené pracovníky, kteří poskytují podporu v základních a běžných situacích, což mi odbouralo cca 90% všech dotazů. Tito uživatelé mají speciální roli ve stromu kategorií jejich fakult, vycházející z omezeného archetypu Manažer.

2 Covid 19

2.1 Co nám vzal a co nám dal

Nikdy nezapomenu na černý pátek 13. 3. 2020. Tento den nás Covid vyhnal domů na dlouhé 3 semestry. Doslova ze dne na den bylo potřeba ze stávajícího ALS, který využívala víceméně dobrovolně cca polovina TUL vytvořit stabilní nástroj pro zhruba 5.800 studentů a 620 pedagogů. Rada z nich měla s Moodle buď nulovou nebo pouze povrchní zkušenost, další nemalé procento již stávajících uživatelů využívalo portál pouze pro publikování statických materiálů. Neexistovalo jednotné prostředí pro videokonference. Bylo třeba skokově posílit hardware a úložný prostor. A hlavně všechny proškolit.

To vše v rámci ultimátního příkazu prorektora – chci řešení, chci ho hned a pokud možno zadarmo.

Nebudu zde podrobně popisovat hektické dny a týdny, než se podařilo dosáhnout v rámci možností optimálního řešení. Během té doby se mnohé nepodařilo, něco představovalo slepou uličku, ale na druhé straně se mnohé zdařilo a posunulo nás dále.

Posílení výkonu a stability byl relativně jednoduchý úkol. Servery provozujeme ve virtuálním prostředí, takže navýšení počtu jader, velikost RAM a diskového prostoru bel vyřešen velice rychle. Stávající jednoserverové řešení bylo distribuováno na 3 servery – aplikační, databázový a cache server s Redis.

Akutní bylo vytvoření jednotného videokonferenčního prostředí. Tady jsme absolvovali neúspěšnou cestu formou Jitsi Meet. Byť jsme provozovali vlastní server, tak i přes velice výkonný hardware se nikdy nepodařilo dosáhnout stabilního fungování pro větší počet uživatelů než 30, což absolutně nevyhovovalo. Univerzita tedy pořídila rozšíření stávající licence na Google a vydali jsme se cestou Google Meet. Tato varianta ve spojení nahráváním a našim vlastním výše zmíněným modulem, který mj. dokázal evidovat i docházku nám ke spokojenosti funguje dodnes.

Velice se nám vyplatilo, že máme na TUL Mediasite a katalog nahrávek přednášek. Řadu z nich bylo možné využít jako podpůrné materiály pro distanční studium. Co se ukázalo jako klíčové bylo proškolení uživatelů, především pedagogů, a to nejen v rámci funkcí samotného prostředí Moodle a pravidel publikování materiálů a aktivit. Jelikož se distančně odehrávalo i zkoušení a testování, bylo nutné pedagogy naučit, jak co nejlépe koncipovat testy, aby se pokud možno zabránilo opisování apod. Jak distančně zkoušet s použitím sluchátek, kamery a mikrofonu.

Na všem se ale dají najít i pozitiva. TUL to donutilo používat jednotné technologie a odbourat určitý chaos směrem ke studentům i pedagogům. Rapidně vzrost počet uživatelů ze strany studentů i pedagogů, jakož i počet kurzů, čímž se také sjednotilo výukové prostředí. Masívní vlna online školení a konzultací ze strany správy ALS nejen zjednodušila zapojení nových

uživatelů bez znalosti používání Moodle/ALS, ale rozšířila všem znalosti o správném strukturování a náplni elearningových kurzů a o používání dalších nástrojů Moodle.

2.2 Trochu slíbených čísel

Na závěr připojím pár čísel, která jasně demonstrují nejen očekávaný nárůst uživatelů a kurzů v době epidemie, ale i skutečnost, že elearning již zůstal naprosto dominantním a používaným doplňkem výuky. V následující tabulce uvádím jednak počty studentů na univerzitě a počty aktivně využívajících ALS, jednak totožný přehled pro učitele. Celkový počet učitelů na ALS je vyšší než počet pedagogů proto, že máme jak řadu externích vyučujících mimo akademickou obec, jednak jsou zde i uživatelé, kteří mají v kurzech roli Učitel, nicméně nejsou z titulu úvazku za pedagogy považováni.

Tabulka 1: Počty studentů a pedagogů na TUL a v ALS

	2019	2020	2021	2022	2023
Počet studentů na TUL	7684	7750	7909	7823	7869
Počet studentů na ALS	5745	6956	7011	7526	7697
Počet pedagogů na TUL	637	668	688	684	694
Počet učitelů na ALS	623	746	943	938	954

Další tabulka demonstruje nárůst počtu nově registrovaných a aktivních kurzů v daném roce. Kurzy dle STAG jsou kurzy odpovídající předmětům ve STAG, kurzy mimo STAG mají jiný účel (výuka předmětu napříč fakultami, akreditace, školení, přijímačky atd.)

Tabulka 2: Nárůst počtu nově založených kurzů

	2019	2020	2021	2022	2023
Nové kurzy dle STAG	724	1855	2024	2096	2105
Nové kurzy mimo STAG	242	68	58	52	63
CELKEM	966	1923	2082	2148	2168

Velice významně se také změnila skladba e-opor v rámci kurzů. Zatímco v době před pandemií tvořil počet interaktivních aktivit pouhých 16% z celkového počtu, během pandemie až do teď toto procento osciluje mezi 55-63%.

3 ALS není jen pro výuku

3.1 Školení a výuka pro zaměstnance

ALS nám neslouží jen pro běžnou výuku, ale máme na něm i specializované kurzy pro zaměstnance, pro některé projekty a jejich projektové manažery, pro vedení bakalářských a diplomových prací, pro Vnitřní jazykovou školu aj. Tyto kurzy jsou publikovány v rámci speciálního stromu kategorií a nejsou logicky synchronizovány se STAGem. Zápisy do nich si spravují uživatelé v roli Učitel.

3.2 Centrum dalšího vzdělávání a Univerzita 3. věku

CDV a v jeho rámci i U3V mají k dispozici vlastní instanci Moodle bez propojení se STAGem. Je pro nás velice příjemným překvapením, jak zejména senioři bezproblémově Moodle používají a po proškolení masívně využívali i Google Meet v době pandemie.

3.3 Přijímací řízení a akreditace předmětů

V rámci ALS existují jednak kurzy sloužící pro distanční přijímací řízení studentů ze zahraničí, převážně z Indie na Fakultě strojní. Za 6 let provozování tímto řízením prošlo každý rok cca 50 studentů. Na ALS máme také specializovaný strom kategorií pro účely akreditačního řízení pro fakulty a jejich předměty. K dispozici jsou samozřejmě speciální uživatelské účty pro akreditační komisi a nástroj pro překlopení akreditovaného kurzu do běžné výuky.

3.4 BOZP a další

Další instance Moodle nám slouží pro online periodické školení zaměstnanců, hlavně v rámci BOZP+PO pro zaměstnance a vedoucí pracovníka a pro školení řidičů. Po prostudování povinného portfolia materiálů musí uživatel absolvovat povinný test. Následně mu je vystaven certifikát a údaje o absolvování školení jsou exportovány do personální agendy (VEMA). Za 3 roky se nám touto formou proškolilo přes 1.000 zaměstnanců.

Závěr

Na závěr bych rád poděkoval všem, kdo jste se při čtení dostali až sem. Svůj příspěvek jsem se snažil koncipovat jako multitématický, což bohužel vzhledem k omezené délce textu nedovolilo jít více do hloubky. Budu se snažit toto nahradit při prezentaci, na kterou Vás srdečně zvu.

Kontaktní údaje

Ing. Igor Kopetschke

Technická univerzita v Liberci

Fakulta mechatroniky, informatiky a mezioborových studií

Studentská 1402/2, 461 17 Liberec 1

e-mail: igor.kopetschke@tul.cz

OPERAČNÍ SYSTÉMY – STARÉ TEORIE, NEBO NOVÉ VÝZVY?

OPERATING SYSTEMS – THE OLD BORING THEORY OR A SET OF NEW CHALLENGES?

Lenka Kosková Třísková

Abstrakt

Základní kurz teorie operačních systémů by měl být součástí libovolného vysokoškolského oboru obsahujícího název “Informatika”. Osnova takového kurzu je už řadu let všude podobná: funkce systému, jádro, synchronizace, správa paměti, správa procesů, I/O a správa zařízení, úložiště a systémy souborů, správa identit, zabezpečení a možná ještě uživatelská rozhraní. Mohlo by se zdát, že kurz operačních systémů je suchá teorie podobná výuce matematické analýzy. Může být, ale rozhodně nemusí. Příspěvek proto shrnuje vývoj ve světě OS, důležité trendy a změny vyplývající z vývoje celého oboru, kterými je možné “suchou teorii” oživit a připravit pro studenty výuku ukazující svět operačních systémů v současném rozsahu s ohledem na všechny aktuální trendy (RTOS, open source, virtualizace, vestavná zařízení, nové hardwarové platformy, modulární systémy a tzv. systems of systems, kolaborativní vývoj a další).

Klíčová slova: Operační systém, RTOS, Linux, embedded

Abstract

An introductory course in operating systems theory should be part of any undergraduate course titled "Computer Science". The curriculum of such a course has been similar everywhere for many years: system functions, kernel, synchronization, memory management, process management, I/O and device management, storage and file systems, identity management, security, and maybe even user interfaces. A course in operating systems may look like a dry theory similar to teaching mathematical analysis. It can be, but it certainly does not have to be. Therefore, this paper summarizes the developments in the OS world, significant trends and changes resulting from the evolution of the whole field, which can help to revitalize the "dry theory" and to prepare for students a course showing the world of operating systems in its current scope, taking into account all current trends (RTOS, open source, virtualization, embedded devices, new hardware platforms, modular systems and so-called systems of systems or collaborative development).

Keywords: Operating system, RTOS, Linux, embedded

Úvod

Teorie operačních systémů je častou součástí kurikula vysokoškolských oborů, jež vzdělávají v oboru vývoje SW (FIT ČVUT, FIT VUT, FM TUL, FIT ZCU, PF UPOL). Většina kurzů i dnes odkazuje na klasické teoretické publikace (Tannenbauma, Silberschatz). Ve shodě s těmito publikacemi mají kurzy podobnou osnovu: jádro OS, správa paměti, správa procesů, správa zařízení, úložiště a systémy souborů.

Poslední vydání obou citovaných publikací je cca 10 let staré, OS Linux existuje již téměř 30 let, standard POSIX ještě déle, a tak by se mohlo zdát, že operační systémy představují statický teoretický blok, jenž nevyžaduje aktualizaci a může být vyučován stále stejně.

Není to tak. Mění se architektura počítačů, mění se procesory, HW i úložiště. Logicky se proto mění i operační systémy. V následujícím textu se zaměřuji na posledních cca 10 let – tedy na dobu od posledního vydání výše citovaných knih. Nejprve v rychlém výčtu představím technologické změny a novinky, následně ve vazbě na strukturu kurzu operačních systémů ukážu, co nového a zajímavého lze ve výuce zmínit a ukázat studentům.

1 Technologické změny

1.1 Procesory

Světů serverových procesorů jednoznačně dominuje 64bitová architektura x86, reprezentovaná procesory společností Intel a AMD. Tyto procesory mají dnes typicky desítky jader – v červnu 2023 uvedla společnost AMD na trh procesor se 128 jádry (viz. AMD's New 128-Core), CPU od Intelu mohou mít více než 50 jader.

Vestavné počítače dnes běžně používají vícejádrové ARM, v konfiguraci pojmenované jako SoC (System on Chip) – v jednom čipu je kromě CPU integrována i celá řada dalších HW součástí jako je grafická či zvuková karta, AD/DA převodníky, jednotky MAC, paralelní výpočetní jádra realizovaná jako FPGA, řadiče USB a jiných průmyslových periférií. Roste počet vestavných zařízení využívajících OS Linux a nebo systém typu RTOS. Současně existence těchto zařízení vynutila změnu přístupu k popisu a konfiguraci HW, stejně jako změnu přístupu v implementaci řadičů zařízení.

1.2 Úložiště

K ukládání dat se stále více užívají paměti flash. Na serverech a v desktopových počítačích v provedení jako disky SSD, ve vestavných systémech nejčastěji v podobě eMMC nebo ve formě karet SD. Tyto typy úložišť vyžadují jiné algoritmy pro zápis a čtení dat, a proto se v posledních cca 10 letech objevují nové typy systémů souborů.

Rychlost nejmodernějších disků SSD je díky jejich konstrukci téměř srovnatelná s rychlostí RAM a v budoucnosti ještě poroste. Roste celková kapacita serverových úložišť, velikost a počty ukládaných souborů, což se v serverovém světě odráží ve vývoji nových typů systémů souborů určených pro tato úložiště.

1.3 Vestavná zařízení

Pokrok ve vývoji CPU, dostupnost levných SoC a vedla k obrovskému nárůstu vestavných zařízení využívajících OS Linux (nebo Android). Rozvoj IoT a požadavek, aby i malá vestavná zařízení plně a bezpečně implementovala různé síťové a komunikační protokoly, vede k rozvoji dalších typů operačních systémů určených pro tuto oblast. Řada těchto systémů je vyvíjena jako open source.

Zajímavým trendem ve světě OS jsou investice poskytovatelů cloudových služeb do vývoje OS určených pro malá zařízení pracující jako koncové uzly IoT. V roce 2017 společnost Amazon koupila freeRTOS, Microsoft následoval akvizicí systému ThreadX v roce 2020. Důvod je v obou případech shodný: společnosti chtějí svým zákazníkům nabídnout softwarové řešení pro uzly využívající jejich cloudová řešení pro ukládání dat.

1.4 Virtualizace

V serverové oblasti v posledních letech pokročil vývoj a nasazení virtualizačních metod. Technologie jako Docker a Kubernetes úplně změnily přístup k sestavování serverových farem, k nasazování serverů, jejich správě a zálohování. Zatímco dříve vyžadovalo spuštění a instalace

nového serveru hodiny práce, dnes je možné podstatnou část automatizovat s pomocí technologií pojmenovaných jako Iaas (infrastructure as a code). Tato skutečnost zcela změnila práci administrátorů a přinesla postupy označované jako devops spolu s metodami práce označovanými jako CI/CD.

1.5 Blízká budoucnost

V relativně blízké budoucnosti lze očekávat dvě zásadní změny, na které budou muset operační systémy zareagovat. Rostoucí přístupové rychlosti moderních pamětí RAM a flash povede k možnosti realizovat tzv. in memory computing, neboli možnost ukládat přímo obraz paměti RAM bez nutnosti implementace systémů souborů. Podle strategického materiálu EU ECS SRIA 2023 lze očekávat takové paměti již v období 2027 – 2031. V trochu delším výhledu stejný dokument očekává nástup kvantových počítačů. Lze očekávat, že se moderní OS budou muset adaptovat na situaci, kdy kromě CPU obsluhují ještě kvantový počítač. Je možné, že se bude postupovat podobně jako při obsluze CPU doplněných o FPGA.

2 Vliv změn na obsah výuky

2.1 Jádro a jeho architektura

Klasický výklad o architektuře jádra OS studenty seznámí s konceptem monolitického jádra, mikrojádra a hybridního jádra. Zde bych ráda upozornila na jeden mýtus: totiž tvrzení, že systém s mikrojádrem ztrácí výkon kvůli častému přepínání kontextu a že se proto tyto systémy příliš neužívají. Pro moderní serverové či desktopové počítače nebyl v posledních letech sestaven žádný benchmark, který by podobné tvrzení potvrdil či vyvrátil.

V systémech navrhovaných pro procesory pracující bez ochrany paměti a bez režimů mikrojádra pokles výkonu nezpůsobuje. Mikrokernelové architektury se zde používají – má ji například systém QNX či RIOT. Příkladem odvozené architektury může být ThreadX. Celkově lze výklad o architekturách jádra zpestřit praktickými ukázkami malých OS a vysvětlit, jak jsou ryze teoretické koncepty adaptovány v praxi, a že se kromě mikrokernelů lze potkat s pikokernely (ThreadX) nebo nanokernely (KeyOS).

Linuxové jádro je často udáváno jako příklad monolitické architektury, což je v hrubém náhledu pravda. Při diskusích o architektuře jádra Linuxu je vhodné zmínit, že jádro OS Linux má řádově 10 000 konfiguračních parametrů, jež výrazným způsobem ovlivňují jeho chování a výkon a umožňují začlenit do jádra další typy software, které mění pohled na jádro jako ryze monolitickou architekturu bez reálného času. Jádra Linuxu určená pro výkonné servery se velmi liší od jader určených pro vestavná zařízení.

Od roku 2019 jádro obsahuje eBPF (extended Berkeley Packet Filter). Jedná se o softwarové rozšíření původně vyvíjené pro filtraci síťových paketů, nyní využívané pro ladění výkonu, implementaci bezpečnostních politik nebo řadičů zařízení. eBPF přináší do jádra možnost řídit část procesů v jádře návaznosti na vnější události s jiným stavovým modelem, než mají klasické procesy. Přístup je blízký konceptu služeb v mikrojádře (podrobněji viz. Extended EBPF, A New Type of Software) a lze mluvit o určité „mikrokernelizaci“ jádra Linuxu.

Další významnou změnou jádra OS Linux je existence tzv. „realtime patch“ neboli PREEMT_RT. Před cca 10 lety skupina vývojářů navrhla úpravy (patch) pro linuxové jádro, jež umožní jádru pracovat v režimu reálného času. Dosažení režimu reálného času znamená především vhodný návrh synchronizačních primitiv a časovačů. Autoři úpravy navrhli nové typy zámků a front spolu s úpravou algoritmu pro scheduling. Tyto úpravy jsou postupně začleňovány do hlavního kódu jádra. I bez úprav lze nyní jádro Linuxu konfigurovat tak, aby

dosáhlo práce v režimu reálného času. Původní úpravy jádra zahrnovaly tisíce změn kódu, dnes se jedná o desítky úprav a očekává se, že se realtime patch v blízké budoucnosti stane plnou součástí jádra.

2.2 Správa procesů

Jak je zmíněno výše, existence eBPF přináší zcela nové schéma pro životní cyklus procesu. Z hlediska výuky teorie může být ve výuce opět velmi inspirativní libovolný volně dostupný RTOS – mnoho z nich má poměrně jednoduchý kód, a tak studenti mohou „číst“ schedulingový algoritmus přímo z kódu jádra a mohou jej případně zkoušet sami upravovat.

Například OS Zephyr podporuje několik teoretických konceptů pro scheduling – stačí změnit konfiguraci. Chceme-li studentům vysvětlit principy těchto algoritmů spolu s praktickým měřením výkonu a pozorováním chování systému v zátěži, je Zephyr velmi dobrý kandidát na přípravu zajímavého praktického cvičení (popis algoritmů viz Scheduling – OS Zephyr).

Podobně lze tyto malé systémy velmi dobře využít i na demonstraci práce s vlákny či procesy, na osvojení konceptu a vysvětlení principu, protože implementace podobných funkcí v Linuxu či hledání v jeho zdrojových kódech je často nad síly studentů základního kurzu z operačních systémů.

2.3 Synchronizace

Většina teoretických principů představovaných v základní literatuře (Silberschatz, Tannenbaum), předpokládá, že OS pracuje jen na jednom CPU. Moderní vícejádrové procesory však vyžadují synchronizaci mezi procesy běžícími na různých jádrech. Zajištění atomicity vyžaduje zohlednit paralelní práci jader a paralelní cache. V současné době není známa jedna dobře fungující metoda, a proto moderní kernely často nabízejí různé typy zámků a doprovodných datových struktur – v API najdeme aktivní zámky, zámky s čekáním, zámky realizované lokálně na jednom CPU a další nástroje.

2.4 Správa paměti

Klasický výklad o řízení přístupu k paměti zmíní virtualizaci, stránkování a stránkovací algoritmy, datové struktury a algoritmy vhodné k vyhledávání volného místa v paměti a metody defragmentace paměti. I zde rozhodně platí, že rostoucí velikost dostupné paměti a nutnost vhodné optimalizace vyžaduje hledání dalších metod pro optimální využití paměti.

Uvádím dva aktuální příklady pro OS Linux. K optimalizaci alokace a snížení míry fragmentace paměti jádro využívalo od roku 2006 tzv. SLAB allocator, který paměť dělí do předem připravených objektů nazývaných SLAB. Postupně přibýly ještě metody SLUB (jednodušší správa úseků paměti oproti SLAB) a SLOB (pracuje s malými úseky paměti a je vhodný pro vestavná zařízení). V tomto roce bylo rozhodnuto, že budoucí jádra budou podporovat jen SLUB a SLOB, což znamená významnou změnu této části jádra po dlouhé době a vede k nutnosti upravit kód řadičů využívajících SLAB.

Jiným příkladem postupné proměny správy paměti v OS Linux je algoritmus „memory compaction“, který se zaměřuje na lokální úpravy obsazení paměti tak, aby bylo možné alokovat současně více stránek. Algoritmus byl představen v roce 2010, v roce 2015 začleněn do kernelu a v roce 2020 je v kernelu implementován ve verzi pro-active – k lokálním úpravám přistupuje kernel aktivně bez vyžádání během řízení práce systémem.

2.5 Správa zařízení

Zatímco dříve při vývoji jádra Linuxu bylo nutné zohlednit omezenou množinu CPU a základního HW, současný Linux je portován na tisíce dalších velmi rozmanitých typů vestavných zařízení.

Proto bylo rozhodnuto oddělit popis hardware od kódu řadičů a nejnižší vrstvy HAL, aby začlenění dalšího typu počítače nevyžadovalo úpravy obecných zdrojových kódů. Proto Linux podporuje od roku 2012 tzv. „device trees“. Jedná se o jednoduchou syntaxi, jež stromovou formou popisuje zapojení HW součástí a definuje základní HW parametry (adresy paměťově mapovaných registrů, taktovací frekvence sběrnic, hierarchii sběrnic, hierarchii řadičů přerušení a další).

Linux má nyní upravené API pro práci s device trees, jež umožní číst v kódu řadiče konfigurační informace přímo z device tree. Kromě Linuxu podporuje device trees také zavaděč uBoot nebo operační systém OS Zephyr.

2.6 Ukládání dat

Obecný výklad o systémech souborů se často soustředí na systémy ext3 a ext4, koncept uzlů, ukládání dat a odkazů na data. Jsme však svědky postupného nástupu řady nových systémů souborů. Vestavná zařízení používají k ukládání dat paměti flash. Na fyzické úrovni vyžaduje flash jiné typy operací, než magnetická média – je nutné kontrolovat počet zápisů, ověřovat úspěšnost zápisu, monitorovat vadné bloky a kontrolovat, že jsou všechny buňky média využívány rovnoměrně. Právě proto se v posledních 10 letech objevilo několik nových systémů souborů určených pro flash (JFFS, JFFS2, YAFFS, F2FS, UBIFS). Užívají se zejména ve vestavných aplikacích, OS Linux podporuje většinu z nich. Rostoucí počet vestavných systémů s OS Linux užívá pro systémové disky read-only SquashFS.

Na serverech se setkáme s novými systémy souborů navrženými pro vysokou zátěž, velký počet souborů, velká média a velké soubory. Například distribuce SuSE dnes jako výchozí systém využívá BTRFS.

Výklad o systémech souborů může poukázat na různé způsoby užití FS v cílových zařízeních, můžeme srovnávat jejich vlastnosti a můžeme se studenty diskutovat modelové situace a vhodné FS, případně jim ve cvičení připravit demonstrační úlohu využívající různé typy systémů souborů – například srovnat start systému ze SquashFS a jiného FS

2.7 Start operačního systému

Největší změnou posledních let je definitivní přesun od BIOSu k UEFI. Díky UEFI je možné implementovat tzv. secure boot – ověřovat důvěryhodnost spouštěného systému s pomocí klíčů a kryptografie.

Vestavná zařízení používají dominantně uBoot, open source určený pro zavádění OS Linux. Dokáže nahradit zavaděč 1. fáze (firmware, BIOS) i 2. fáze. I uBoot dokáže s pomocí klíčů ověřit důvěryhodnost zaváděného jádra, dokáže také připravit zařízení ke startu ze sítě se síťovými disky. Protože vestavná zařízení často nejsou během své práce fyzicky dostupná, umí uBoot realizovat před startem update jádra i systému souborů metodou OTA (over the air) a je vybaven řadou dalších velmi užitečných funkcí sloužících k ověření, že cílové zařízení pracuje správně. Zdrojový kód je strukturován podobně jako kód jádra OS Linux. Chceme-li studentům ukázat systém konfigurace a překladu jádra či jiných linuxových programů, je uBoot vhodný kandidát – celkově je mnohem jednodušší, pro studenty představuje nižší bariéru a během jednoho cvičení zvládnou testovat celý proces konfigurace, překladu, úprav i nasazení software.

Závěr

Svět operačních systémů není statický, ale reaguje na rychlý vývoj technologií v celé oblasti IT. V posledních 10 letech došlo díky rostoucímu počtu levných a výkonných SoC k růstu počtu jednodeskových počítačů pracujících s OS Linux nebo systémy RTOS. Vznikly úplně nové operační systémy pro jednodeskové počítače (OS Zephyr, MBed), změnil se způsob, jakým systémy startují i způsob, jakým ukládají data.

Ani Linux není statický monolit – jádro Linuxu užívá nový systém popisu hardware, podporuje řadu nových technologií, vznikly robustní systémy pro vývoj linuxových distribucí (Yocto, Buildroot). Linux dokáže pracovat jako systém reálného času. Vznikají oborová sdružení snažící se nabízet referenční řešení pro průmyslové oblasti (AutomotiveGradeLinux, Linux4Space).

A vše zmíněné se dá krásně ukázat studentům a pro téměř všechno se dají připravit zajímavá cvičení, aby studenti cítili kontakt teorie s praxí.

Literatura

AMD's New 128-Core CPU Elevates Efficiency of Cloud Data Centers [online][2023-08-28].

Dostupné z: <https://www.electronicdesign.com/technologies/embedded/article/21268245/electronic-design-compact-128core-cpu-elevates-efficiency-of-cloud-data-centers>

Extended BPF, A New Type of Software [online][2023-08-27]. Dostupné z

https://www.brendangregg.com/Slides/UM2019_BPF_a_new_type_of_software/#25

ECS – Strategic Research And Innovation Agenda [online][2023-08-27] Dostupné z:

<https://ecssria.eu/introduction>

Scheduling – OS Zephyr [online][2023-08-28]. Dostupné z:

<https://docs.zephyrproject.org/latest/kernel/services/scheduling/index.html>

Silberschatz, A., Galvin, P.B., Gagne, G.: Operating System Concepts, 10. vydání, John Wiley & Sons, 2018.

Tanenbaum, A. S. "Modern Operating Systems (2nd Edition)". Prentice Hall, 2001

Kontaktní údaje

Ing Lenka Kosková Trísková, Ph. D.

Ústav nových technologií a aplikované informatiky

Technická univerzita v Liberci

Studentská 1402/2

461 17 Liberec

e-mail: lenka.koskova.triskova@tul.cz

DIDAKTICKÉ METODY A EXPERIMENTÁLNÍ PŘÍSTUPY VE VÝUCE PROGRAMOVÁNÍ

DIDACTIC METHODS AND EXPERIMENTAL APPROACHES IN TEACHING PROGRAMMING

Vojtěch Lapuník

Abstrakt

Programování je komplexní a specifická dovednost, která i na vysokých školách představuje pro studenty velkou výzvu. Jakožto student Ph.D. programu na Elektrotechnické fakultě Západočeské univerzity v Plzni mi bylo umožněno vést cvičení kurzů Základy programování pro elektrotechniku a Modelování v MATLABu a Simulinku. Oba kurzy jsou určeny pro první ročník bakalářského studia. V tomto článku budou nastíněny netradiční, ale účinné formy výuky a komunikace, které byly aplikovány. Cílem bylo kromě prostého nalezení didakticky vhodných nástrojů pro výuku programování také podpora kreativity a logického myšlení, podnětí motivace ke studiu a v neposlední řadě vybudování pozitivních vztahů mezi studenty, lektorem a univerzitou.

Klíčová slova: didaktické metody, výuka programování, vzdělání v informačních technologiích

Abstract

Even in universities, programming is a challenging skill that poses significant challenges for students. As a student of the PhD programme at the Faculty of Electrical Engineering at the University of West Bohemia in Pilsen, I had the privilege to teach the courses Fundamentals of Programming for Electrical Engineering and Modelling in MATLAB and Simulink. Both courses are designed for the first year of undergraduate studies. In this article, unconventional but effective methods of teaching and communicating skills in programming are presented. In addition to finding didactically appropriate tools for teaching programming, the goal was to encourage creativity and logical thinking, stimulate motivation to study, and last but not least, build positive relationships between the students, the instructor, and the university.

Keywords: didactic methods, teaching programming, information technology education

Úvod

Vzdělávání je doménou, kde se projevuje značný nesoulad mezi rychlou inovací moderních technologií (internet, umělá inteligence) a pomalou transformací vzdělávacích systémů, metodik a obsahů. Tento rozpor tvoří tlak na vyučující, kteří se musí vyrovnat s novými výzvami a požadavky na jejich roli a kompetence. Vyučující nemohou být pouhým médiem pro přenos informací, ale musí se přetvořit ve facilitátory, mentory, kouče a inspirátory. Měli by být schopni stimulovat studenty k rozvoji jejich digitálních znalostí, mediální gramotnosti, kreativity, spolupráce, kritického myšlení a řešení problémů. Nové technologie by měly být využity jako nástroj pro učení, nikoli být považovány za hrozbu, konkurenci či dokonce překážku. V oboru výuky informačních technologií a programování je toto téma obzvláště aktuální, protože požadavky na vyučující se mění s ještě větší frekvencí, než je tomu v jiných oblastech. Tento článek se věnuje možným změnám ve výukových strategiích a v přístupu k vzdělávání obecně, především v kontextu rozvoje vyučovaných technologií.

1 Současný stav vyučování programovacích předmětů a motivace práce

Rád bych úvodem zdůraznil, že všechny názory a myšlenky, které zde budou prezentovány, jsou moje osobní a zcela subjektivní. Jsou založeny na osobně nashromážděných datech se statisticky nevýznamnou skupinou studentů, proto zde žádná data uváděna nejsou. Cílem příspěvku rozhodně není zpochybňovat a kritizovat práci druhých, nýbrž se vším respektem a pokorou podnítit zamyšlení nad daným tématem a pokusit se o změnu pohledu na výuku nejen v oblasti programování. Jako bývalý student a nyní vyšší vysokoškolský lektor programovacích předmětů jsem se setkal s mnoha problémy a nedostatky, které brzdí efektivní a motivující učení této důležité dovednosti. Základní problémy budou nastíněny podle typického vysokoškolského schématu na témata týkající se přednášek a témata týkající se cvičení.

Přednáška, jako spojitý monolog s cílem předat strohé informace je obecně jeden z nejhorších způsobů, jak studenty učit (Bloom, 1956). Při tomto způsobu učení nejsou zapojeny vyšší kognitivní procesy a studenti tak nemohou přednášené informace v hlavě dlouhodobě uchovat (Kohoutek, 2008). U programování je navíc potíž, že informace na sebe úzce navazují a pouhé částečné nepochopení části přednášky tak většinou vede k nevyhnutelnému nepochopení i zbytku přednášky. Cvičení, by oproti přednáškám, měla obsahovat praktické zaměření a studenti by si měli vyzkoušet procvičit nabrané vědomosti z přednášek. Avšak jak bylo zmíněno výše, vědomosti z přednášek mohou být často prakticky zanedbatelné (nikoliv vinou studentů), a cvičení tak často ztrácí svůj účel. Ne výjimečně jsem byl svědkem situace, kdy vyučující studentům pouze promítá svůj kód a doplňuje ho komentářem. Studenti jsou nuceni kód opisovat, aniž by nad ním měli čas přemýšlet, a kvůli tomu se naučí pouze povrchně reprodukovat kód, nikoli tvořivě a kriticky programovat.

Naprostou zásadní výzvou ve výuce musí být zodpovězení otázky, co mají studenti po úspěšném absolvování předmětu umět, nebo ještě lépe, čeho mají být schopni. Není možné používat stejné metody a techniky, pokud se výuka zaměřuje na syntaxi a konstrukci programovacího jazyka, jako když je hlavním cílem předmětu rozvíjet studenty v oblasti tvorby algoritmů a programátorského myšlení. Tyto dvě oblasti vyžadují odlišné úrovně abstrakce, kreativity a logiky. Bohužel se lze často setkat s vyučujícími, kteří učí jedno a od studentů potom očekávají druhé. Obecně platí, že studenti na hodinách se musí věnovat tomu, co od nich očekáváme u zkoušek a zápočtů. Pokud je zkouška ve formě psaní delšího kódu, musí být drtivá většina cvičení věnována samostatné práci studentů. Pokud je zkouška ústního charakteru ve formě dialogu, musí být dialog se studenty veden také na hodinách. Nelze čekat, že pasivní poslouchání informací a přepisování kódu vytvoří studentovi stabilní základy k získání analytického myšlení, nebo schopnosti vést o tématu odbornou debatu.

2 Didaktika ve výuce programování

Jako řada jiných technicky zaměřených oblastí i výuka programování se potýká s problémem, že pouze malá část vyučujících má formální pedagogické vzdělání, které by jim umožnilo efektivně předávat své znalosti a dovednosti. Nelze předpokládat, že vysoká odborná způsobilost vyučujícího je zárukou jeho pedagogické kompetence. Možná i z toho důvodu spousta vyučujících zapomíná, že programování je praktická dovednost, která vyžaduje aktivní trénink a opakované cvičení, a nelze ji osvojit pouze pasivním sledováním nebo posloucháním. Přitom nesmí být zapomenuto, že kritickým faktorem v tréninku jakékoliv činnosti je čas. Ten je u každého studenta odlišný a vyučující by to měl reflektovat ve výuce. Zaměříme-li se na situaci, kdy předmět slouží pro studenty jako úvod do programování, může být osvojení této dovednosti obzvláště zdlouhavé. Navíc je nutno brát v úvahu, že v takovém předmětu se budou vyskytovat jak studenti s naprostou nulovými zkušenostmi, tak studenti pokročilejší.

Vyučující tak nesmí propadnout falešnému dojmu, že studenti jsou dál, než ve skutečnosti jsou. I za předpokladu, že výuka probíhá správně a studenti na cvičeních skutečně trénují svoje algoritmické schopnosti, tak i přes to je čas strávený programováním na cvičeních většinou nedostatečný. Tím se odkrývá první zásadní, ale často opomíjená úloha vyučujícího, a sice koučování a dodávání motivace studentům. Pokud student během cvičení nenapiše ani řádek kódu, ale i přes to, díky nadšení z vyučovací hodiny stráví celý večer programováním, tak lze hodinu považovat za velmi úspěšnou. V minulosti mohl přednášející sehrávat klíčovou roli především v předávání informací, protože zdroje informací byly omezené a dostat se k nim nebylo snadné. V dnešní době však nedává smysl, aby přednášející reprodukoval základní principy, které již jsou několikanásobně a velmi kvalitně zpracovány a vysvětleny na internetu. Místo toho by měl demonstrovat praktický význam a užitek přednášené látky, motivovat studenty k tomu, aby si uvědomili všechny důsledky, které může studium dané problematiky nabízet. Nadšený student je jeden z nejdůležitějších cílů, kterého by měl vyučující dosáhnout. Čisté předání informací potom může suplovat pomocí vhodně zpracovaných skript. Ve skriptech by ale měli být obsaženy pouze informace nutné pro zvládnutí předmětu. Přemíra informací snižuje přehlednost a studenty od jejich využívání odrazuje.

Další důležitou součástí učitelství práce v dnešní době je tvorba prostředí vhodného pro učení. Ze studie (Čejková, 2018) vyplývá, že lidská stránka vyučujícího má zcela zásadní vliv na oblíbenost předmětů a oblíbenost předmětů zase úzce koreluje s množstvím a kvalitou studenty získaných informací. Ani vysoká expertiza vyučujícího nedokáže nahradit stav, kdy nebere ohledy na potřeby a emoce studentů, nebo kdy má slabě rozvinuté komunikační schopnosti. Ač každý student má jiné představy o ideální výuce (Chetty et al., 2019), tak za každým oblíbeným předmětem nutně stojí studentův oblíbený vyučující. Typické studijní prostředí, kde studenti sedí v pozoru, nesmějí spolu mluvit, nesmějí používat svá mobilní zařízení a jakékoliv prohrěšení vede k trestu, je z didaktického hlediska zcela nežádoucí. Přitom o výhodnosti pozitivní výuky pojednává např. (Barnes, 2002). V dobře nastaveném prostředí nebojuje student proti vyučujícímu o zápočet, ale vyučující je studentovi partnerem v dosažení společného cíle, a sice naučení studenta nutné vědomosti a dovednosti. V tomto kontextu nedává žádný smysl budovat si u studentů formální autoritu, nýbrž je žádoucí vytvářet vzájemný respekt a budovat vzájemnou důvěru.

Přístupem, který dokáže vytvořit příjemné prostředí, vyvolat nadšení a zaujmout studenty je zaměření se na zábavné aspekty předmětu. Pokud je docházka na přednášku v řádech nižších desítek procent, měl by si vyučující přiznat nezajímavost svého pojetí předmětu a vymyslet pojetí takové, které studenty zaujmout dokáže. Každý student samozřejmě odpovídá za svoji důslednost při studiu, ale pokud je tento jev opakující se rok, co rok na většině studentů, je jednoduše chyba na straně vyučujícího. Zavedení povinné docházky pak jen vytváří nežádoucí prostředí čili nežádoucí stav pro proces učení viz odstavec výše. Zábavnost může být vyvolána interaktivním pojetím a dynamikou předmětu, nebo např. způsobem přednesu a využitím humoru.

3 Konkrétní postupy, metody a strategie

V minulé kapitole byla nastíněna role vyučujícího a dílčí části jeho práce, která se s informační dostupností stále proměňuje. Následující sekce bude věnována konkrétním metodám a postupům, pro naplnění výše nastíněných principů. Jedná se pouze o malý vzorek všech možných metod, avšak může sloužit jako příklad, pro lepší pochopení problematiky. Text bude rozdělen na metody obecné, a týkající se pouze cvičení.

3.1 Obecné

Klíčové je uvědomění, že každé normě zavedené na hodině předchází určitý precedent. Pokud je žádoucí v hodinách změnit konkrétní aspekt (např. chuť studentů interagovat), tak je nutné nalézt precedent, který výskyt normy umožní. Např. pokud na první hodině je položena otázka, na kterou nikdo ze studentů nezná odpověď, tak odpověď vyučující nedostane. Kvůli tomu vzniká norma, že na otázky není potřeba odpovídat. Oproti tomu, pokud bude zvolena otázka otevřená a ideálně velmi jednoduchá, a navíc budu vyučující čekat tak dlouho, jak je třeba, než odpověď skutečně obdrží, tak vytváří normu, že na jeho otázky musí vždy přijít odpověď.

Pokud je s precedenty vhodně nakládáno, lze si hodinu podle preferencí vědomě přizpůsobit. Velkou problematikou je obecně nejistota studentů. A to jak v oblasti zkoušek a zápočtů, v oblasti náplně a užitečnosti předmětu, nebo např. ve stylu výuky. Pro studenty je významně jednodušší připravovat se kvalitně na předmět, pokud přesně vědí, jak bude vypadat zápočtový test, popř. ústní zkoušení. Vhodnou metodou je poskytnutí zkušebního testu včetně řešení i vysvětlení a ukázat simulaci ústní zkoušky s dobrovolníkem. Dalšími nástroji pro upevnění jistoty je např. představení náplně předmětu, jeho zasazení do kontextu studovaného oboru a vysvětlení jeho významu pro budoucí rozvoj studenta. Poskytnutí skript, které jsou úzce zaměřeny na informace vyžadované u zkoušky či zápočtu také působí na studenty velmi uklidňujícím efektem a dopomáhá oblibě předmětu. Práci se studenty je možné zahájit již před začátkem semestru pomocí elektronických prostředků. Možné je zaslání emailu s osnovou předmětu, skripty, nebo jen pouhé pozdravení a několik informací o předmětu. Studenti vždy poznají míru připravenosti vyučujícího. Z toho důvodu je vhodné připravit si na hodiny rámcové scénáře, a to i včetně otázek, které budou studentů položeny, a jejich možných odpovědí.

Ve druhé kapitole tohoto článku byl kladen důraz na vyučujícího v roli inspirátora, mentora a kouče. K motivování studentů lze použít různé metody. U programovacích předmětů se mi osvědčila silná motivační trojice věda, hry a peníze. Nehledě na to, zda se nám to líbí, či nikoliv, velká část studentů jde studovat s vidinou lepšího zaměstnání s vysokým finančním ohodnocením. Proto je v předmětech zaměřených na programování užitečné prezentovat tabulku s průměrnými platy v ČR podle typu zaměstnání, kde si oblast informačních technologií vede obzvláště dobře. Od jednoho bývalého studenta mi bylo sděleno, že pro něj programování představuje značnou výzvu, avšak s perspektivou finanční stability v budoucnosti je motivován a věnuje této činnosti maximální možný čas. Motivace s vidinou vývoje vlastních her je zase zaměřena na ty, kteří rádi počítačové hry hrají, což na technicky zaměřených školách bývá obvykle i více než 50 % studentů. Poslední motivační nástroj, který využívám je věda. Ta naopak zaujme snaživé studenty, kteří svou budoucnost vidí v oblasti výzkumu a vývoje už před nástupem na vysokou školu. Jako prostředky používám projekty, na kterých právě pracuji, nebo aktuální novinky ze světové vědy. U motivace je navíc vhodné, aby byla studentům často připomínána, např. prostřednictvím narážek, vtipů, nebo příběhů z vlastního okolí během celého semestru.

3.2 Cvičení

Budování příjemného prostředí (viz druhá kapitola), vzhledem k počtu účastníků, lze výrazně jednodušeji tvořit na cvičeních než přednáškách. Získání důvěry a vytvoření přátelského vztahu mezi vyučujícím a studenty přináší do hodin velkou přidanou hodnotu. Vhodná metoda může být tykání a oslovování křestními jmény již od první hodiny. Studenti získají pocit rovnosti a důležitosti. Nepsané didaktické pravidlo říká, že jméno každého studenta by mělo na každé hodině alespoň jednou zaznít a každý student by měl v hodině alespoň jednou před ostatními mluvit. Pomůckou k zapamatování jmen v co nejkratším čase může být tahák s rozložením lavic

v učebně, kam si vyučující jména napíše. Do taháku lze vpisovat také stručné osobní informace o studentech, které mu následně pomáhají dělat výuku více personalizovanou.

Každého studenta potěší, že si vyučující pamatuje jeho oblíbenou kapelu, nebo sportovní aktivitu.

Jeden z nejužitečnějších nástrojů, pro účely poznání a spřátelení je tzv. check-in okénko na začátku každé hodiny. Jedná se o krátkou rozpravu, typicky složenou z několika jednoduchých otázek. V prvních hodinách je vhodné tento časový prostor věnovat poznávacím otázkám typu: „Jak se jmenuješ?“, „Z jaké střední školy přicházíš?“, „Co od předmětu očekáváš?“, až se postupem semestru dostat k dotazům situačním jako např. „Jaká nejuvtipnější situace se ti v minulém týdnu přihodila?“. Další typy otázek v tomto okénku mohou být směřovány na pocity a náladu. Pro vyučujícího je užitečné vědět, zda jsou studenti v dobrém rozmaru, nebo jestli jdou z nepříjemného zápočtového testu a na vtipy nejsou naladěni. Tento check-in zabere několik málo minut, ale okamžitě vytvoří normu interaktivního prostředí. Úvod hodiny může být věnován také určité aktuální události nebo zajímavosti z oboru, které studenty zaujme a pomůže jim rozmluvit se. Po takovém úvodu jsou studenti výrazně lépe naladěni na výuku a pro vyučujícího je práce s nimi jednodušší. Ač to na první pohled nemusí být zřejmé, jedná se o velmi výhodně investované minuty z vyučovací hodiny. Méně důležitou, ale také užitečnou metodou může být zahrnutí tzv. check-out okénka na konci hodin. Opět se jedná o rozpravu se studenty, ale nyní s cílem zjistit co se studenti skutečně naučili a čemu naopak neporozuměli, popř. zda je hodina bavila nebo co by si v rámci vyučování představovali jinak. Čím přátelštější atmosféra v hodinách panuje, tím konstruktivnější zpětné vazby jsou studenti schopni. Pro získání skutečně upřímné zpětné vazby je možné pozvat studenty na konzultaci u piva, ideálně v době před testem, nebo těsně po něm.

Pokud má být výuka interaktivní a aktivizační, tak vhodné užívání a práce s otázkami hraje klíčovou roli. Celá hodina by měla být postavena okolo zásadních otázek, které by měly být zodpovězeny, nebo okolo zásadních úkolů, které by měly být splněny. Cílem tedy není informaci studentům říct, ale položit správnou otázku a snažit se studenty dovést k odpovědi. V té chvíli dochází k vysokému zapojení studentů do problematiky a učí se nad ní přemýšlet. Pokud na položený dotaz nikdo neodpovídá, vyučující se nesmí spokojit s tím, že správnou odpověď studentům vyzradí. Musí otázku dělit, zjednodušovat, popř. nechat studenty vybírat mezi několika možnými odpovědi a následně je nechat vysvětlovat, proč se zrovna k dané odpovědi přiklonili. Skvělým nástrojem jsou vyšší kognitivní otevřené dotazy, které je potřeba promyslet a neexistuje na ně jedna definitivně správná odpověď. Každou otázku lze doplňovat navazujícími otázkami typu „Proč si myslíš, že to tak je?“, „Co by se stalo, kdyby to bylo jinak?“, nebo „Jakými kroky si k této odpovědi dospěl?“. V tomto kontextu otázka není prostředek k získání informací, ale didaktickým nástrojem pro vyvolání debaty, ideálně mezi studenty navzájem. Studentům také pomáhá, pokud u složitějších otázek mají čas na rozmyšlenou. Stačí jen několik vteřin, nebo minut a odpovědi studentů bývají výrazně promyšlenější. Během této doby je vhodné, aby studenti své odpovědi psali na papír. Při následném procházení odpovědí stačí studentům odpověď pouze přečíst z papíru a lépe tak ucelí myšlenky. V žádném případě by se však student po své odpovědi neměl cítit hloupě, aby nebyl od aktivního přístupu v hodině odrazen. K tomu může dopomoci poděkování za odpověď, pochvala za jiný pohled na problematiku, či rozvinutí myšlenky studenta, i přes to, že přímo necílí k očekávané odpovědi. Závěrem je vždy vhodné uvést kontext otázky a vysvětlit proč byla položena.

Závěr

Hlavním cílem článku bylo otevřít diskusi a prezentovat poznatky týkající se pedagogické činnosti v technicky orientovaných předmětech v roce 2023. I když tento článek pokrývá jen

část metod a přístupů, které by mohly být užitečné pro budoucí rozvoj učitelské profese, tak nabízí rámec pro konceptualizaci a možné směry vývoje. Role vyučujícího jako prostředníka pro přenos informací se zdá být v současnosti stále více zbytná, což ale zvyšuje důraz na jiné aspekty pedagogické práce. Schopnost zaujmout, zabavit, motivovat, nadchnout a vést, jsou v tomto textu shledány za zásadní. Nahrazení autoritativního vztahu mezi studentem a vyučujícím za vztah přátelský může být rozhodujícím faktorem v budoucím vzdělávání nových generací. Věřím, že tyto poznatky a poznání mohou napomoci k zlepšení kvality a výsledků výuky v technicky zaměřených oborech.

Poděkování

Tento článek vznikl za podpory projektu SGS-2021-011: Rozvoj technik snižování řádu systému v elektrotechnických aplikacích.

Literatura

- Barnes, R. (2002). *Positive teaching, positive learning*, Abingdon-on-Thames: Routledge.
- Bloom, B. S., Englehart, M. D., Furst, E. J., Hill, W. H., & Krathwohl, D. R. (1956). *Taxonomy of Educational Objectives, Handbook 1: Cognitive Domain*. Boston: Addison Wesley.
- Čejková, I. (2018). Žákovské hodnocení výuky na gymnáziu: výpovědi sexty A. *Pedagogická orientace*, Roč. 28 (č. 3), Brno: Masaryk University, 408-434.
- Chetty, N. D. S., Handayani, L., Sahabudin, N. A., Ali, Z., Hamzah, N., Rahman, N. S. A., & Kasim, S. (2019). Learning Styles and Teaching Styles Determine Students' Academic Performances. *International Journal of Evaluation and Research in Education*, Roč. 8 (č. 4), Yogyakarta: Institute of Advanced Engineering and Science, 610-615.
- Kohoutek, R. (2008). Kognitivní vývoj dětí a školní vzdělávání. *Pedagogická orientace*, Roč. 18 (č. 3), Brno: Masaryk University, 3-22.

Kontaktní údaje

Ing. Vojtěch Lapuník
Katedra elektrotechniky a počítačového modelování
Fakulta elektrotechnická, Západočeská univerzita v Plzni
Univerzitní 26, 301 00 Plzeň
lapunik@fel.zcu.cz

ZÁKULISÍ VÝUKY PŘEDMĚTU DIGITÁLNÍ TRANSFORMACE PODNIKU – PODNIKOVÁ ARCHITEKTURA (MEZI TEORIÍ A PRAXÍ)

BEHIND THE SCENES OF THE DIGITAL TRANSFORMATION OF THE ENTERPRISE – ENTERPRISE ARCHITECTURE (BETWEEN THEORY AND PRACTICE)

Martin Lukáš

Abstrakt

Od roku 2021 je vyučován předmět Digitální transformace podniku. Reaguje na současnou poptávku po mezioborových znalostech z oblasti ICT, systémových přístupů, manažerských metod, teorie řízení a procesní analýzy v oblasti soukromoprávních (např. Průmysl 4.0), ale i veřejnoprávních subjektů (eGovernment). Předmět je volitelný v několika výukových koridorech. Jeho cílem je seznámit posluchače s principy digitální transformace a podnikové architektury, modelech a nástrojích (TOGAF, ArchiMate, EA Sparx, Archi), logickém členění a grafickém zápisu. Procvičovaná teorie i praxe je formou případové studie na konkrétním subjektu vybraném studenty. Výukové bloky jsou rozděleny dle tematických úseků, které využije student při řešení klíčových oblastí digitální transformace.

Klíčová slova: digitální transformace, enterprise architektura, modely, procesy, informační systémy, aplikace, infrastruktura, TOGAF, ArchiMate

Abstract

From 2021, the subject Digital Transformation of the Enterprise is taught. It responds to the current demand for interdisciplinary knowledge in the field of ICT, system approaches, management methods, management theory and process analysis in the field of private (e.g. Industry 4.0), but also public entities (eGovernment). The course is optional in several learning corridors. It aims to introduce students to the principles of digital transformation and enterprise architecture, models and tools (TOGAF, ArchiMate, EA Sparx, Archi), logical structuring and graphical notation. The theory and practice is practiced in the form of a case study on a specific entity selected by the students. The teaching blocks are divided according to thematic sections that the student will use to address key areas of digital transformation.

Keywords: digital transformation, enterprise architecture, models, processes, information systems, applications, infrastructure, TOGAF, ArchiMate

Úvod

Předmět vznikl jako reakce na potřebu propojit poznatky ICT, manažerských metod, systémové analýzy s dalšími obory, které jsou potřebné při řízení podniků, firem, subjektů a organizací (ekonomice, informační bezpečnost). Je vyučován v druhém ročníku navazujícího studia, jeho délka je semestr. Do výuky je zapojen garant a vyučující. Předmět je koncipován jako tři hodinový vyučovací blok, ve kterém dle úvahy vyučujícího se kombinuje tradiční přednášky s cvičením.

1 Cíl předmětu

Cílem předmětu je seznámit posluchače se základními principy digitální transformace a podnikové architektury (Enterprise Architecture). Nejčastěji se podniková architektura používá pro popis, definici a následné řízení rozvoje organizace jako celku (tzn. ve všech úrovních a vrstvách – motivace, strategie, business, aplikace a informační systémy, technologická a komunikační infrastruktura. Tento – celostní přístup - je pro podnik důležitý z pohledu udržitelnosti a aktuálnosti procesů, které jsou informačními systémy zajištěny. Zároveň je důležitý při řízení změn, ať už dílčích nebo komplexních, jako je např. výměna informačního systému.

Díky zavedení podnikové architektury, např. pomocí architektonického rámce TOGAF, získá organizace přehled o stávajícím stavu (AS-IS) v jednotlivých vrstvách. Lze tak efektivněji navrhnout budoucí stav (TO-BE) a řídit změny s využitím existujících nebo nově vytvořených technologických celků.

Předmět poskytuje základní představu o přístupech, modelech a nástrojích digitální transformace (TOGAF, ArchiMate, BPMN, EA Sparx, BizDesign, Archi), logickém členění a grafickém zápisu jednotlivých vrstev organizace. Přednášky a cvičení jsou rozděleny dle tematických úseků, které využije student při řešení klíčových oblastí digitální transformace. V rámci cvičení je na příkladech procvičen komponentový přístup k podnikové architektuře a transformaci procesů. Základem cvičení jsou modelové příklady organizace nebo podniku, které si studenti sami vyberou, zakreslení stávajícího stavu pomocí nástroje Archi a návrh budoucího stavu včetně způsobu digitální transformace.

2 Témata přednášek

Předmět je tematicky rozčleněn takto:

1. Historické souvislosti – vývoj tématu podnikové architektury
2. Podniková architektura a digitální transformace
3. Role ICT v podnikové architektuře a digitální transformaci
4. Přístupy, modely a nástroje pro digitální transformaci (TOGAF, ArchiMate, BPMN)
5. Notace ArchiMate a její logické vrstvy
6. Notace ArchiMate a elementy jednotlivých vrstev
7. Role podnikové architektury a projektového řízení digitální transformace
8. Klasifikace a prioritizace procesů a ICT komponenty pro digitální transformaci
9. Přínosy digitální transformace – příprava projektu, výzvy, rizika, problémy, řízení
10. Využití agilního a waterfall přístupu v digitální transformaci
11. Kontinuální navrhování a zlepšování digitální transformace (DevOps)
12. Pracovní role a tým digitální transformace – profily rolí a osobnostní předpoklady

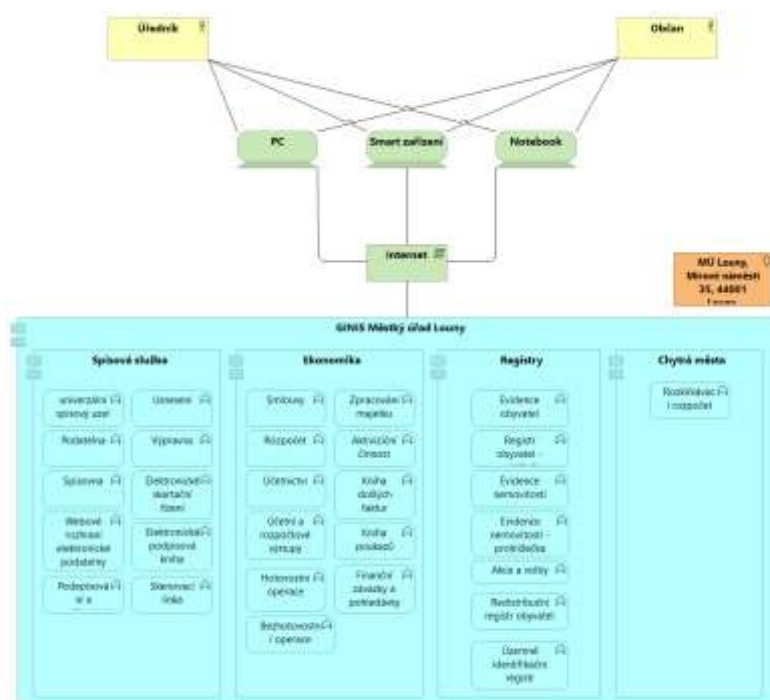
3 Shrnutí dosavadních poznatků z výuky

Rozdělení do týmů je pro studenty zajímavé, komplexita předmětu a případové studie se tak rozloží mezi členy týmu (, kterých je max 5).

Studenti bakalářského studia využívají freeware nástroj Archi <https://www.archimatetool.com/>

Studenti navazujícího magisterského studia používají nástroj Enterprise architect SPARX <https://sparxsystems.com/> .

Práce jsou mnohdy pestré, více či méně formalizované, vždy záleží na osobnosti vedoucího studenta.



Obrázek 1: Ukázka studentského výsledku použití grafického jazyka ArchiMate v případové studii – městský úřad. Zdroj: studentská práce (2022)

4 Rozvoj předmětu a jeho budoucnost

Předpokládáme, že zaměříme na převody modelů mezi nástrojem Archi a Enterprise architect SPARX s cílem ukázat platformní nezávislost a brát nástroj jako prostředek, nikoliv cíl.

Závěr

Dílčí témata z tohoto předmětu lze vhodně vkládat, a tak kombinovat s látkou jiných inženýrských předmětů (např. Informační systémy, Elektronické podnikání a obchod). Tím se vytvoří znalostní přechod a propojení poznatků, které studenti mají v oblibě. Dochází k posílení interdisciplinarity i transdisciplinarity ve výukovém procesu.

Literatura

LUKÁŠ, Martin. *Digitální transformace podniku, přednášky a cvičení předmětu, podniku (prezentace přednášek, cvičení, modelové příklady k řešení, řešené vzorové příklady)*, Moodle (kód předmětu PAENK, PAEN2), ČZU v Praze, PEF, KIT, 2021, . [on-line] <https://moodle.czu.cz/>

Článek v časopise:

LUKÁŠ, Martin. Digitální transformace veřejné správy, časopis *Obec & finance*, Ročník 27, výtisk 5/2022, str. 44-45, Triada a. s., ISSN 1211-4189

LUKÁŠ, Martin. Jak mohou municipality řídit rozvoj IT pomocí EA (1, 2 a 3), časopis *Moderní obec*, výtisk 7/2021, 9/2021, 11/2021, str. 48-49, 40-41 a 26-27, Profi Press, ISSN: 1211-0507

LUKÁŠ, Martin. Applying the Enterprise architecture describing the processes of gathering data and reporting Czech national environmental indicators within EU, *International Journal of Public Administration, Management and Economic Development IJPAMED*, 2021, VOL. 6, NO. 2 ISSN: 2533-4077, pg. 66-77, Faculty of Administration and Economic Studies in Uherské Hradiště, Jagiellonian College in Toruń, Studentské náměstí 1531, 686 01 Uherské Hradiště, Czech Republic, <https://www.fves.eu/ijpamed.html>

Článek v časopise (online verze časopisu):

LUKÁŠ, Martin, ULMAN, Miloš, Lost in Translation: Enterprise Architecture in e-Government Projects., EGOV-CeDEM-ePart 2020, *Proceedings of Ongoing Research, Practitioners, Workshops, Posters, and Projects of the International Conference EGOV-CeDEM-ePart 2020*, 31 August-2 September 2020 Linköping University, Sweden (Online), <http://dgsoc.org/egov-2020/>

LUKÁŠ, Martin, BUREŠOVÁ, Tereza, ULMAN, Miloš, A Novel Approach To Business Process Analysis And Optimization Based On Enterprise Architecture: A Case Study Of Local Public Administration In The Czech Republic, proceedings of *IDIMT-2022 Conference (Digitalization of Society, Business and Management in a Pandemic)*, p.59 – p.66, 30th Interdisciplinary Information Management Talks, September 7th – 9th, 2022, ISBN 978-3-99113-758-0, University of Economics and Business, Prague, Czech Republic

Elektronický zdroj bez uvedení autorů (online):

THE OPEN GROUP. ArchiMate® 3. 1 Specification, Van Haren Publishing, 2019. ProQuest Ebook Central, <https://ebookcentral-proquest-com.infozdroje.czu.cz/lib/czup/detail.action?docID=6191452>

HARRISON, Rachel. TOGAF® Business Architecture Level 1 Study Guide. Van Haren, 2018. ProQuest Ebook Central, <https://ebookcentral-proquest-com.infozdroje.czu.cz/lib/czup/detail.action?docID=6191460>

Kontaktní údaje

Ing. Martin Lukáš, Ph.D.

email: lukas@pef.czu.cz

Katedra informačních technologií

Provozně ekonomická fakulta

Česká zemědělská univerzita v Praze

Kamýcká 129, 165 00 Praha – Suchbátka

tel: +420 731 137 393

VÝUKA PŘEDMĚTU STROJOVÉ UČENÍ

COURSE AND LECTURING APPROACHES IN MACHINE LEARNING

Jan Mittner

Abstrakt

Příspěvek se zabývá výzvami při výuce strojového učení na Vysoké škole polytechnické Jihlava a Vysoké škole ekonomické v Praze. Zaměřuje se na efektivní metody, které kombinují teoretický základ a praktické úkoly s využitím technologie Jupyter Notebook. Příspěvek zdůrazňuje nutnost průběžné adaptace kurzu a poskytuje přehled zpětné vazby od studentů k použitým metodám.

Klíčová slova: strojové učení, umělá inteligence, jupyter notebook, python, neuronové sítě

Abstract

The paper discusses challenges in teaching machine learning at the College of Polytechnics Jihlava and Prague University of Economics and Business. It concentrates on effective methods that combine theoretical foundations with practical tasks using Jupyter Notebook technology. The paper emphasizes the need for continuous course adaptation and provides an overview of student feedback on the methods used.

Keywords: machine learning, artificial intelligence, jupyter notebook, python, neural networks

Úvod

Strojové učení je v současné době jedním z nejrychleji se rozvíjejících oborů informačních technologií. Ať už ve vědeckém výzkumu, v průmyslu nebo v každodenním životě, jeho dopad je nezpochybnitelný (Aggarwal et al. 2022).

Strojové učení je podoborem umělé inteligence, který se zabývá vývojem algoritmů schopných učit se z dat. Na rozdíl od tradičního programování, kde jsou pravidla explicitně specifikována pro dané úkoly, strojové učení umožňuje počítačům naučit se pravidla z dat a adaptovat je na nové situace (Géron, 2019).

Tento příspěvek se zaměřuje na způsoby, jakými se strojové učení vyučuje na Vysoké škole polytechnické Jihlava a Vysoké škole ekonomické v Praze. Hlavním cílem je představit různé praktiky použité při výuce a zhodnotit zpětnou vazbu od studentů, do jaké míry se tyto postupy osvědčily v praxi.

1 Výzvy ve výuce strojového učení

Strojové učení je multidisciplinární obor, který kombinuje prvky statistiky, matematiky, informatiky a oborově specifických znalostí (Géron, 2019). S tímto širokým spektrem témat přicházejí také výzvy, které je třeba při výuce překonat.

1.1 Komplexnost a široká škála disciplín

Strojové učení není monolitický obor. Pokrývá širokou škálu tématických oblastí od základních algoritmů, jako jsou lineární regrese a klasifikace, až po složitější témata, jako jsou konvoluční neuronové sítě nebo transformery. Je výzvou vytvořit kurz, který je současně srozumitelný, komplexní a aktuální.

1.2 Prerekvizity

Aby mohli studenti úspěšně porozumět strojovému učení, potřebují určitou znalost v oblastech jako je lineární algebra, pravděpodobnost, statistika a programování. Ne všichni studenti mají tento základ, což může vést k nerovnostem v rychlosti chápání látky, a ztížit tak efektivní výuku.

1.3 Udržení aktuálnosti kurzu v rychle se vyvíjejícím oboru

Strojové učení je dynamický obor s rychlým vývojem. Co bylo relevantní dva tři roky zpět, nemusí být dnes. Toto představuje problém v kontextu akademických kurzů, které obvykle nejsou tak agilní. Udržet kurz aktuální a relevantní je náročná úloha, která vyžaduje pravidelné aktualizace teoretických i praktických podkladů a sylabů.

2 Pedagogické přístupy

Výuka strojového učení vyžaduje intenzivní kombinaci teorie a praxe. Namísto zahlcení studentů složitými matematickými vzorci jednotlivých modelů se kurz soustředí na poskytnutí silného, ale srozumitelného teoretického základu. Cílem je, aby studenti porozuměli základním principům, aniž by se ztratili v matematických detailech.

Pro lepší zapojení a pochopení teorie studenti okamžitě po teoretickém výkladu aplikují nově nabyté znalosti na reálné úlohy v předpřipravených pracovních sešitech v technologii Jupyter Notebook. Tyto sešity kombinují text a spustitelný kód v Pythonu společně s jeho výstupy, a to i grafickými, jak je vidět na obr. č. 1. Tento krok zajišťuje, že teorie je rychle převedena do praxe, což podporuje efektivní učení.



Obrázek 1: Uživatelské rozhraní sešitů Jupyter Notebook. Zdroj: autor

Kromě připravených praktických úloh v Jupyter Notebooku studenti řeší i doplňující úlohy, které jim umožní hlubší pochopení látky. Tyto úlohy jsou obvykle formou domácích úkolů,

kteře rozšiřují vzorovou úlohu a za jejichž vypracování mohou získat bonusové body, které se jim mohou hodit v další fázi kurzu. Tímto způsobem jsou studenti motivováni k neustálému zlepšování.

Studenti mají možnost si v průběhu semestru vybrat vlastní projekt, který budou zpracovávat formou semestrální práce. Pokud průběžně získají dostatek bonusových bodů, mohou si vybrat méně pracnou variantu projektu, což slouží jako další forma motivace.

Kurz je zakončen teoretickým testem, který průřezově ověřuje znalosti studentů ze všech témat.

3 Obsah a struktura kurzu

Kurzy na VŠPJ a VŠE v Praze mají podobný obsah i strukturu. Na VŠE v Praze je kurz vypsán jako 4IT574 – Praktické aplikace strojového učení. Na VŠPJ pak jako PASU – Praktické aplikace strojového učení.

Základní koncept využívající Jupyter Notebooky vychází z kurzu Hands-On Machine Learning (Géron, 2019), v oblasti neuronových sítí témata vycházejí z (Chollet, 2019). Náměty na jednodušší úlohy strojového učení jsou zpracované v (Coelho, Richert, 2013) a (Segaran, 2007). Aktualizace kurzu o současná témata zejména v oblasti generativní umělé inteligence vycházejí z autorovy praktické zkušenosti v předmětné oblasti.

Současná struktura kurzu:

- Úvod do strojového učení
 - hlavní pojmy, historie, současnost, algoritmy, typy úloh, výzvy a problémy,
 - prakticky: lineární regrese - predikce HDP podle země.
- Základní praktiky strojového učení
 - doporučený pracovní postup při řešení úloh strojového učení,
 - prakticky: projekt strojového učení od začátku do konce – predikce ceny nemovitosti.
- Klasifikace entit
 - binární a multiclass klasifikace,
 - prakticky: rozpoznávání psaných číslic.
- Trénování modelů
 - metoda Gradient Descent,
 - prakticky: Titanic dataset – binární klasifikace nad strukturovanými daty.
- Pokročilé metody klasifikace
 - Support Vector Machines, Rozhodovací stromy, Random Forest,
 - prakticky: SPAM filtr – binární klasifikace nad nestrukturovanými daty.
- Učení bez učitele
 - shlukování, redukce dimenzionality, detekce anomalií,
 - prakticky: systém pro doporučování filmu ke zhlédnutí.

- Neuronové sítě a Deep Learning
 - principy a klíčové pojmy, framework Keras,
 - prakticky: binární a multiclass klasifikace, regrese.
- Deep Learning pro počítačové vidění
 - konvoluční neuronové sítě,
 - prakticky: rozpoznávání objektů v obraze, rozpoznávání obličejů.
- Deep Learning pro texty a sekvenční data
 - word embeddings, předtrénované modely,
 - prakticky: rekurentní neuronové sítě – predikce teploty z meteo údajů.
- Generativní Deep Learning
 - variační autoenkodéry, GAN, transformery,
 - prakticky: generování textu, přenos stylu obrazů, generování obrazů.

4 Hodnocení kurzu a zpětná vazba

Zpětná vazba od studentů formou anket a dotazníků na konci semestru poskytuje cenné informace o tom, co funguje a co je potřeba vylepšit.

Na VŠE v Praze se anket zúčastnilo 20 respondentů s těmito souhrnnými výsledky¹:

- Obsah a náročnost předmětu: 88/100.
- Překryv s jinými předměty: 97/100.
- Celková spokojenost s předmětem a náměty na zlepšení: 98/100.
- Jak na mě působil vyučující: 98/100.

Odpovědi na otevřenou otázku „Co se mi na předmětu líbí“:

- „Náplň předmětu, domácí úkoly na procvičení a přípravu ke zkoušce, praktické ukázky.“
- „Jupyter notebooky, praktické ukázky.“
- „Skvěle odhadnutý poměr mezi teoretickými poznatky a praktickým užitím. Zároveň je skvělé, že jsme si mohli vyzkoušet funkční a praktická řešení, která jsme si během hodiny vyvinuli. Skvělé je i to, že jsme zpracovávali reálné úlohy, ne učebnicové a uměle vykonstruované problémy. Zdůraznění výhod a nevýhod různých přístupů, best practices.“
- „Přínos do praxe.“
- „Bohužel musím říci, že toto je jeden z mála předmětů na VŠE, který je skutečně záživný, užitečný, dobře odvykládaný s dobrými materiály. Pan doktor Mittner měl vše perfektně připraveno, jeho výklad byl srozumitelný a bylo vidět, že rozumí i složité

¹ Hodnocení 1-4 (1 nejhorší, 4 nejlepší, převedeno na škálu 0-100)

teorii za výkladem. Možnost si interaktivně zkusit připravená cvičení byla skvělá. Co bylo také skvělé, byly průběžné domácí úkoly. Vlastně myslím, že to bylo snad to nejpodstatnější na tomto předmětu - na nich se člověk nejvíc naučil. Sestavování semestrální práce už mohlo být "snazší".

- „Připravenost přednášek a souborů s kódem, Přístup vyučujícího, volitelné úkoly, praktické využití znalostí“

Odpovědi na otevřenou otázku „Co a jak by se na výuce mohlo zlepšit (cvičení i přednášky), abych se více naučil/a“:

- „Chápu, že se předmět snaží obsáhnout co největší spektrum využití ML, ale přišlo mi, že tempo bylo velmi svižné a od úplně základních věcí jsme se dostali k velmi komplexním dost rychle. V druhé polovině semestru jsem byl často ztracen.“

Na VŠPJ byl kurz poprvé vyučován v LS22/23, proto jsou zatím k dispozici výsledky jedné ankety se třemi respondenty:

Otázka: Ohodnoťte studijní materiály, které jsou vám poskytnuty prostřednictvím e- learningu. Jsou studijní materiály v e-learningu dostačující k tomu, abyste zvládali průběžně samostudium? Odpověď: 3x ano.

Závěr

Výuka strojového učení na Vysoké škole polytechnické Jihlava a Vysoké škole ekonomické v Praze ukazuje, že kombinace teorie a praxe ve vzdělávacím procesu přináší pozitivní výsledky. Přes existující výzvy, jako je rychlý vývoj oboru, multidisciplinární charakter látky a variabilní základní znalosti studentů, se ukázalo, že aktuální podoba kurzu je studenty pozitivně vnímaná. Zpětná vazba od studentů potvrzuje spokojenost s kurzy a poukazuje na úspěšnou integraci teoretických znalostí s praktickými dovednostmi.

Literatura

- Aggarwal, K. et al. (2022). [online] Has the Future Started? The Current Growth of Artificial Intelligence, Machine Learning, and Deep Learning. , Iraqi Journal for Computer Science and Mathematics, Vol. 3No. 1 (2022) p. 115-123. Dostupné online z <https://www.iasj.net/iasj/download/cefbfd60eb11898a>.
- Coelho, L. P.; Richert, W. (2013) Building machine learning systems with Python. Birmingham: Packt Publishing. ISBN 978-1-78216-140-0.
- Géron, A. (2019) Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition. O'Reilly Media, Inc. ISBN 9781492032649.
- Chollet, F. (2019) Deep Learning v jazyku Python. Knihovny Keras, TensorFlow. Grada Publishing, a.s. ISBN 978-80-247-3100-1.
- Segaran, T. (2007) Programming collective intelligence: building smart web 2.0 applications. Beijing: O'Reilly Media. ISBN 0-596-52932-5.

Kontaktní údaje

Ing. Jan Mittner, Ph.D.

e-mail: mittner@vspj.cz

Katedra technických studií

Vysoká škola polytechnická Jihlava

Tolstého 16, 58601 Jihlava

APLIKACE PRO AUTOMATIZOVANÉ HODNOCENÍ ÚKOLŮ

APPLICATION FOR AUTOMATED ASSESSMENT OF PROGRAMMING TASKS

Marek Musil

Abstrakt

Hodnocení úkolů je nedílnou součástí výuky programování. Tato aktivita je časově náročná a počet studentů ji úměrně zvyšuje. Otázka složitosti hodnocení úkolů je diskutovaným tématem, hodnocení úkolů se stalo objektem zájmu vyučujících a vědců. Problém časové náročnosti provádění hodnocení zaznamenal značnou kritickou pozornost. V průběhu času již bylo představeno několik studií. Tento příspěvek shrnuje problém časové složitosti hodnocení úkolů a představuje aplikaci pro automatizované hodnocení úkolů používanou ve výuce programování na VŠPJ. Aplikace je primárně určena pro jazyk C a C++ a je vyvinuta s využitím frameworku cUnit pro implementaci testů.

Klíčová slova: automatické vyhodnocení C kódu, cUnit, hodnocení úkolů, vzdělávání, výuka programování

Abstract

Assessment of assignments is an integral part of teaching programming. This activity is time-consuming and the number of students increases it proportionally. The question of the complexity of task evaluation is a debated topic, task evaluation has become an object of interest for teachers and scientists. The issue of the time-consuming nature of the evaluation has received considerable critical attention. Several studies have already been presented over time. This paper summarizes the problem of the time complexity of task evaluation and presents an application for automated task evaluation used in teaching programming at VŠPJ. The application is primarily intended for C and C++ and is developed using the cUnit framework for test implementation.

Keywords: automated assessment of assignments, cUnit, task assessment, education, programming

Úvod

Hodnocení úkolů je nedílnou součástí výuky programování. Časová složitost (režije) pro ohodnocení odevzdaných řešení je klíčovým problémem. Hodnocení úkolů je objektem zájmu a problém časové složitosti zaznamenal značnou kritickou pozornost ze strany učitelů a vědců. Otázka složitosti hodnocení je diskutovaným tématem ve vědecké komunitě. V průběhu let bylo prezentováno mnoho studií. Aktuálnost tohoto tématu potvrzují nedávné studie, uveďme pouze některé (Varga and Kristóf Fekete, 2023; Mekterović et al., 2023; Muzikář and Zemánek, 2022).

Ve výuce programování, časová režie provedení ohodnocení úkolu a počet studentů úměrně zvyšující tuto složitost nás vedl k zamyšlení nad způsobem realizace hodnocení úkolů. Tento

příspěvek sumarizuje dílčí klíčové kroky hodnocení a představuje aplikaci automatizovaného hodnocení úkolů. Tato aplikace byla vyvinuta v rámci bakalářské práce (Makedonenko, 2023) a již byla nasazena.

Ověření praktických dovedností je jedním z aspektů hodnocení ve výuce programování. V této věci ve svém příspěvku zaměřeném na kontrolu linuxových skriptů zaujímají postoj autoři Muzikář and Zemánek (2022). *"Zatímco ověření teoretických znalostí je technicky poměrně jednoduché (ústní či písemná zkouška), ověření praktických znalostí je složitější a naráží minimálně na dva problémy."* (Muzikář and Zemánek, 2022) Jedním problémem je ověření správnosti chování skriptu a druhý se týká možnosti realizace skriptu.

Časová režije hraje klíčovou roli. Důležitým aspektem je také objektivnost hodnocení. V nedávné době, hodnocení domácích úkolů se stalo předmětem výzkumu a vědci projevíli zájem o představení implementovaných aplikací. Vhodným nástrojem je framework pro jednotkové testy, resp. například, cUnit pro jazyk C nebo jUnit pro jazyk Java. Nástroj pro jednotkové testy jUnit je představen, například, v bakalářské práci (Sládek, 2021).

Východiska a omezení diskutovaná v předchozích studiích: Jako ilustrující příklad zde uvádíme dvě nedávné studie (Varga and Kristóf Fekete, 2023) a (Mekterović et al., 2023). V první uvedené studii, autoři Varga and Kristóf Fekete se odkazují na řadu předcházejících povětšinou nedávných studií (Douce et al., 2005; Combéfis, 2022; Ihantola et al., 2010; Cipriano et al., 2022; Mohamed et al., 2013; McCabe, 1976). Uvedme několik citací: *"Hodnocení úkolů klade značné nároky na čas a další zdroje instruktora"* (Douce et al., 2005) a *"podpora automatického vyhodnocování kódu je zásadní pro úsporu času a usnadnění práce instruktorů."* (Combéfis, 2022) *"To se obvykle provádí automatizací kompilace a provádění kódu studenta v omezeném prostředí a ověřením jeho správnosti pomocí porovnávání výstupu, testování jednotek nebo jiných technik."* (Ihantola et al., 2010)

Autoři Mekterović et al. (2023) se zabývají pokročilými problémy hodnocení jako jsou plagiáty a statická analýza kódu. Říkají zajímavý úvod do problému. *"The first automated assessment of student programs was reported more than 60 years ago, but this topic remains relevant and highly topical among computer science researchers and teachers. In the last decade, several factors have contributed to the popularity of this approach, such as the development of massive online courses, where large numbers of students can hardly be assessed manually, the COVID-19 pandemic with a strong online presence and physical relocation of students, and the ever-increasing shortage of personnel in the field CS. Modern Automated Programming Assessment Systems (APASs) are nowadays implemented as web applications."* (Mekterović et al., 2023)

5 Definování problému a postup řešení

Studenti odevzdávají řešení úkolů do moodle. Aktuální postup hodnocení úkolů je tvořen následujícími manuálními kroky. Komprimovaný soubor stažený z moodle obsahuje komprimované řešení každého odevzdání. Každé odevzdání se musí extrahovat a projekt otevřít ve vývojovém prostředí. Následuje zkontrolování zdrojového kódu a zadání hodnocení s komentářem do moodle. Prodlevy vznikající při otevírání projektu a při ukládání hodnocení ve webové aplikaci moodle zvyšují značně složitost ohodnocení dílčího úkolu.

Ve výše popsaném problému identifikujeme tři části: extrakce souboru s odevzdaným řešením, procházení jednotlivých odevzdání a ověření správnosti řešení, zápis hodnocení do moodle. V rámci bakalářské práce (Makedonenko, 2023) byla vytvořena aplikace pro hodnocení úkolů v jazyce C a C++. Pro testování funkčnosti je použit framework cUnit.

6 Výsledky a diskuze

Byla implementována aplikace pro automatizované hodnocení úkolů v předmětu programování. Jako vstup se definuje testovací scénář a odkaz na soubor s úkoly (ručně stažený z moodle). Každý testovací případ má ve skriptu označeno bodové hodnocení. Hodnocení probíhá způsobem, testovací případ se vykonal úspěšně/neúspěšně. Do textového souboru se zapisuje celkové bodové hodnocení. Tento soubor se pak použije pro ruční import hodnocení do moodle.

```
// @grade 1.5
TEST(binary_plus, complex_plus_double) {
  //SECTION("complex + double")
  {
    //SECTION("#1")
    {
      complex c1(1.22, 3.44);
      auto sum = c1 + 2.22;
      ASSERT_EQ(sum, complex(3.4, 3.4)) << "@complex + double,#1;";
    }
    //SECTION("#2")
    {
      complex c1(-22, 22);
      auto sum = c1 + 11;
      ASSERT_EQ(sum, complex(-11, 22)) << "@complex + double,#2;";
    }
    //SECTION("#3")
    {
      complex c1(-7.44, 12);
      auto sum = c1 + 7.44;
      ASSERT_EQ(sum, complex(0, 12)) << "@complex + double,#3;";
    }
  }
}
```

Obrázek 1: Definice testovacího případu s bodovým hodnocením.

Zdroj: Makedonenko (2023)

Testovací scénář se definuje jednoduše. Aplikace funguje a poskytne spravedlivé výsledky. Soubor s výsledky obsahuje kromě bodového hodnocení i informace o dílčích chybách. Tyto informace slouží jako komentář pro studenta. Aplikace byla ověřena na dvou odevzdání. V rámci ladění byly identifikovány další případy hodnocení: špatný formát odevzdání, nekompilovatelný kód a nedodržení programování proti danému rozhraní. Byla ověřena funkčnost aplikace. Je funkční podle předpokladu. Nevýhodou toto řešení je programování proti rozhraní. V rámci úkolu je zadána předloha a student pouze doplňuje kód do předpřipravených testovacích metod. Nemůže tak být zadán úkol, kde se očekává ověření návrhu řešení.

ID	Přijmení	Jméno	Známka	Komentář
99990	Student1	Student1	20	
99992	Student2	Student1	0	There is no tiny-05.cpp file in student's folder
99998	Student3	Student1	96.5	complex + complex, #i;
99994	Student4	Student1	0	Compilation error
99995	Student5	Student1	0	complex + complex, #i;complex + double, #i;double + complex, #i;com
99996	Student6	Student1	0	File is not a .cpp file

Name	Studentname4
Surname	Student4
ID	999994

```

File: Student4 Studentname4_999994_assignsubmission_file_cv1.zip
Unzipping - done
Copying files - done
Compiling - done
Error: Compilation error

```

Obrázek 2: Výsledky hodnocení a výsledek hodnocení jednoho chybového řešení.

Zdroj: Makedonenko (2023)

Dalším rozšířením může být implementace funkce prohlížení výsledků, poukázání na krajní chybové případy a možnost korekce bodového hodnocení na základě ručního posouzení chybových nedostatků.

Poděkování

Děkuji studentovi Oleksandr Makedonenko za realizaci bakalářské práce.

Závěr

S využitím nástroje cUnit byla vytvořena funkční aplikace pro automatizované hodnocení úkolů v jazyce C a C++. Aplikace splňuje účel a výrazně řeší problém složitosti hodnocení úkolů programování. Aplikace je snadno přenositelná na jiný programovací jazyk (např. Java).

Literatura

Cipriano, B.P., Fachada, N., Alves, P., 2022. Drop project: An automatic assessment tool for programming assignments. *SoftwareX* 18, 101079. URL: <https://www.sciencedirect.com/science/article/pii/S2352711022000577>, doi:<https://doi.org/10.1016/j.softx.2022.101079>.

Combéfis, S., 2022. Automated code assessment for education: Review, classification and perspectives on techniques and tools. *Software* 1, 3–30. URL: <https://www.mdpi.com/2674-113X/1/1/2>, doi:10.3390/software1010002.

Douce, C., Livingstone, D., Orwell, J., 2005. Automatic test-based assessment of programming: A review. *J. Educ. Resour. Comput.* 5, 4–es. URL: <https://doi.org/10.1145/1163405.1163409>, doi:10.1145/1163405.1163409.

Ihantola, P., Ahoniemi, T., Karavirta, V., Seppälä, O., 2010. Review of recent systems for automatic assessment of programming assignments, in: *Proceedings of the 10th Koli calling international conference on computing education research*, pp. 86–93.

Makedonenko, O., 2023. Application for automated assessment of programming tasks. *Bakalářská práce*. Vysoká škola polytechnická Jihlava.

McCabe, T., 1976. A complexity measure. *IEEE Transactions on Software Engineering SE-2*, 308–320. doi:10.1109/TSE.1976.233837.

Mekterović, I., Brkić, L., Horvat, M., 2023. Scaling automated programming assessment systems. *Electronics* 12. URL: <https://www.mdpi.com/2079-9292/12/4/942>, doi:10.3390/electronics12040942.

Mohamed, N., Sulaiman, R.F.R., Endut, W.R.W., 2013. The use of cyclomatic complexity metrics in programming performance's assessment. *Procedia - Social and Behavioral Sciences* 90, 497–503. URL: <https://www.sciencedirect.com/science/article/pii/S1877042813020077>, doi:<https://doi.org/10.1016/j.sbspro.2013.07.119>. 6th International Conference on University Learning and Teaching (InCULT 2012).

Muzikář, Z., Zemánek, P., 2022. Technická systémová podpora výuky operačních systémů, in: škola polytechnická Jihlava, V. (Ed.), *Informatika 2022 - Sborník příspěvků z konference*, pp. 59–62.

Sládek, M., 2021. Jednotkové testy, význam a využití v praxi. Bakalářská práce. Vysoká škola polytechnická Jihlava.

Varga, E.B., Kristóf Fekete, A., 2023. Applications for automatic c code assessment, in: 2023 24th International Carpathian Control Conference (ICCC), pp. 21–26. doi:10.1109/ICCC57093.2023.10178987.

Kontaktní údaje

Ing. Marek Musil

Katedra technických studií, Vysoká škola polytechnická Jihlava

Tolstého 16, 586 01 Jihlava

e-mail: marek.musil@vspj.cz

PROBLEMATIKA E-LEARNINGU A DIGITÁLNÍCH ZDROJŮ VE VÝUCE ČEŠTINY PRO CIZINCE

ISSUES OF E-LEARNING AND DIGITAL RESOURCES IN TEACHING CZECH TO FOREIGNERS

Dean Posavec-Malok

Abstrakt

Tento příspěvek se zaměřuje na problematiku digitálních výukových zdrojů ve výuce češtiny pro cizince. V průběhu textu jsou diskutována klíčová témata, která zahrnují porovnání klasických a digitálních učebních materiálů, využití umělé inteligence a přístup učitelů k digitálním zdrojům. Dále se zabýváme otázkou, jakým způsobem vyučující organizují digitální zdroje, jak je hodnotí a jaké metodické postupy využívají při interakci se studentem. Cílem příspěvku je poskytnout přehledné a komplexní informace o využívání digitálních výukových zdrojů a jejich vlivu na výuku češtiny pro cizince.

Klíčová slova: čeština pro cizince, digitální výukové zdroje, klasické výukové zdroje, (umělá inteligence)

Abstract

This paper focuses on the issue of digital learning resources in teaching Czech to foreigners. In the text, key themes are discussed, which include a comparison of traditional and digital teaching materials, the use of artificial intelligence and teachers' access to digital resources. We also address the question of how teachers organize digital resources, how they evaluate them, and what methodological approaches they use when interacting with students. The aim of the paper is to provide clear and comprehensive information on the use of digital teaching resources and their impact on teaching Czech to foreigners.

Keywords: Czech for foreigners, digital teaching resources, classical teaching resources, (artificial intelligence)

Úvod

Čeština pro cizince je obor, který se v posledních letech těší nemalému zájmu, a to nejen v České republice (viz více v Posavec-Malok 2023: 3-11). Již v roce 2000 psal Hrdlička (2000: 7), že „lze bez nadsázky konstatovat, že se český jazyk těšil a těší nemalému zájmu jinojazyčných mluvčích a v rámci jazyků slovanských se řadil a řadí k těm žádanějším, vyhledávanějším,“ čímž se potvrzuje naše tvrzení. Obor čeština pro cizince se časem též musel přizpůsobovat době. Jako příklad lze uvést koronavirovou pandemii, během níž se veškerá výuka, nejen tedy výuka češtiny pro cizince, přesunula do digitálního světa. Vznikaly její různé podoby a nástroje, jež ji měly nejen usnadňovat, ale i zpestřovat. Po skončení pandemie nicméně spousta lektorů zůstala u on-line výuky a své materiály tomu přizpůsobila.

Cílem tohoto článku je poukázat, do jaké míry po skončení pandemie používají lektori digitální, respektive klasické učební materiály. Dále se snaží zmapovat výukové aplikace, jež se při výuce

používají a v neposlední řadě se v článku zmíním o aplikacích založených na umělé inteligenci, které lektoři využívají.

1 Výukové zdroje

Výukové zdroje neboli materiály se vyskytují v mnoha podobách. Šebesta (2017: 166) je definuje jako „veškeré učební pomůcky a technické vybavení, které napomáhají edukačnímu procesu a které může využít učitel či žák při vyučování/osvojování znalostí a dovedností v daném jazyce.“ Pro potřeby tohoto článku jsme se výukové zdroje rozhodli rozdělit do dvou kategorií – výukové zdroje klasické a výukové zdroje digitální. Jako jakýsi „mezistupeň“ mezi oběma typy materiálů lze vymezit autentické materiály, jež se ale již objevují i v digitálních podobách (viz 1.2). Jedná se tedy o „všechny materiály, které nebyly primárně vytvořeny k didaktickému účelu a které se používají ve výuce cizích jazyků jako materiály výukové.“ (Šebesta 2017: 17)

1.1 Výukové zdroje klasické

Výukovými zdroji klasickými se rozumí všechny zdroje využívané před dobou technologickou, jde tedy o zdroje, které nejsou závislé na moderních technologiích, nebo na nich závisí minimálně. Jedná se především o klasické tištěné zdroje, jež jsou používány lektory při výuce i studenty při osvojování cizího jazyka. Zde lze zařadit mimo jiné tištěné učebnice a pracovní sešity, tištěné pracovní listy, autentické materiály (např. novinové články, krásná literatura) apod. Pro účely osvojování češtiny jako cizího jazyka prostřednictvím krásné literatury vznikly adaptované prózy známých českých autorů (Karel Čapek, Jan Neruda, Michal Viewegh).² Čeština pro cizince disponuje dle našeho názoru největším množstvím klasických výukových zdrojů. Situace se ale začíná měnit.

1.2 Výukové zdroje digitální

Výukové zdroje digitální jsou úzce spjaty s termínem e-learning. S vývojem různých technologií docházelo i k vývoji nových nástrojů, jež se využívaly a využívají i dodnes ve výuce nejen cizích jazyků. Eger (2020: 8) e-learning definuje jako „vzdělávací proces, který je spojen s počítači a informačními a komunikačními technologiemi.“ Digitální výukové zdroje lze tedy definovat jako všechny typy vzdělávacích prostředků, jež jsou závislé na moderních technologiích, především na internetu (multimediální zdroje³, výukové aplikace), nebo různých softwarech/hardwarech (zde máme na mysli zdroje, které nejsou závislé na internetu a lektoři/studenti je mohou využívat i bez přístupu k internetu). Mezi digitální zdroje můžeme zařadit již zmíněné multimediální zdroje a výukové aplikace, ale také interaktivní e-učebnice⁴, interaktivní pracovní listy. Mezi digitální výukové zdroje lze zařadit i nástroje umělé inteligence. Tyto nástroje mohou usnadnit práci lektorům a pomoci studentům při zvládnutí vyučované látky. Při použití nástrojů umělé inteligence vzniká řada etických otázek (především u velkých jazykových modelů, např. ChatGPT), protože zvládnou odvést práci za studenta. Výběr vhodného výukového materiálu je zcela v kompetencích lektorů, jež při přímém kontaktu se studentem mohou některé materiály doporučit, či nedoporučit.

² Plný výčet těchto publikací je k dispozici zde: <https://www.czechstepbystep.cz/ucebnice-kategorie/adaptovana-proza>. [poslední přístup: 26.07.2023]

³ Multimediálními zdroji se zde rozumí především filmy, seriály, podcasty, různé typy videí, ale také hudba (písň, muzikály apod.).

⁴ Rozlišujeme učebnice a e-učebnice, protože již dnes vznikají učebnice, jež jsou v digitální podobě a studentům nabízí různé typy interaktivity, např. jedním kliknutím se studentovi přehraje poslechové cvičení nebo jakékoli jiné doplňující cvičení v rámci e-učebnice. Příklady e-učebnic včetně návodů na jejich použití jsou uvedeny zde: <https://www.czechstepbystep.cz/e-verze>. [Poslední přístup: 26.07.2023]

V další části článku se budeme zabývat proběhnuvším výzkumem mezi lektory češtiny pro cizince.

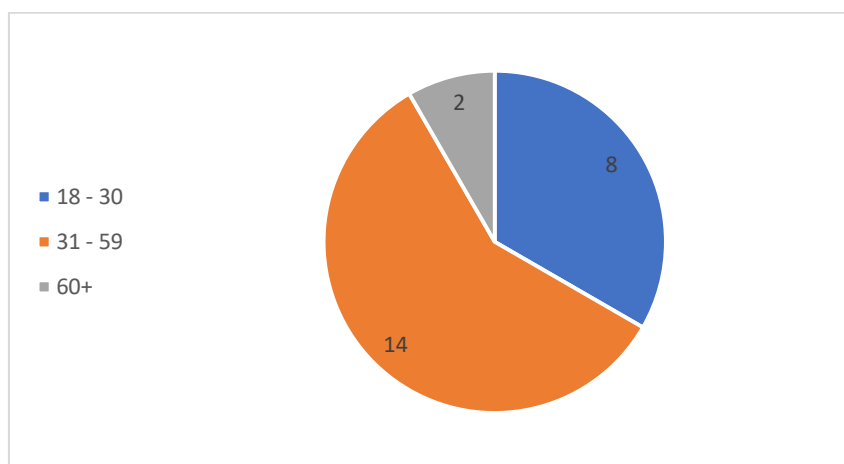
2 Problematika e-learningu a digitálních zdrojů ve výuce češtiny pro cizince

E-learning a digitální zdroje jsou čím dál populárnější. Nejen, že usnadňují práci lektorům, ale pomáhají i studentům při lepším osvojování probírané látky. Hlavní částí výzkumu bylo dotazníkové šetření, jehož cílem bylo zjistit, jak se lektori staví vůči digitálním zdrojům, a hlavně zmapovat, jaké zdroje jsou, případně nejsou k dispozici. V neposlední řadě jsme zjišťovali, zda a do jaké míry používají lektori umělou inteligenci.

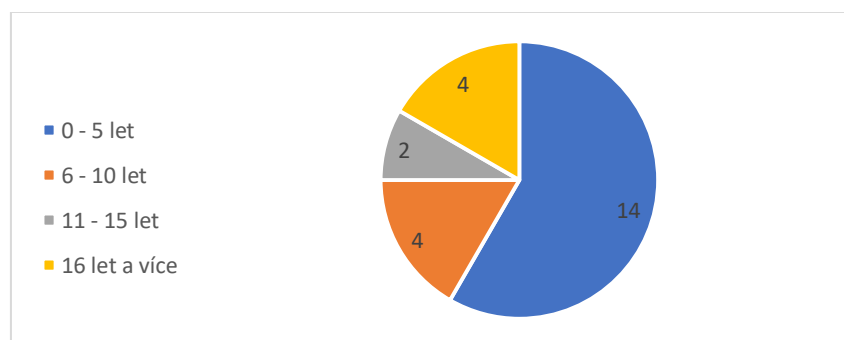
Vymezili jsme si proto několik výzkumných otázek, jež se staly klíčovými pro náš výzkum. Zajímalo nás především:

- zda lektori při své výuce používají spíše klasické či digitální zdroje;
- které typy zdrojů preferují při výuce jednotlivých jazykových dovedností (poslech, mluvení, psaní, čtení);
- jaké digitální zdroje používají a odkud je čerpají;
- důvody, proč při své výuce používají digitální materiály;
- jaké materiály jim chybí;
- zda a jak používají umělou inteligenci.

Sběr dat probíhal během dvou týdnů. Účastníci dotazníkového šetření byli seznámeni s cílem výzkumu a zúčastnili se jej anonymně. Celkem se dotazníku zúčastnilo 24 lektorů různých věkových kategorií a s různě dlouhou výukovou praxí.



Graf 1. Věkové kategorie.



Graf 2. Počet let praxe výukové praxe dotázaných.

Na Grafu 1 vidíme jednotlivé věkové kategorie. Největší počet dotázaných patří do střední věkové kategorie, tedy 31 až 59 let. Na Grafu 2 jsme zobrazili, kolik let praxe jednotliví dotázaní mají. Největší procento lektorů má do 5 let praxe.

3 Výsledky výzkumu

Zde shrneme výsledky našeho výzkumu a zodpovíme výzkumné otázky. Z našeho výzkumu vyplývá, že lektori se pořád při své výuce přiklánějí ke klasickým výukovým zdrojům. Jedenáct účastníků vybralo, že při výuce spíše používají klasické zdroje, oproti sedmi účastníkům, kteří preferují digitální zdroje. Pět účastníků nemá preferenci a jeden z nich na otázku neodpověděl. Zajímavé je však to, že u jednotlivých úrovní upřednostňují lektori spíše digitální zdroje. Tuto diskrepanci lze vysvětlit tím, že se výběr učebních materiálů mění dle potřeb/přání studentů, nebo kvůli praktičnosti – digitální výukové zdroje jsou snadno přístupné a studenti často potřebují pouze připojení k internetu a chytrý telefon či notebook/počítač.

U jednotlivých jazykových dovedností⁵, stejně jako u jazykových úrovní, dominovaly digitální zdroje. Pouze u mluvení je poměr klasických a digitálních zdrojů 50-50 a s výjimkou psaní, kde měly převahu klasické zdroje. Toto nepřekvapuje vzhledem k tomu, že poslechová cvičení jsou většinou k dispozici on-line⁶, a to samé platí i v případě čtení – učitelé mohou využívat autentické texty na různých portálech. Pokud jde o samotné digitální materiály, rozdělili jsme je do pěti skupin – nejpoužívanější jsou výukové aplikace (seznam přikládáme v Tabulce 1), následují interaktivní e-učebnice (např. publikace Lídy Holé), multimediální zdroje (filmy, seriály, podcasty atp.), interaktivní pracovní listy. Mezi námi jmenovanými skupinami je hudba (písně, muzikály atp.) nejméně používaným digitálním materiálem.

Název aplikace/aplikací	Četnost zastoupení mezi všemi odpověďmi
Wordwall	8
Kahoot	6
LearningApps	5
Quizlet	4
Google Jamboard	3
Bamboozle, Čeština levou zadní, Čeština2, Flippiti, Genially, Mentimeter, Microsoft Whiteboard, Moodle, Padlet, Socrative, ToyTheater	1

Tabulka 1. Seznam jmenovaných aplikací mezi dotázanými lektory.

Jak jsme zmínili výše, v Tabulce 1 jsme uvedli seznam aplikací, o nichž se lektori zmínili při vyplňování dotazníku. Nejpoužívanějšími aplikacemi mezi námi dotázanými lektory jsou Wordwall, Kahoot, LearningApps. Toto nás nepřekvapuje, protože aplikace Wordwall a LearningApps nabízí množství různých typů cvičení, které lektori mohou upravovat dle potřeb a sdílet je mezi studenty při synchronní a asynchronní výuce.⁷ Kahoot umožňuje okamžitou interakci se studenty ve formě kvízů, studenti potřebují pouze mobilní telefon, prostřednictvím kterého odpovídají na otázky. Studenti mohou pracovat

⁵ Zde vycházíme z publikace *Společný evropský referenční rámec pro jazyky: jak se učíme jazykům, jak je vyučujeme a jak v jazycích hodnotíme* (Ivanová 2002). Jedná se o poslech, mluvení, psaní a čtení. Pro češtinu jako cizí jazyk byly dosud popsány 4 úrovně (A1, A2, B1 a B2).

⁶ Viz například: <https://www.czechstepbystep.cz/materialy/cesky-krok-za-krokem-1-materialy#audiohavrky>. [poslední přístup: 01.08.2023]

⁷ Více o aplikacích zde: <https://wordwall.net/> a <https://learningapps.org/>. Aplikace Wordwall nabízí zdarma a placenou verzi, aplikace LearningApps je zdarma. [poslední přístup: 01.08.2023]

jak samostatně, tak ve skupinách. Z vlastních zkušeností víme též, že se jedná o aplikaci, která se mezi studenty těší značné popularitě.⁸

Samotné oficiální zdroje digitálních materiálů přinášíme v další tabulce (Tabulka 2).

Název portálu/portálů	Četnost zastoupení mezi všemi odpověďmi
czechstepbystep.cz	16
czechitup.upol.cz	4
meta-ops.eu	3
cestinajehra.cz, inkluzivniskola.cz, mozaika.eu	2
akorpolis.info, cestina-pro-cizince.cz, digitalnilektori.cz, dumy.cz, edu.ceskatelevize.cz, Nakladatelství Karolinum, ProCzeFor	1
Jiné nevýukové zdroje (např. zdroje autentických materiálů) – Internetová jazyková příručka, irozhlas.cz, vaclavhavel.cz	1

Tabulka 2. Seznam oficiálních zdrojů.

V Tabulce 2 jsme uvedli seznam zmíněných oficiálních (a také méně oficiálních, nevýukových) zdrojů. Největší počet námi dotázaných lektorů využívá metodický portál czechstepbystep.cz, který slouží jako metodická podpora k učebnicím Lídy Holé (Česky krok za krokem 1 a 2, Čeština expres 1, 2, 3 a 4, Čeština expres Start, New Czech Step By Step a Tschechisch Schritt für Schritt). Nejen, že tento portál nabízí metodickou podporu ke všem zmíněným učebnicím, ale jsou zde k dispozici i on-line cvičení, krátké české zprávy, tabulky s různými gramatickými jevy, slovní banky a další. Právě proto nás nepřekvapuje význam a popularita tohoto portálu. Dalšími významnými portály jsou též czechitup.upol.cz, portál sloužící jako podpora k učebnicím CzechItUP 1-5, a také webové stránky organizace Meta, jež nabízí velké množství pracovních listů na různá témata.

Digitální materiály lektori nejčastěji používají proto, že učí on-line a tímto si usnadňují práci, dále proto, že digitální materiály mohou přizpůsobit vlastním potřebám a stejně tak je mohou snadněji aktualizovat. Dalším důležitým důvodem je okamžitá interakce studentů (např. zmíněna aplikace Kahoot!) a lektori se domnívají, že digitální materiály jsou často lepší než klasické.

Jedním z cílů tohoto výzkumu bylo zjistit, jaké materiály lektorům chybí. Jedni jsou spokojeni se situací, jiní by ale uvítali změny, například širší nabídku poslechových cvičení pro všechny úrovně, větší množství interaktivních pracovních listů a učebnic. Lektorům chybí též materiály pro vyšší úrovně (B2-C1) a také cvičení zaměřená na složitější gramatické jevy. Dále chybí cvičení na čtení s porozuměním (například pro úrovně A1 a A2, kdy studenti nemohou ještě úplně využívat autentické materiály) a cvičení na psaní.

Práce s umělou inteligencí se ještě mezi námi dotázanými lektory nevžila. 58 % z nich s umělou inteligencí vůbec nepracuje, dalších 42 % naopak ano, ale používají pouze ChatGPT jako pomocníka při přípravě a plánování lekcí.

⁸ Více zde: <https://kahoot.com/>. Jsou k dispozici zdarma a placená verze, která nabízí větší množství aktivit. [poslední přístup: 01.08.2023]

Závěr

Cílem tohoto příspěvku je poukázat na problematiku e-learningu, a hlavně digitálních výukových zdrojů ve výuce češtiny pro cizince. Klasické výukové materiály nejsou zatím na ústupu a mezi lektory jsou stále populární. Ač většina dotázaných lektorů preferuje klasické materiály, jmenovali i různé výukové aplikace, jež ve výuce používají. Příspěvek zahrnoval také mapování aktuální situace, byť na omezeném vzorku, co se týče využívání aplikací a digitálních výukových zdrojů lektory. Tématu umělé inteligence zde byla věnována menší pozornost, nicméně jedná se o téma velice aktuální. Čeština pro cizince je oborem specifickým, nejedná se o světový jazyk a výukové zdroje a aplikace se teprve připravují. Výsledky tohoto výzkumu by mohly sloužit jako základ pro plánovanou disertační práci, kde se bych se tématem zabýval detailněji. Příspěvek může přispět k hlubšímu pochopení problematiky e-learningu a digitálních výukových zdrojů ve výuce češtiny pro cizince a pomoci navrhnout inovativní řešení pro vzdělávání v tomto oboru.

Literatura

Knížní publikace:

BISCHOFOVÁ, Jana, Helena CONFORTIOVÁ, Jitka CVEJNOVÁ, et al. *Prahová úroveň - čeština jako cizí jazyk*. Strasbourg: Council of Europe Publishing, 2001. ISBN 92-871-4761-2.

ČADSKÁ, Milada, Vladimír BIDLAS, Helena CONFORTIOVÁ, Milada TURZÍKOVÁ. *Čeština jako cizí jazyk úroveň A2 podle „Společného evropského referenčního rámce pro jazyky. Jak se učíme jazykům, jak je vyučujeme a jak v jazycích hodnotíme“*. Praha: Univerzita Karlova v Praze, Ústav jazykové a odborné přípravy, 2005.

EGER, Ludvík. *E-learning a jeho aplikace: s orientací na vzdělávání a profesní vzdělávání Millennials*. Plzeň: Západočeská univerzita v Plzni, 2020, 288 s. ISBN 978-80-261-0952-5.

HÁDKOVÁ, Marie, Josef LÍNEK, Kateřina VLASÁKOVÁ. *Čeština jako cizí jazyk úroveň A1*. Olomouc: Univerzita Palackého v Olomouci, Katedra bohemistiky Filozofické fakulty, 2005.

HOLUB, Jan, Ana ADAMOVIČOVÁ, Jana BISCHOFOVÁ, Jitka CVEJNOVÁ, Hana GLADKOVA, Jiří HASIL, Milan HRDLIČKA, Petr MAREŠ, Jiří NEKVAPIL, Zdena PALKOVÁ, Milan ŠÁRA. *Čeština jako cizí jazyk úroveň B2 podle „Společného evropského referenčního rámce pro jazyky. Jak se učíme jazykům, jak je vyučujeme a jak v jazycích hodnotíme“*. Praha: Univerzita Karlova v Praze, Ústav bohemistických studií, 2005.

HRDLIČKA, Milan. *Předložky ve výuce češtiny jako cizího jazyka*. Praha: Univerzita Karlova v Praze, nakladatelství Karolinum, 2000, 142 s. ISBN 80-246-0031-5.

IVANOVÁ, Jaroslava. RADA EVROPY. COUNCIL FOR CULTURAL CO-OPERATION. *Společný evropský referenční rámec pro jazyky: jak se učíme jazykům, jak je vyučujeme a jak v jazycích hodnotíme*. [1. vyd.]. Olomouc: Univerzita Palackého, 2002, xviii, 267 s. ISBN 80-244-0404-4.

ŠEBESTA, Karel. *Vyučování cizího jazyka: terminologický slovník*. Praha: Filozofická fakulta Univerzity Karlovy, 2017, 225 stran; 21 cm. ISBN 978-80-7308-743-2.

Závěrečná práce (bakalářská, diplomová, disertační):

POSAVEC-MALOK, Dean. Problematika osvojování předložek ve výuce češtiny pro Slovany na příkladu chorvatských studentů. Diplomová práce (Mgr.)--Univerzita Karlova. Filozofická fakulta, 2023, 2023, 1 online zdroj.

Učebnice:

HOLÁ, Lída, Pavla BOŘILOVÁ a Zdena MALÁ. Česky krok za krokem 2 = Czech step by step 2 = Tschechisch Schritt für Schritt 2 = Češskij šag za šagom 2. Praha: Akropolis, 2009 - 2016, 4 svazky . ISBN 978-80-86903-92-7.

HOLÁ, Lída a Pavla BOŘILOVÁ. Čeština expres 2: [úroveň] A1/2. Praha: Akropolis, 2011, 96 s. ISBN 978-80-87481-26-4.

HOLÁ, Lída. New Czech step by step. [5. vyd.]. Praha: Akropolis, 2012, 2 sv. (256, 127 s.) . ISBN 978-80-7470-019-4.

HOLÁ, Lída a Pavla BOŘILOVÁ. Čeština expres 3: [úroveň] A2/1. Praha: Akropolis, 2014, 106 s. ISBN 978-80-7470-032-3.

HOLÁ, Lída. Tschechisch Schritt für Schritt. 3., opr. vyd. Praha: Akropolis, 2014, 2 sv. (256, 128 s.) . ISBN 978-80-7470-060-6.

HOLÁ, Lída, Petra BULEJČÍKOVÁ a Silvie PŘEVŘÁTILOVÁ. Česky krok za krokem 1 = Czech step by step 1: A1-A2. Praha: Akropolis, 2016 - 2017, 4 svazky . ISBN 978-80-7470-129-0.

HOLÁ, Lída a Pavla BOŘILOVÁ. Čeština expres. 1 = A1/1. Praha: Akropolis, 2017, 100 stran + 68 stran přílohy . ISBN 978-80-87481-22-6.

HOLÁ, Lída a Pavla BOŘILOVÁ. Čeština expres 4: [úroveň] A2/2. Praha: Akropolis, 2019, 119 s. ISBN 978-80-7470-205-1.

HOLÁ, Lída. Čeština expres START: přípravný kurz k výuce češtiny pro studující. Praha: Akropolis, 2022 - 2022, 95 s. ISBN 978-80-7470-440-6.

HRADILOVÁ, Darina, Božena BEDNAŘÍKOVÁ, Michal ČERMÁK, et al. Czech it UP! 5: čeština pro cizince. Olomouc: Univerzita Palackého v Olomouci, 2019, 2 svazky . ISBN 978-80-244-5592-1.

HRADILOVÁ, Darina, Božena BEDNAŘÍKOVÁ, Michal ČERMÁK, et al. Czech it up! 4: čeština pro cizince. V Olomouci: Univerzita Palackého, 2020, 2 svazky . ISBN 978-80-244-5771-0.

HRADILOVÁ, Darina, Božena BEDNAŘÍKOVÁ, Michal ČERMÁK, Markéta DOSOUDILOVÁ, Michaela KOPEČKOVÁ, Jana MÁŠOVÁ, Jindřiška SVOBODOVÁ a Kateřina SZOKALOVÁ. Czech it up! 3: čeština pro cizince. V Olomouci: Univerzita Palackého, 2021, 2 svazky . ISBN 978-80-244-5945-5.

ŠVARCOVÁ, Tereza a Jakub WENZEL. Czech it up! 1: čeština pro cizince. Olomouc: Univerzita Palackého v Olomouci, 2020, 2 svazky. ISBN 978-80-244-5772-7.

WENZEL, Jakub, Tereza ŠVARCOVÁ a Alena KLIMEŠOVÁ. Czech it up! 2: čeština pro cizince. Olomouc: Univerzita Palackého v Olomouci, 2021 - 2022, 2 svazky . ISBN 978-80-244-6102-1.

Webové stránky:

Czech Step by Step [online], [cit. 06.08.2023]. Dostupné z: <https://www.czechstepbystep.cz/>

Wordwall [online], [cit. 06.08.2023]. Dostupné z: <https://wordwall.net/>

LearningApps [online], [cit. 06.08.2023]. Dostupné z: <https://learningapps.org/>

Kahoot! [online], [cit. 06.08.2023]. Dostupné z: <https://kahoot.com/>

Quizlet [online], [cit. 06.08.2023]. Dostupné z: <https://quizlet.com/>

Google Jamboard [online], [cit. 06.08.2023]. Dostupné z: <https://jamboard.google.com/>

Baamboozle [online], [cit. 06.08.2023]. Dostupné z: <https://www.baamboozle.com/>

Levou zadní – čeština jako druhý jazyk [online], [cit. 06.08.2023]. Dostupné z: <https://ceskylevouzadni.cz/>

Čeština 2 [online], [cit. 06.08.2023]. Dostupné z: <https://www.cestina2.cz/>

Flippity [online], [cit. 06.08.2023]. Dostupné z: <https://www.flippity.net/>

Genially [online], [cit. 06.08.2023]. Dostupné z: <https://genial.ly/>

Mentimeter [online], [cit. 06.08.2023]. Dostupné z: <https://www.mentimeter.com/>

Microsoft Whiteboard [online], [cit. 06.08.2023]. Dostupné z: <https://www.microsoft.com/en-us/microsoft-365/microsoft-whiteboard/digital-whiteboard-app>

Moodle [online], [cit. 06.08.2023]. Dostupné z: <https://moodle.org/?lang=cs>

Padlet [online], [cit. 06.08.2023]. Dostupné z: <https://cs.padlet.com/>

Socrative [online], [cit. 06.08.2023]. Dostupné z: <https://www.socrative.com/>

ToyTheater [online], [cit. 06.08.2023]. Dostupné z: <https://toytheater.com/>

Czech It UP [online], [cit. 06.08.2023]. Dostupné z: <https://czechitup.upol.cz/>

META [online], [cit. 06.08.2023]. Dostupné z: <https://meta-ops.eu/>

Čeština je hra [online], [cit. 06.08.2023]. Dostupné z: <https://cestinajehra.cz/>

Inkluzivní škola [online], [cit. 06.08.2023]. Dostupné z: <https://inkluzivniskola.cz/>

Mozaika [online], [cit. 06.08.2023]. Dostupné z: <https://www.mozaika.eu/>

Nakladatelství Akropolis [online], [cit. 06.08.2023]. Dostupné z: <https://akropolis.info/>

Čeština pro cizince [online], [cit. 06.08.2023]. Dostupné z: <https://cestina-pro-cizince.cz/>

Digitální lektoři [online], [cit. 06.08.2023]. Dostupné z: <https://digitalnilektori.cz/>

DUMY [online], [cit. 06.08.2023]. Dostupné z: <https://dumy.cz/>

ČT edu [online], [cit. 06.08.2023]. Dostupné z: <https://edu.ceskatelevize.cz/>

Nakladatelství Karolinum [online], [cit. 06.08.2023]. Dostupné z: <https://karolinum.cz/>

ProCzeFor (PRONunciation of CZEch for FOReigners) [online], [cit. 06.08.2023]. Dostupné z: <https://www.proczefor.cz/>

Internetová jazyková příručka [online], [cit. 06.08.2023]. Dostupné z: <https://prirucka.ujc.cas.cz/>

iROZHLAS [online], [cit. 06.08.2023]. Dostupné z: <https://www.irozhlas.cz/>

Knihovna Václava Havla [online], [cit. 06.08.2023]. Dostupné z: <https://www.vaclavhavel.cz/>

Kontaktní údaje

Mgr. Dean Posavec-Malok

Centrum pro podporu e-learningu, Ústřední knihovna Univerzity Karlovy

José Martího 407/2, 162 00 Praha 6

e-mail: dean.posavec-malok@ruk.cuni.cz

ZKUŠENOSTI SE SOFTWAREM DVWA VE VÝUCE PŘEDMĚTU BEZPEČNOST INFORMAČNÍCH SYSTÉMŮ

ACTIVATING TEACHING METHODS AND TECHNIQUES FOR SYNCHRONOUS DISTANCE EDUCATION AT UNIVERSITY

Radim Remeš, Ladislav Beránek, Jan Fiala

Abstrakt

Aktivizující výukové metody a techniky podporují vyšší dynamiku výuky a aktivitu studentů ve výuce inženýrských předmětů a přispívají k vyšší úspěšnosti studentů při učení. Příspěvek shrnuje zkušenosti s některými metodami a technikami při vysokoškolské výuce předmětu Bezpečnost informačních systémů a prezentuje výsledky získané výukou tohoto předmětu.

Klíčová slova: výuková metoda, aktivizující metody a techniky, bezpečnost informačních systémů, etický hacking

Abstract

Activating teaching methods and techniques support higher teaching dynamics and student activity in the teaching of IT subjects and contribute to higher student success in learning. The paper summarizes experiences with some methods and techniques in university teaching of the subject Information Systems Security and presents the results obtained by teaching this subject.

Keywords: teaching method, activating methods and techniques, information systems security, ethical hacking

Úvod

Obor informatiky se neustále vyvíjí a s jeho rozvojem roste i význam kybernetické bezpečnosti. Ve světě, kde většina lidí pracuje na systémech, které jsou vzájemně propojené a připojené do společné počítačové sítě, internetu, je ochrana digitálních dat klíčová. Abychom studentům poskytli praktické zkušenosti s pochopením bezpečnostních zranitelností a představili účinné obranné strategie, začlenili jsme do našeho kurikula webovou aplikaci Damn Vulnerable Web Application (DVWA). V tomto článku se podělíme o naše zkušenosti a poznatky získané z výuky předmětu Bezpečnost informačních systémů s využitím DVWA jako praktického nástroje pro výuku na Ekonomické fakultě Jihočeské univerzity.

1 Webová aplikace Damn Vulnerable Web Application

DVWA (GitHub – digininja, 2023) je záměrně navržena jako zranitelná webová aplikace, aby pomohla uživatelům dozvědět se o běžných zranitelnostech webových aplikací. Aplikace poskytuje bezpečné prostředí pro studenty k prozkoumávání a experimentování s různými bezpečnostními chybami, jako jsou SQL injection, cross-site scripting (XSS), command injection nebo brute force. Řešením DVWA příkladů studenti získají cenné poznatky o bezpečnostních zranitelnostech v reálném světě a hlouběji porozumí tomu, jak je může útočník zneužít.

Dostupné příklady a úkoly k prozkoumání obsahují následující zranitelnosti:

- Brute Force
- Command Injection
- Cross Site Request Forgery (CSRF)
- File Inclusion
- File Upload
- Insecure CAPTCHA
- SQL Injection (regular)
- SQL Injection (blind)
- Weak Session ID
- Cross Site Scripting (DOM)
- Cross Site Scripting (Reflected)
- Cross Site Scripting (Stored)
- Content Security Policy Bypass
- JavaScript

Aplikace DVWA konfrontuje studenty s různými bezpečnostními zranitelnostmi, ale zároveň jim umožňuje nalézat a rozvíjet účinné obranné strategie. Zkoumáním různých scénářů útoků a pochopením toho, jak lze zranitelnosti zneužít, jsou studenti lépe vybaveni k identifikaci a implementaci vhodných protiopatření. Studenti se učí důležitosti ověřování vstupů, postupům bezpečného kódování a významu průběžného testování a opravování nedostatečného zabezpečení.

2 Burp Suite

V rámci naší výuky jsem společně s DVWA využívali též nástroj Burp Suite (PortSwigger, 2023). Jedná se o komplexní nástroj pro testování zabezpečení webových aplikací, který nabízí širokou škálu funkcí pro identifikaci a analýzu zranitelností. Tato aplikace je často používána profesionálními penetračními testery a bezpečnostními analytiky pro provádění rozsáhlých testů zabezpečení. V průběhu naší výuky jsme využili zejména funkce *proxy intercept* a *inspector*. Funkcionalitu *proxy intercept* jsme využívali pro zachytávání HTTP požadavků a odpovědí zasílaných mezi prohlížečem a cílovým serverem. Funkci *inspector* pak studenti využívali zejména pro zobrazování a editaci HTTP zpráv, např. při úpravě hodnot u dat cookies.

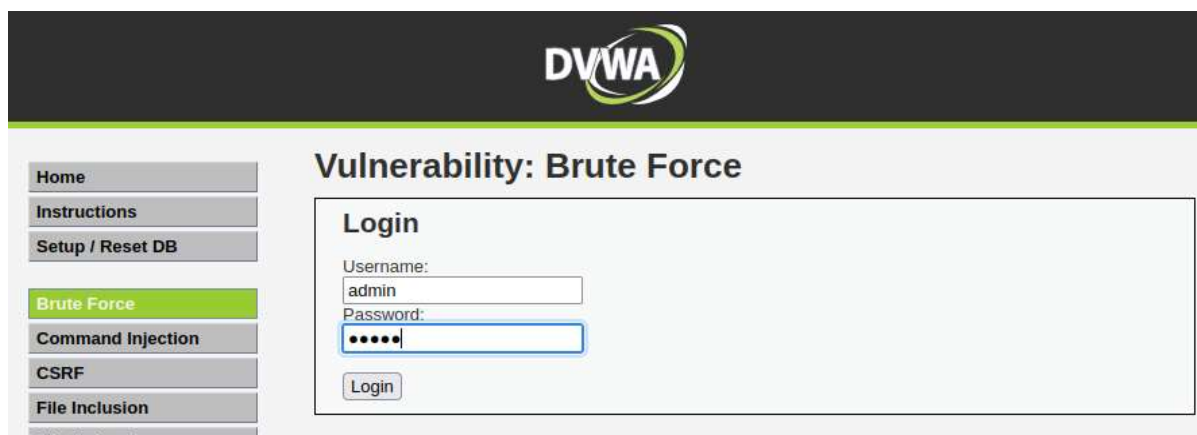
3 Příklady řešených úkolů

V rámci cvičení předmětu Bezpečnost informačních systémů studenti, kromě jiného, řeší několik úkolů, jak obejít zadaná pravidla a zneužít systém s bezpečnostními chybami. Stručný popis některých vybraných úkolů je zmíněn v následujícím textu.

3.1 Útok hrubou silou

Útok hrubou silou (brute force) je útok, který funguje tak, že útočník zkouší různé kombinace slov nebo frází. Účelem je uhodnout např. heslo, adresář nebo cokoliv, co chce útočník zjistit. Obvykle se pro útoky používají velké slovníky. K provedení útoku můžeme použít Burp Suite

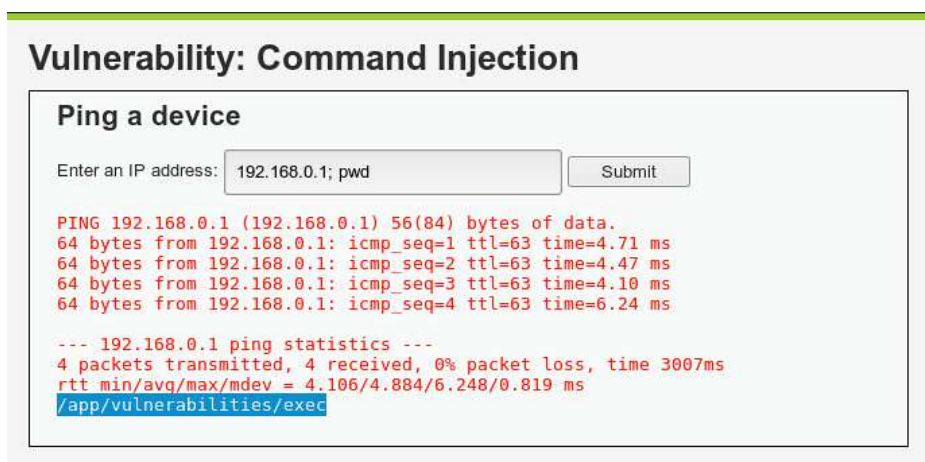
nebo jiný podobný nástroj (např. John The Ripper). Pak jen stačí opakovaně posílat požadavek (viz obr. 1) a čekat, až bude pokus úspěšný.



Obr. 1: DVWA: Ukázka příkladu na útok hrubou silou (brute force)

3.2 Útok vložení příkazu

Cílem tohoto útoku je provádění libovolných příkazů na hostitelském operačním systému prostřednictvím zranitelné aplikace. Tyto útoky bývají aplikovány v případě, když aplikace předává nezabezpečená data od uživatele (např. přes formuláře, cookies, HTTP hlavičky atd.) systémovému shellu. Při tomto útoku jsou útočnickem zadané příkazy v operačním systému obvykle prováděny s oprávněními zranitelné aplikace. Útoky vložení příkazu jsou možné z velké části kvůli nedostatečnému ověření vstupu. Obr. 2 demonstruje přidání příkazu *pwd* (zobrazení aktuálního pracovního adresáře).



Obr. 2: DVWA: Ukázka příkladu na útok vložení příkazu (command injection)

3.3 Útok vložení SQL příkazu

Útok vložení SQL příkazu (SQL injection) je považován za vysoce rizikovou zranitelnost vzhledem ke skutečnosti, že může vést k úplné kompromitaci vzdáleného systému. Proto téměř ve všech penetračních testech webových aplikací jsou aplikace vždy kontrolovány na zranitelnost tohoto útoku. Na obr. 3 je demonstrován příklad, kdy SQL dotaz vrací vždy hodnotu *true* a útočník se tak dostane ke skrytým informacím.



Obr. 3: DVWA: Ukázka příkladu na útok vložení SQL příkazu (SQL injection)

3.4 Útok nahrání souboru

Zranitelnost při nahrávání souborů (file upload) nastává, když webový server umožňuje uživatelům nahrávat soubory do jeho souborového systému, aniž by dostatečně ověřoval věci, jako je jejich název, typ, obsah nebo velikost. Nedůslednost kontroly v těchto případech může znamenat, že např. i základní funkci nahrávání obrázků lze použít k nahrávání libovolných a potenciálně nebezpečných souborů. To mohou být i soubory skriptů na straně serveru, které umožňují vzdálené spuštění kódu (viz obr. 4).



Obr. 4: DVWA: Ukázka příkladu na útok nahrání nežádoucího souboru PHP (file upload)

4 Zpětná vazba od studentů

Výuka bezpečnosti v informatice pomocí DVWA se ukázala jako velmi poutavá a interaktivní zkušenost jak pro studenty, tak pro vyučující. Praktická povaha cvičení podporuje aktivní účast, kritické myšlení a dovednosti při řešení problémů. Studenti nejsou pouze pasivními příjemci teoretických znalostí, ale jsou aktivními účastníky při rozpoznávání, využívání a eliminování

zranitelností. Jednou z největších výhod používání DVWA je jeho schopnost propojit teoretické koncepty se scénáři ze skutečného světa. Díky práci se zranitelnou aplikací, jako je DVWA, studenti získávají praktické zkušenosti, které se přímo projevují v pochopení a kladení důrazu na bezpečné postupy při kódování a implementaci robustních bezpečnostních opatření. Studenti tak mohou být přímými svědky potenciálních důsledků zanedbání bezpečnosti a na základě toho mohou rozvíjet svůj přístup k řešení zranitelných míst.

Studenti sami hodnotí použití DVWA aplikace jako velmi přínosné. Po absolvování předmětu se studenti nejčastěji vyjadřují ve prospěch možnosti samostatného vyzkoušení si etického hackování a možnosti zkusit si hledat další různé zranitelnosti systému. Vzhledem k povaze samotného systému DVWA si každý student může zkusit posouvat své možnosti a limity, neboť aplikace nabízí více úrovní obtížnosti. Při nastavených vyšších úrovních obtížnosti však mnozí studenti již nebyli schopni samostatně úlohy vyřešit a bylo potřeba je nasměrovat několika nápovědami a radami.

Závěr

Výuka bezpečnosti s DVWA podporuje kolaborativní učení mezi studenty. Když procházejí výzvami, které představují zranitelné aplikace, mohou si vyměňovat nápady, sdílet poznatky a spolupracovat na identifikaci zranitelností a navrhování řešení. Tento přístup založený na spolupráci podporuje týmovou práci, komunikaci a hlubší porozumění vyučované látce.

Integrace DVWA do našich osnov se ukázala jako cenný nástroj pro výuku bezpečnosti v informatice. Poskytnutím praktického výukového prostředí studenti získají praktické zkušenosti se zranitelnostmi v reálném světě a rozvinou základní dovednosti v rozpoznání a eliminování bezpečnostních rizik. Interaktivní a poutavá povaha DVWA podporuje proaktivní přístup ke kybernetické bezpečnosti a připravuje studenty na výzvy, kterým mohou čelit ve své budoucí kariéře. Vzhledem k tomu, že se oblast kybernetické bezpečnosti neustále vyvíjí, bude využití nástrojů, jako je DVWA, i nadále klíčové při přípravě odborníků na informatiku připravených chránit počítačové systémy.

Zdroje a relevantní literatura

GitHub – digininja. (2023). DVWA: Damn Vulnerable Web Application (DVWA). Dostupné z: <https://github.com/digininja/DVWA>.

Graham, D. G. (2021). Ethical Hacking. No Starch. ISBN 9781718501874.

Harper, A., Linn, R., Sims, S., Baucom, M., Tejada, H., Fernandez, D., Frost, M. (2022). Gray Hat Hacking: The Ethical Hacker's Handbook, 6th Edition. McGraw-Hill. ISBN 9781264268955.

Parasram, S. V. N. (2020). Digital Forensics with Kali Linux, 2nd Edition. Packt. ISBN 9781838640804.

PortSwigger. (2023). Burp Suite – Application Security Testing Software – PortSwigger. Dostupné z: <https://portswigger.net/burp>.

Rahalkar, S. (2020). Metasploit 5.0 for Beginners, 2nd Edition. Packt. ISBN 9781838982669.

Sabih, Z. (2018). Learn Ethical Hacking from Scratch. Packt. ISBN 9781788622059.

Sinha, S. (2018). Beginning Ethical Hacking with Kali Linux: Computational Techniques for Resolving Security Issues. Apress. ISBN 9781484238912.

Solomon, M. G., Oriyano, S.-P. (2022). Ethical Hacking: Techniques, Tools, and Countermeasures, 4th Edition. Jones & Bartlett Learning. SISBN 9781284249002.

Kontaktní údaje

Mgr. Radim Remeš, Ph.D.

Jihočeská univerzita v Českých Budějovicích

Ekonomická fakulta, Katedra aplikované matematiky a informatiky

Studentská 13, 370 05 České Budějovice,

e-mail: inrem@jcu.cz

Doc. Ing. Ladislav Beránek, CSc., MBA

Jihočeská univerzita v Českých Budějovicích

Ekonomická fakulta, Katedra aplikované matematiky a informatiky

Studentská 13, 370 05 České Budějovice,

e-mail: beranek@jcu.cz

PhDr. RNDr. Jan Fiala, Ph.D.

Jihočeská univerzita v Českých Budějovicích

Ekonomická fakulta, Katedra aplikované matematiky a informatiky

Studentská 13, 370 05 České Budějovice,

e-mail: fiala@jcu.cz

APLIKACE PRO ZRYCHLENÉ VYTVÁŘENÍ TESTŮ PRO LMS MOODLE

APPLICATION TO ACCELERATE THE CREATION OF TESTS FOR LMS MOODLE

František Smrčka

Abstrakt

Tento článek se zabývá vyřešením problémů, jak rychle vytvořit otázky testů do LMS Moodle a jejich možností importu do banky úloh. Tato aplikace řeší pracnost vytváření jednotlivých otázek v bance úloh, kde kromě vložení textu otázek a odpovědí je nutné zaškrtnout a vyplnit řadu dalších nastavení. Aplikace podporuje otázky typu výběr z více odpovědí, otázka ano/ne, krátká tvořená odpověď, dlouhá tvořená odpověď, spojování a numerická úloha. Aplikace je webová, tedy je možné ji používat pomocí webového prohlížeče.

Klíčová slova: Moodle, XML, banka úloh, test, otázky, webová aplikace

Abstract

This article deals with solving the problems of how to quickly create test questions in LMS Moodle and the possibility of importing them into the task bank. This application solves the labor of creating individual questions in the task bank, where, in addition to inserting the text of the questions and answers, it is necessary to check and fill in a number of other settings. The app supports multiple choice, yes/no question, short constructed answer, long constructed answer, matching and numerical problem. The application is web-based, so it is possible to use it using a web browser.

Keywords: Moodle, XML, task bank, test, questions, web application

Úvod

V dnešní době se stále více používá nejen pro potřeby distanční výuky na vysokých školách LMS Moodle (Learning Management System). Vytváření elearnigových kurzů pro jednotlivé předměty v tomto systému není jednoduché a je časově náročné. Větší problém dělá vytváření testů, a to především vytváření otázek v bance úloh, kde je nutné kromě textů otázek a odpovědí nutné nastavit i další parametry, jako je procenta hodnocení, zamíchání odpovědí, zobrazení dalších možností odpovědí nebo možnost zaškrtnutí více odpovědí. Řešením tohoto problému je vytvořit aplikaci, do které by se texty otázek a odpovědí přímo vkládaly a vše by se jednoduše importovalo do LMS Moodle a to do správné kategorie.

1 Název kapitoly první úrovně (např. Shrnutí dosavadních poznatků)

Otázky do banky LMS Moodle je možné otázky vkládat přímo v tomto systému, kde je pro to připravený formulář. Další možností vkládání otázek do banky úloh je využití importu otázek z předpřipraveného vhodného externího souboru. LMS Moodle nabízí několik souborových

formátů pro import dat. (Moodle,2023). Nejvhodnější formáty pro import, které nabízí LMS Moodle jsou GIFT, XML. Tyto formáty jsou obecné, lze v nich nastavit kategorii, do které se importované otázky vloží.

Formát GIFT umožňuje používat textový editor k psaní otázek s výběrem z více odpovědí, pravdivých a nepravdivých, krátkých odpovědí, hledání chybějících slov a číselných otázek v jednoduchém formátu, který lze importovat. Formát GIFT je také formát exportního souboru dostupný v bance otázek. (Gift format, 2023)

XML formát je univerzální, má podobné vlastnosti jako GIFT, a navíc umožňuje import obrázků. Také dokumentace Moodle obsahuje popis převodu textu do XML.

My jsme zvolili pro naši aplikaci formát XML, kde pro převod textu otázek do XML byla naprogramována webová aplikace.

Z hlediska funkčních požadavků jsme uvažovali o těchto typech nejčastěji používaných otázek.

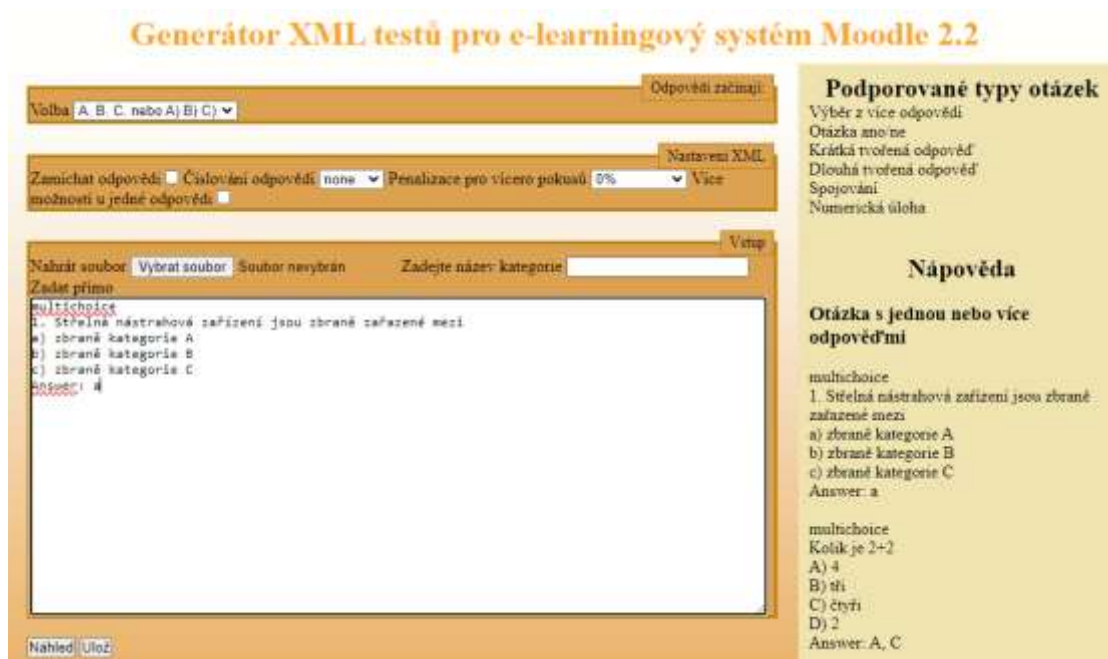
Otázka typu multichoice, kde je možné nastavit více možných odpovědí, včetně penalizace chybných odpovědí. Dále shortanswer, kde student zadává jedno slovo nebo číslo odpovědi. Zde je možnost nastavit více slov se správnou odpovědí. Dalším typem otázky je dlouhá tvořená odpověď, kde student musí vyplnit odpověď více slovy. V tomto typu otázky není automatické hodnocení, ale musí ho lektor dělat ručně. Dále je počítáno s implementací otázek typu ANO/NE, matching, tedy přiřazování dvou sloupců pojmů a numerická úloha, kde po výpočtu musí student zadat konkrétní číslo výsledku.

Mezi další funkční požadavky patří možnost číslovat odpovědi a to A, B, C, nebo A), B) C) nebo (A), (B), (C) případně (1), (2), (3).

Dále je nutné, aby šlo nastavit zamíchání odpovědí, penalizaci pro více pokusů a více možných kladných odpovědí. Po zadání textových otázek a odpovědí, ještě než proběhne generování xml souboru, je vhodné mít náhled otázek s přehledem, zda jsou správně zadané a na kterém místě je případná chyba.

2 Název kapitoly první úrovně (např. Výsledky a diskuze)

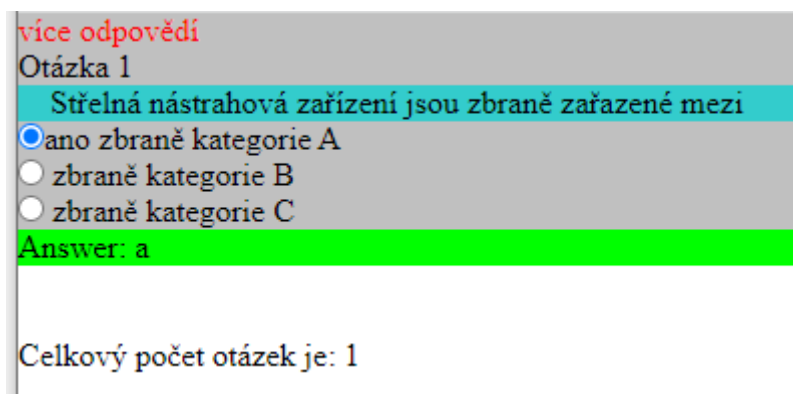
Hlavní obrazovka je rozdělena na uživatelskou část a na část s nápovědou, nacházející se v pravém sloupečku. V nápovědě se nachází seznam podporovaných typů úloh a jednoduchá nápověda, jak zadávat jednotlivé typy úloh. Uživatelská část je rozdělena do třech částí, které jsou pojmenovány: Odpovědi začínají, Nastavení XML a Vstup. Náhled hlavní obrazovky je vidět na obrázku 1.



Obrázek 1: Obrazovka aplikace. Zdroj: vlastní

Název kategorie využívá systém Moodle k vytvoření nové kategorie, pokud není, případně vložení otázek do kategorie, která v bance úloh v Moodle již existuje. Pokud se nezadá název kategorie, aplikace si sama doplní vlastní název před samotným vygenerováním XML souboru.

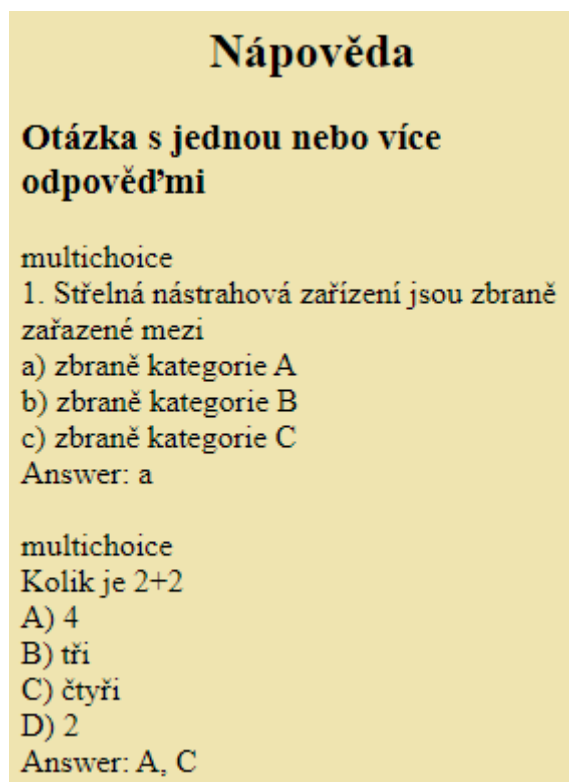
Před samotným vygenerováním XML souboru s pomocí tlačítka zobrazíme náhled. Tento náhled slouží spíše pro kontrolu, než aby nám ukázal, jak systému Moodle. Ukázkou náhledu můžeme vidět na obrázku 2. Zde je vidět typ otázky, správná odpověď a že v otázce nejsou chyby.



Obrázek 2: Náhled zadaných otázek. Zdroj: vlastní

Po kliknutí na tlačítko Ulož se vytvoří XML soubor, který je kompatibilní s importovacím XML souborem do banky úloh v Moodle.

Ke všem otázkám je vytvořena jednoduchá nápověda s příkladem zadání textů do textových otázek. Vše je intuitivní. V rámci generování souboru otázkami je možné zvolit u všech otázek zamíchání odpovědí, nastavení více správných možností u odpovědí a penalizace pro více odpovědí. V aplikaci je také možnost zadat název kategorie, do které se otázky v rámci importu do LMS Moodle zařadí.



Obrázek 3: Část nápovědy pro zadávání otázek. Zdroj: vlastní

První verze této aplikace byla vytvořena již v 2011 (Informatika,2013). Tato poslední verze je doplněna o otázky typu dlouhá tvořená odpověď, spojování (matching) a numerická úloha. Je zpřístupněna na URL adrese: <http://195.113.207.163/~moodle/>

Práce s touto aplikací je sice jednoduchá, ale je potřeba dbát na některé zásady správného vkládání, jako je? nedávat mezery před text, nečíslovat odpovědi v textu případně jsou problémy s některými znaky typu @ a podobně, které se xml používají, jaké řídicí znaky.

Závěr

Tento článek ukazuje vyřešení problémů s rychlým vytvořením otázek testů do LMS Moodle a jejich importem v rámci LMS Moodle do banky úloh. Aplikace podporuje téměř všechny typy otázek, jako jsou: výběr z více odpovědí, otázka ano/ne, krátká tvořená odpověď, dlouhá tvořená odpověď, spojování a numerická úloha. Aplikace je webová, tedy je možné ji používat pomocí webového prohlížeče. První verze této aplikace byla vytvořena jako bakalářská práce (Jonák, 2011). V poslední verzi jsou odstraněny chyby, například s náhledy některých typů otázek a přidána možnost vkládat další typy otázek.

Literatura

Gift format (2023) [online], [2023-06-01]. Oficiální stránky. Dostupné na https://docs.moodle.org/39/en/GIFT_format

Jonák, M. (2011) *Generátor XML testů pro elearningový systém Moodle 2.2*, Bakalářská práce. Jihlava, Vysoká škola polytechnická.

Moodle (2023). [online], [2023-06-01]. Oficiální stránky. Dostupné na: <https://moodle.com/>

Smrčka, F. (2013). Konvertor testů pro LMS Moodle, Sborník abstraktů z mezinárodní odborné pedagogicky zaměřené konference, *Informatika XXVI/2013*, Provozně ekonomická fakulta Mendlova univerzita Brno.

Kontaktní údaje

PaedDr. František Smrčka

Katedra technických studií, Vysoká škola polytechnická Jihlava

Tolstého 16, 586 01 Jihlava

e-mail: smrcka@vspj.cz

VÝUKA PŘEDMĚTU BUSINESS INTELLIGENCE NA SU OPF V KARVINĚ

TEACHING THE COURSE OF BUSINESS INTELLIGENCE AT SU SBA IN KARVINA

Petr Suchánek

Abstrakt

Řízení moderních organizací je závislé na kvalitě výstupů analýz dostupných dat a modelovaných a simulovaných prognóz předpokládaného budoucího vývoje sledovaných ukazatelů. Nástroje, metody, postupy a principy zpracování podkladů pro rozhodování náleží do oblasti zvané Business Intelligence (BI). Na SU OPF v Karvině je vyučován předmět Business Intelligence jehož cílem je seznámit studenty s rolí BI v současných systémech řízení podniků a institucí. Cílem článku je prezentovat zkušenosti s výukou daného předmětu v prezenční a kombinované formě studia.

Klíčová slova: Business Intelligence, Competitive Intelligence, Manažerská informatika, Power BI

Abstract

The management of modern organizations depends on the quality of outputs of analyses of available data and modeled and simulated forecasts of the expected future development of monitored indicators. Tools, methods, procedures and principles of data processing for decision-making belong to the area called Business Intelligence (BI). At SU SBA in Karvina, the course Business Intelligence is taught and its goal is to introduce students the role of BI in current management systems of enterprises and institutions. The goal of the article is to present the experiences of teaching the given course.

Keywords: Business Intelligence, Competitive Intelligence, Managerial Informatics, Power BI

Úvod

Strategický cíl Obchodně podnikatelské fakulty v Karvině Slezské univerzity v Opavě (SU OPF) je zajišťovat kvalitní vzdělávání v profesně a akademicky zaměřených studijních programech tak, aby absolventi SU OPF nacházeli odpovídající uplatnění na tuzemském i zahraničním trhu práce. Tento cíl je naplňován, mimo jiné, vhodnou strukturou jednotlivých akademicky a profesně zaměřených studijních programů obsahujících předměty, které by měly studentům, resp. absolventům poskytnout adekvátní základ i hlubší znalosti, odborné a praktické dovednosti a kompetence vázané, mimo jiné, na oborové oblasti ekonomiky, managementu a informatiky vesměs v interdisciplinárním pojetí. Cílem a nutností managementů moderních podniků je vydávat kvalifikovaná rozhodnutí směřující k zvyšování efektivity a produktivity a snižování rizik. Základem kvalifikovaných rozhodnutí jsou podklady vzniklé důkladnou analýzou dostupných relevantních dat a v celé řadě případů výstupy simulací poskytující s různou mírou nejistoty predikce chování vztažných systémů a monitorovaných ukazatelů. Zdroje dat jsou dnes velmi obsáhlé a v této souvislosti hovoříme o souborech dat označovaných jako Big Data (Hendl, 2021). Právě zde sehrává dnes již standardní roli Business Intelligence (BI), což je, jak je uvedeno například v (Sharda, Delen a Turban, 2017), (Suchánek, 2018) a celé řadě dalších publikací, soubor technologií, procesů, nástrojů a metod umožňujících organizacím sbírat, analyzovat, interpretovat a v neposlední řadě prezentovat data z různých

interních a externích zdrojů. Výstupy BI v podobě reportů pak umožňují manažerům, podnikatelům a dalším uživatelům lépe poznat a pochopit důležité souvislosti a vydávat kvalifikovanější rozhodnutí zvyšující efektivitu jejich podnikatelských činností. Všechny uvedené skutečnosti byly klíčovým důvodem, proč byl na SU OPF předmět s názvem Business Intelligence zaveden. V akademickém roce bude probíhat v pořadí třetí výuka daného předmětu. Cílem článku je seznámit čtenáře s daným předmětem z hlediska jeho zařazení, obsahu a způsobu výuky

1 Zařazení předmětu

Předmět Business Intelligence je povinným předmětem akademicky zaměřeného navazujícího magisterského studijního programu Manažerská informatika (dle Nařízení vlády č. 275/2016 Sb., ze dne 24. srpna 2016 o oblastech vzdělávání ve vysokém školství jde o kombinovaný studijní program s podílem ekonomické a infromatické oblasti vzdělávání cca 50% / 50%) a dále je zařazen jako povinně volitelný předmět do studijního plánu akademicky zaměřeného navazujícího magisterského studijního programu Ekonomika a management, specializace Podnikání. Mimo studentů uvedených studijních programů si předmět mohou vybrat jako volně volitelný i studenti všech ostatních navazujících magisterských studijních programů uskutečňovaných na SU OPF (Bankovníctví, peněžnictví, pojišťovnictví, Ekonomika a management, specializace Obchod a marketing a Finance, účetnictví a daně, Veřejná ekonomika a správa).

2 Cíl a obsah předmětu

Cílem předmětu je seznámit studenty s problematikou a rolí BI v současných systémech řízení podniků a institucí a realizací datové analýzy z interních a externích zdrojů s využitím nástroje Power BI. Obsahová náplň předmětu je obsažena v Tabulce 1.

Tabulka 1: Osnova předmětu

Tematický celek	Obsahová náplň
Business Intelligence (BI) - základní principy	Vymezení podstaty BI. Vývoj BI. BI v řízení firmy. Hlavní komponenty BI - datový sklad, datové tržiště, ETL, OLAP databáze, dočasné úložiště dat, operativní úložiště dat. Typy aplikací BI. Vrstvy BI. Integrace komponent BI.
Realizace úloh BI	Analýza připravenosti podniku na implementaci BI. Analýza uživatelských požadavků. Modelování a model BI řešení. Návrh architektur BI. Návrh technologické platformy. Aplikační oblasti BI.
Řízení projektů a aplikací BI v podniku	Přístupy k řešení BI. Strategie rozvoje BI v podniku. Projekty BI. Plánování projektu, projektový tým a klíčové role v projektu řešení BI. BI v kontextu řízení SI/ICT. Podstatné parametry a charakteristiky pro implementaci a řízení řešení BI.
Aplikační oblasti BI	Aplikace BI ve financích, marketingu, výrobě, logistice, řízení lidských zdrojů, informatice, a dalších. Příklady konkrétních řešení a aplikací.
Competitive Intelligence (CI)	CI jako podmnožina BI. Funkce CI. Metody CI. Technologické nástroje CI. Aplikace CI. CI jako podpora strategického řízení podniku. Vývojové trendy CI.
Faktory úspěšnosti a trendy vývoje BI	Úroveň řízení firmy jako celku a IS/ICT. Identifikace přínosů BI. Vývoj standardů BI. Vývoj architektur BI. Vývoj komponent BI. Dodavatelský způsob vývoje řešení BI.

Zdroj: <https://is.slu.cz/predmet/opf/INMNPBIN>

3 Výuka předmětu

Předmět je vyučován pro studenty prezenční (rozsah 2 hodiny přednášky a 1 hodina semináře na počítačové učebně) i kombinované (rozsah 16 hodin / semestr) formy studia. V rámci přednášek pro prezenční formu studia jsou studenti seznamováni s jednotlivými obsahovými celky, přičemž snahou je prezentovat jim vždy danou problematiku na konkrétních ukázkách z praxe. V této návaznosti jsou součástí prezentací, mimo jiné, odkazy na vhodné internetové zdroje, ze kterých jsou zřejmé příklady dobré praxe a dále příklady využívání BI v některých spolupracujících firmách, které poskytly vybrané příklady. Minimálně jednou za semestr je výuka realizována odborníkem z praxe, který studentům představí konkrétní nové možnosti, postupy, nástroje a technologie využívané v příslušné firmě.

V rámci přednášek jsou studentům prezentovány a vysvětlovány tematické celky vázané na architekturu a způsoby využití a implementace BI ve firmách, přičemž vše je vesměs vázáno na produkční a zdrojové systémy, dočasná úložiště dat (DSA - Data Staging Area), operativní úložiště dat (ODS - Operational Data Store), transformační nástroje (ETL - Extract, Transform, Load), integrační nástroje (EAI - Enterprise Application Integration), datové sklady (DWH - Data Warehouse), datová tržiště (DMA - Data Marketplace), OLAP (Online Analytical Processing), reporting, manažerské aplikace (EIS - Executive Information System), dolování dat (Data Mining), nástroje pro zajištění kvality dat, nástroje pro správu dat, metody, nástroje a postupy z oblasti Competitive Intelligence a v neposlední řadě datovou analytiku.

Obsah seminářů navazuje na obsah přednášek a prakticky během celého semestru je naplní seminářů datová analytika realizovaná s využitím analytické služby Power BI vyvinuté společností Microsoft. V rámci datové analýzy se studenti seznamují s dílčími kroky, kterými jsou definice cílů analýzy, sběr dat, příprava dat, tvorba datového modelu, analýza dat, vizualizace dat, interpretace výsledků, prezentace, sdílení, revize výsledků a případné opakování analýzy a definice rozhodnutí.

Hodnocení předmětu v prezenční i kombinované formě studia je založeno na ověřování praktických i teoretických dovedností. Celkově mohou studenti získat za celý semestr 60 bodů, přičemž minimum bodů pro úspěšné absolvování předmětu je 36. Během výuky mohou studenti získat 20 bodů za zpracování seminární práce, její odevzdání a obhajobu (vše uvedené je jednou z nutných podmínek pro absolvování předmětu a pro možnost přihlásit se na zkuškový termín) a 40 bodů za ústní zkoušku. Ústní část zkoušky je pro studenty přínosná v tom, že si svým způsobem vyzkouší, jak by to probíhalo, kdyby si u státní závěrečné zkoušky náhodně vybrali některý z okruhů týkajících se dané tematiky.

Seminární práce představuje realizaci datové analýzy reflektující teoretické znalosti a praktické dovednosti prezentované a trénované během přednášek a seminářů. Úkolem studentů je v souladu s postupem cílené datové analýzy definovat manažerské otázky, nalézt, připravit a analyzovat příslušná data, vytvořit adekvátní reporty, provést příslušnou interpretaci a definovat konkrétní manažerská rozhodnutí. Jako ukázky témat seminárních prací vytvořených studenty lze uvést řešení problematiky návrhu stanovení lokalit pro hromadná ubytování (kempy) v daném kraji, dostupnost odtahových služeb, optimalizace obchůzek a tvorba kampaní v návaznosti na řešení obecné kriminality atd.

SU OPF se aktuálně zaměřuje na zkvalitňování výuky prostřednictvím nových přístupů a aplikací moderních výukových metod do výuky jednotlivých předmětů. Slezská univerzita jako celek je zapojena do CRP projektu Standardy kvality výuky na českých vysokých školách a na SU OPF již více než rok působí tzv. akční skupina pro inovace výuky, která pořádá pro akademické pracovníky různé akce, na které zve kompetentní odborníky na vysokoškolskou pedagogiku, didaktiku atd. Jednotlivé akce jsou zdrojem možných nových přístupů a metod

uplatnitelných ve výuce. Předmět Business Intelligence svým charakterem umožňuje jak v rámci přednášek, tak i seminářů, a to i s ohledem na počty studentů, aplikovat během výuky celou řadu metod postavených na způsobech komunikace se studenty, skupinových činnostech, samostatné přípravě, podpoře kritického myšlení při analýze dat a interpretaci výsledků atd.

4 Výukové materiály

Výukové materiály jsou studentům přístupné prostřednictvím Informačního systému Slezské univerzity (dále jen IS SU), kterým je informační systém od Masarykovy univerzity v Brně. Studenti prezenční i kombinované formy studia mají k dispozici všechny studijní materiály a informace k předmětu a výuce, přičemž konkrétně jde o údaje o předmětu a jeho výuce, požadavky na absolvování předmětu, způsoby komunikace a konzultací, studijní text (studijní opora, resp. text splňující požadavky kladené na distanční studijní texty), prezentace přednášek, literaturu a další materiály, které jsou aktuálně k dispozici například z firem nebo Internetu. Strukturu tzv. interaktivní osnovy využívané na SU OPF lze nalézt v (Suchánek, 2022). Studijní opora je vytvořena jako distanční studijní text. Vývoj současné informatiky je velmi rychlý, a proto knižní publikace zaměřené na dané oblasti dosti rychle zastarávají. Pravdou ovšem je, že pokud jde o metody, přístupy apod. tak tyto jsou v celé řadě případů platné a použitelné i ze starších publikací.

5 Power BI v rámci výuky seminářů

Jak již bylo uvedeno v kapitole 3, jako nástroj pro zpracování dat v rámci seminářů je využíván produkt Power BI umožňující uživatelům vytvářet interaktivní reporty a dashboardy, které lze sdílet s ostatními. Power BI je využíván v celé řadě firem, které má SU OPF jako strategické nebo odborné partnery, a proto je využívání tohoto nástroje v rámci výuky zcela opodstatněné. Tematika Power BI je obsahem celé řady, vesměs zahraničních, publikací, kdy jako příklady lze uvést (Aspin, 2020), (Clark, 2020), (Decker a Powell, 2022) nebo (Singh a Collie, 2021). Studenti mají uvedené publikace jako povinnou nebo doporučenou literaturu, ovšem hlavním zdrojem jsou odkazy na Internetu, které jsou primárním zdrojem pro práci na seminářích.

Hlavními zdroji pro výuku jsou odkazy z webu (anglické i české) <https://learn.microsoft.com>, v rámci kterého lze volně nalézt prakticky všechny postupy a návody potřebné pro výuku daného předmětu. Primárním odkazem pro seznámení se s využitím Power BI, instalací, připojení k datům a prostředím je <https://learn.microsoft.com/cs-cz/power-bi/fundamentals/desktop-getting-started>. Po seznámení se s prostředím a základními postupy se dále pracuje nad databázemi, které jsou dostupné v podobě ukázek pro Power BI, a nad kterými jsou vysvětlovány a prakticky prováděny jednotlivé úkony a vytvářeny konkrétní dílčí výstupy. Dílčími oblastmi, se kterými se studenti postupně seznamu, jsou:

- Zdroje a import dat - <https://learn.microsoft.com/cs-cz/power-bi/connect-data/desktop-data-sources>;
- Vytváření a správa relací - <https://learn.microsoft.com/cs-cz/power-bi/transform-model/desktop-create-and-manage-relationships>;
- DAX v Power BI - <https://learn.microsoft.com/cs-cz/power-bi/transform-model/desktop-quickstart-learn-dax-basics>;
- Vytváření měř a vlastních měř pro analýzu dat
 - <https://learn.microsoft.com/cs-cz/power-bi/transform-model/desktop-measures>;

- <https://learn.microsoft.com/cs-cz/power-bi/transform-model/desktop-tutorial-create-measures>;
- Tvorba interaktivních sestav - <https://learn.microsoft.com/cs-cz/power-bi/fundamentals/service-get-started>;
- Sdílení sestav a řídicích panelů
 - <https://learn.microsoft.com/cs-cz/power-bi/collaborate-share/service-share-dashboards>;
 - <https://learn.microsoft.com/cs-cz/power-bi/collaborate-share/service-share-reports>.

Obsah uvedený pod daným odkazy představuje základní východiska, přičemž na seminářích jsou průběžně realizovány úkoly představující různé varianty, na kterých se studenti učí důkladněji se orientovat v dané problematice a pochopit i hlubší souvislosti.

6 Vybrané skutečnosti a zkušenosti s výukou předmětu

- V návaznosti na zařazení předmětu uvedeného v kapitole 1 je nutné konstatovat, že během dvou běhů výuky daného předmětu ho zatím absolvovali pouze studenti navazujícího magisterského studijního programu Manažerská informatika (jako povinně nebo volně volitelný si ho zatím žádný student nevybral). Důvodem je zřejmě obava studentů z infromatického zaměření, používaných metod apod.
- Studenti hodnotí výuku předmětu a jeho obsah kladně, protože vidí možnosti konkrétního využití, a to nejen pro praxi po absolvování VŠ, ale i například pro tvorbu závěrečných prací.
- Minimálně jednou za semestr je výuka realizována odborníkem z praxe. Tato výuka proběhne vesměs vždy cca v 9-10 týdnu výuky, kdy studenti již mají zvládnutou datovou analytiku a prostředí Power BI. Studenti byli zatím s výukou externího odborníka vždy spokojeni a kladně výuku hodnotil i samotný externí vyučující, který zatím vždy vyzdvihl znalosti studentů, kteří s ním během výuky drželi krok, a to i přes skutečnost, že výuka probíhala v angličtině.
- Ústní část zkoušky je pro studenty přínosná v tom, že si vyzkouší odpovídat na konkrétní dotazy, na základě, kterých je možné identifikovat, zda rozumí dané problematice a souvislostem. Tento způsob zkoušení je kladně hodnocen i studenty, pro které je to v zásadě i příprava na státní závěrečnou zkoušku, která probíhá formou ústní zkoušky.

Závěr

Předmět Business Intelligence patří k relativně novým předmětům vyučovaným na SU OPF. Během zatím dvou běhů výuky se ukázalo, že pro studenty akademicky zaměřeného navazujícího magisterského studijního programu Manažerská informatika je předmět přínosný a považují ho za smysluplný a užitečný. Dá se předpokládat, že povědomí o daném předmětu se mezi studenty rozšíří a postupně si ho budou vybírat jako povinně nebo volně volitelný i studenti jiných studijních programů uskutečňovaných SU OPF. Cílem je dále rozvíjet výuku daného předmětu jak z obsahového, tak i metodického hlediska.

Poděkování

Tento článek byl podpořen Ministerstvem školství, mládeže a tělovýchovy ČR v rámci Institucionální podpory dlouhodobého rozvoje výzkumné organizace v roce 2023.

Literatura

- Aspin, A. (2020). Pro Power BI Desktop: Self-Service Analytics and Data Visualization for the Power User. New York: Apress. ISBN 978-1484257623.
- Clark, D. (2020). Beginning Microsoft Power BI: A Practical Guide to Self-Service Data Analytics. New York: Apress. ISBN 978-1484256190.
- Deckler, G., Powell, B., (2022). Mastering Microsoft Power BI - Second edition. Birmingham: Packt Publishing. ISBN 978-1801811484.
- Hendl, J. (2021). Big data - Věda o datech, základy a aplikace. Praha: Grada. ISBN 978-80-271-3031-3.
- Sharda, R., Delen, D., Turban, E. (2017). Business Intelligence, Analytics, and Data Science: A Managerial Perspective. London: Pearson. ISBN 978-0134633282.
- Singh, A., Collie, R. (2021). Power Pivot and Power BI: The Excel User's Guide to DAX, Power Query, Power BI & Power Pivot in Excel 2010-2016. Holy Macro Pr. ISBN 9781615470754.
- Suchánek, P. (2018). Business Intelligence. Karviná: SU OPF. ISBN 978-80-7510-308-6.
- Suchánek, P. (2022). Vývoj výuky předmětu Podnikání na internetu na SU OPF v Karviné. Konference Informatika 2022. ISBN 978-80-88064-63-3.

Kontaktní údaje:

doc. Mgr. Petr Suchánek, Ph.D.

Slezská univerzita v Opavě, Obchodně podnikatelská fakulta v Karviné

Katedra informatiky a matematiky

Univerzitní nám. 1934/3, 73340 Karviná

e-mail: suchanek@opf.slu.cz

AUTOMATICKÁ KONTROLA ÚLOH Z PROGRAMOVÁNÍ

AUTOMATIC GRADING OF PROGRAMMING TASKS

Martin Vitek

Abstrakt

Klíčová slova: CodeRunner, Jobe, opravování úkolů, test, známkování

Abstract

In our programming courses, we were looking for a solution to test hundreds of first-year students' knowledge effectively. We switched from manually grading homework and tests to grading them automatically in Moodle, thanks to the CodeRunner plugin. This article describes how the system works, its capabilities, and our experience with testing students over the past few semesters.

Keywords: CodeRunner, grading, Jobe, quiz, validation

Úvod

Katedra elektrotechniky a počítačového modelování zajišťuje dva programovací předměty, které jsou povinné pro studenty prvního ročníku. Jedná se o předmět Základy programování pro elektrotechniku, ve kterém se vyučují základy C++ a předmět Modelování v Matlabu a Simulinku. Na tyto předměty bývá zapsáno 300 až 500 studentů.

Pro studenty potřebujeme mít procvičovací úlohy, úkoly a testy, na základě, kterých jsou v semestru hodnoceni. Opravovat testy ručně je časově náročné, může být také chybové a případně i neobjektivní. Některé testy tedy byly vytvořeny v e-learningovém systému Moodle. Jednak jako otázky s výběrem jedné, nebo více správných odpovědí a otázky s připravenou kostrou kódu, ve které je nutné vybrat správné příkazy z nabídky připravených.

Toto řešení je mnohem lepší, než ruční opravování, ale stále má několik problémů. Otázky jsou složité na vytváření, protože zvláště otázky se strukturou kódu se v základních typech Moodle otázek tvoří poměrně komplikovaně. Studenty ale také chceme testovat více prakticky. Tedy tak, aby museli sami vymyslet a naprogramovat řešení zadaného problému.

Zvažovali jsme tedy vytvoření vlastního systému, který by umožňoval spouštět kód studentů, testoval by jeho funkci a ohodnotil by ho. Nebo by bylo možné využít existující nástroje pro Continuous Integration/Continuous Delivery (CI/CD), které slouží k automatickému sestavování, testování a nasazení programů, a vytvořit k nim jen uživatelské rozhraní. Nakonec jsme vybrali plugin CodeRunner, který do systému Moodle přidává právě tyto schopnosti a mnoho dalších. Tento článek popisuje, jak plugin CodeRunner zprovoznit, jakým stylem se v něm vytvářejí úlohy a jaké s ním máme zkušenosti po dvou letech používání.

1 Architektura systému

Celý kontrolní systém se skládá ze dvou částí. První je plugin CodeRunner, který se instaluje do systému Moodle. Plugin vytváří nový typ otázky CodeRunner, která umožňuje testování kódu. Testy se poté skládají z těchto otázek, které lze kombinovat i s otázkami jiných typů.

Při požadavku na kontrolu dané otázky se studentův kód, včetně nastavení pro požadované prostředí, odešle z Moodle na validační server. Kód je na něm spuštěn v požadovaném prostředí a výsledky jsou odeslány zpět do Moodle, kde se porovnají se vzorovým řešením, ohodnotí a zobrazí se uživateli. Server komunikuje se systémem Moodle přes REST API. Je tedy možné naprogramovat si server vlastní. Není to ale nutné, jelikož autor pluginu implementoval vlastní validační server Jobe, který má všechny potřebné vlastnosti. Kód je kvůli bezpečnosti spuštěn v sandboxu a tím je izolován od systému serveru. Pro jednotlivé úlohy lze nastavit limit na dostupnou RAM, maximální čas běhu apod.

Jedna instance Moodle může komunikovat s více validačními servery, které lze nastavit jednotlivě pro každou otázku. Každý vyučující si tedy může spravovat svůj server, na kterém může mít např. stejnou verzi překladače jako na učebně. Zároveň tak lze rozložit zátěž, pokud by docházelo ke kontrole velkého množství úloh v jeden okamžik.

2 Testové otázky

2.1 Programovací jazyk

Podpora pro některé programovací jazyky (C, C++, Java, Python atd.) je integrována ve validačním serveru Jobe. Lze je tedy použít bez dalšího nastavování. Další jazyky je možné přidat přidat přes tzv. prototypy otázek. Jedná se o jednoduchý systém, kdy se využije některý z integrovaných jazyků, ve kterém se napíše „recept“, který definuje, jak se má kód v požadovaném jazyce spustit a zkontrolovat.

Tento přístup má tu výhodu, že umožňuje plnou kontrolu nad zpracováním otázky a díky tomu má uplatnění také u integrovaných jazyků. Např. umožňuje spustit požadovaný překladač se sadou vlastních parametrů, nebo před samotným překladem spustit linter, či jinou kontrolu kódu.

2.2 Kontrola řešení

Řešení se kontroluje libovolným počtem testovacích případů. V každém lze nastavit kód, který volá testované řešení a vstup na konzoli, který je programu předán. Výsledné hodnocení otázky je založeno na počtu splněných a chybných testovacích případů.

Kontrola správnosti testovacího případu je založena na porovnání jeho správné odpovědi s textovým výstupem programu do konzole. V jednoduchých otázkách může testovaný program přímo vypsat výsledek výpočtu. Ve složitějších může být třeba do otázky přidat kód (skrýt před studenty), který výsledek programu zvaliduje a vypíše např. „dobře“/„špatně“.

3 Příklady otázek

3.1 C++ jednoduchá funkce

Student má vytvořit funkci, která ze zadaného čísla vypočítá třetí mocninu a výsledek je návratová hodnota funkce. Jednotlivé testovací případy volají studentovu funkci s různými hodnotami parametru a hodnotu získanou z funkce vypisují. Prototyp otázky nejprve vytvoří cpp soubor, do kterého vloží studentovu funkci, funkci main() s kódem z daného testovacího případu a poté soubor přeloží a spustí.

```
1 double tretí_mocnina(const double cislo)
2 {
3     return cislo * cislo;
4 }
```

Test	Expected	Got	
std::cout << tretí_mocnina(2.0);	8	4	✘
std::cout << tretí_mocnina(-3.0);	-27	9	✘
std::cout << tretí_mocnina(0.0);	0	0	✔

Obrázek 1: Příklad jednoduché C++ otázky s provedenými testovacími případy.

3.2 C++ celý program

Student má vytvořit program, který z konzole načte slovo (povolání) a použije ho ve větě, kterou vypíše. Zde prototyp otázky vytvoří cpp soubor pouze se studentovo kódem, soubor přeloží a spustí. Testovací případy mají nastaveny texty, které se programu předají jako vstup z konzole.

		Input	Expected	Got	
1	<code>#include <iostream></code>				
2	<code>#include <string></code>				
3					
4	<code>int main()</code>	student	student je narocne povolani	student je narocne povolani	✓
5	<code>{</code>	ucitel	ucitel je narocne povolani	ucitel je narocne povolani	✓
6	<code> std::string povolani;</code>				
7	<code> std::cin >> povolani;</code>				
8	<code> std::cout << povolani << " je narocne povolani";</code>	kuchar	kuchar je narocne povolani	kuchar je narocne povolani	✓
9	<code>}</code>				

Obrázek 2: Příklad otázky programu C++ s provedenými testovacími případy.

3.3 Matlab skript

Student má vytvořit Matlab skript, který vytvoří proměnnou s poloměrem koule a vypočítá její objem, který uloží do další proměnné. Prototyp otázky sloučí studentův skript se skriptem z testovacího případu a výsledný skript spustí v Matlabu. Jelikož se v Matlabu často zpracovávají poměrně velké matice, tak je u otázky přiložena funkce, která porovnává dvě matice (studentovu a správné řešení) a vypisuje „ok“, pokud jsou shodné a jinak vypíše porovnání matic. Pro kontrolu správnosti výsledků jsou přiloženy další funkce, které dle zadání počítají správné hodnoty a porovnávají je s výsledky studenta.

	Test	Expected	Got	
1	<code>r=7;</code>			
2	<code>objem=2/3*pi*r*r*r;</code>			
	zkontrolujObjem(objem)	ok	Očekáváme: 1436.76... Dostali jsme: 718.38... chybicka-se vloudila	✗
	zkontrolujPolomer(r)	ok	ok	✓

Obrázek 3: Příklad otázky Matlab skriptu s provedenými testovacími případy.

3.4 Matlab graf

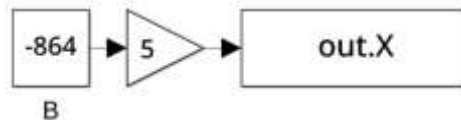
Student má vytvořit skript, který vygeneruje hodnoty x , vypočítá jejich $\sin(x)$ a vykreslí je ve správně naformátovaném grafu dle vzoru. K otázce je přiložená funkce, která si načte poslední vykreslený graf a kontroluje jeho požadované vlastnosti.

<pre> 1 x = linspace(-360, 360, 25); 2 y_sin = sin(deg2rad(x)); 3 plot(x, y_sin, "r*"); 4 title("sin") 5 xlabel("x [°]") 6 ylabel("y [-]") 7 grid on </pre>	<table border="1"> <thead> <tr> <th>Test</th> <th>Expected</th> <th>Got</th> <th></th> </tr> </thead> <tbody> <tr> <td>zkontrolujVektorx(x);</td> <td>ok</td> <td>ok</td> <td>✓</td> </tr> <tr> <td>zkontrolujVektory_sin(y_sin);</td> <td>ok</td> <td>ok</td> <td>✓</td> </tr> <tr> <td>zkontrolujGraf();</td> <td>graf ok</td> <td>Chybná barva čáry</td> <td>✗</td> </tr> </tbody> </table>	Test	Expected	Got		zkontrolujVektorx(x);	ok	ok	✓	zkontrolujVektory_sin(y_sin);	ok	ok	✓	zkontrolujGraf();	graf ok	Chybná barva čáry	✗
Test	Expected	Got															
zkontrolujVektorx(x);	ok	ok	✓														
zkontrolujVektory_sin(y_sin);	ok	ok	✓														
zkontrolujGraf();	graf ok	Chybná barva čáry	✗														

Obrázek 4: Příklad otázky Matlab grafu s provedenými testovacími případy.

3.5 Simulink

Student má vytvořit Simulinkový model, který obsahuje konstantu, která je vynásobena zadanou hodnotou a výsledek je uložen do bloku To Workspace. Soubor s modelem se přikládá při odevzdání řešení. Přiložená funkce v Matlabu načte odevzdaný model, nastaví hodnotu konstantě, spustí simulaci a vyhodnotí výsledek získaný přes blok To Workspace.



Test	Expected	Got	
soubor = simulinkZkontrolujNahranySoubor(uploaded_file); zkontrolujNasobeni(soubor);	ok	Máte špatný název bloků! Chtěli jsme nastavit blok - A	✗

Obrázek 5: Příklad otázky Simulinkového modelu s provedenými testovacími případy.

4 Poznátky

Plugin CodeRunner používáme již dva roky pro procvičování, domácí úlohy a testování studentů. Z pohledu vyučujících nám šetří velké množství času, které můžeme věnovat např. vytváření lepších úloh. Studenti jsou zároveň testováni více prakticky, což by mělo odpovídat tomu, co budou potřebovat v zaměstnání. Pro studenty má tento systém také výhody – je objektivní a mají okamžitou zpětnou vazbu, což jim umožňuje program upravit, pokud něco nefunguje dle zadání (neodevzdávají naslepo). Postupem času přicházíme na nové poznatky a lepší řešení jednotlivých typů otázek. Některé jsou popsány dále.

Neosvědčilo se nám používat penalizaci za vícenásobné odevzdání řešení. Tedy že odevzdání/kontrola kódu vícekrát než jednou, snižuje maximální počet bodů, který lze za otázku získat. Studenti většinou naprogramují první verzi v IDE, poté ji zkopírují do Moodle a nechají ji zkontrolovat. Pokud jejich řešení neprojde všemi testovacími případy, tak se ne vždy vracejí do IDE, ale spíše kód upravují přímo v editoru Moodle. S tím souvisí i častý problém, kdy musí výstup svého programu odladit tak, aby odpovídal tomu, který je definovaný v otázce. Často v tom dělají chyby, překlepy apod. Dává tedy smysl, aby si mohli svůj kód kontrolovat vícekrát a bez postihu.

Po kontrole kódu je možné nastavit, jestli se mají zobrazit všechny testovací případy, nebo jen některé a případně žádné. Lze tím omezit situaci, kdy student ví, jaké všechny vstupy se budou testovat a může obejít vyžadované řešení. V praxi jsme toto chování v naprosté většině případů

nezaznamenali. Pokud jsou některé testovací případy schované, tak dochází k situaci, kdy student dostane výsledek, že je něco špatně, ale už neví co mu nefunguje.

Je dobré dbát na to, aby testovací prostředí bylo co nejpodobnější tomu výukovému. Validační server provozujeme na Linuxu, pro C++ s překladačem GCC, ale výuka probíhá na Windows ve Visual Studiu. Překladače GCC a MSVC se v některých případech chovají odlišně a stává se, že studentům kód funguje ve Visual Studiu, ale nefunguje v Moodle. Jelikož se většinou jedná o začátečníky, tak se s touto situací nedokáží sami vypořádat.

Při testech hraje roli doba, kdy studenti čekají na kontrolou svého kódu. Validační server provozujeme ve virtuálním stroji s 24 CPU (Epyc 74F3) a 32 GB RAM. Kontrola jednoduchých C++ úloh trvá jednotky sekund. Úlohy v Matlabu se kontrolují jednotky až desítky sekund, protože se pokaždé znovu spouští Matlab. Simulinkové úlohy jsou nejhorší a většinou se zkontrolují do minuty, opět je to dáno tím, že se spouští Matlab a k němu ještě Simulink. CodeRunner v základním režimu spouští každý testovací případ samostatně. Pro každý tedy dojde k novému přeložení programu, spuštění Matlabu atd. Umožňuje ale i režim, kdy se všechny testovací případy sloučí do jednoho, což výrazně zkracuje dobu kontroly. Tento režim ale nefunguje, pokud se používá vstup na konzoli programu.

Velkou výhodou je, že pokud student objeví chybu v otázce, tak ji může vyučující okamžitě opravit a oprava se hned projeví u všech studentů.

Závěr

Článek představil plugin CodeRunner pro e-learningový systém Moodle, který používáme pro vyhodnocování otázek z programování na našich předmětech. Plugin je přizpůsobitelný ve velkém rozsahu, což znamená, že ho lze použít na velkou škálu programovacích jazyků a různé typy úloh. Byly uvedeny základní typy úloh, které používáme pro C++, Matlab a Simulink. Jedná se o velice užitečný nástroj, ale je třeba dbát na to, aby se nestalo, že studenti ve výsledku více bojují s validačním systémem, než aby programovali řešení úlohy.

Poděkování

Tento článek vznikl za podpory projektu SGS-2021-011: Rozvoj technik snižování řádu systému v elektrotechnických aplikacích.

Literatura

Moodle [online], [2023-08-01]. Dostupné z: <https://moodle.org>

CodeRunner [online], [2023-08-01]. Dostupné z: <https://coderunner.org.nz>

Jobe, GitHub [online], [2023-08-01]. Dostupné z: <https://github.com/trampgeek/job>

Kontaktní údaje

Ing. Martin Vítek

Katedra elektrotechniky a počítačového modelování

Fakulta elektrotechnická, Západočeská univerzita v Plzni

Univerzitní 26, 301 00 Plzeň

martinv@fel.zcu.cz

INOVATIVNÍ AKTIVITY VE VÝUCE IT NA TECHNICKÉ UNIVERZITĚ V LIBERCI

INNOVATIVE ACTIVITIES IN COMPUTER SCIENCE EDUCATION AT THE TECHNICAL UNIVERSITY OF LIBEREC

Jana Vitvarová

Abstrakt

V příspěvku budou představeny různé aktivity, které byly v posledních letech realizovány v rámci informatických předmětů na Fakultě mechatroniky, informatiky a mezioborových studií a na Fakultě zdravotnických studií Technické univerzity v Liberci. Tyto aktivity měly za cíl předávat studentům nové znalosti prostřednictvím zážitkových metod, umožnit jim pracovat na projektech pro skutečné zákazníky a podpořit je v přemýšlení nejen nad technickou stránkou, ale i nad užitečností a potenciálem navržených řešení. Studenti se rovněž učili práci v multioborových týmech a důležitosti učení se od druhých, a naopak sdílení svých znalostí a dovedností s ostatními.

Klíčová slova: inovativní aktivity ve vzdělávání v oblasti IT, zážitková pedagogika, projekty z praxe, sdílení znalostí.

Abstract

The paper presents activities that have taken place in recent years within the informatics courses at the Faculty of Mechatronics, Informatics, and Interdisciplinary Studies, as well as the Faculty of Health Studies at the Technical University of Liberec. These activities aimed to facilitate the transfer of knowledge through experiential learning, provide students with opportunities to work on real-world projects and encourage them to consider not only the technical aspects but also the benefits and potential of the proposed solutions. Furthermore, the students experienced multidisciplinary teamwork and were motivated to learn from the more experienced and share their own knowledge.

Keywords: innovative activities in IT education, experiential learning, real-world projects, knowledge sharing

Úvod

V textu představíme dva netradiční projekty, které realizujeme v rámci výuky IT na Technické univerzitě v Liberci. Projekty slouží k oživení výuky a podpoře získávání tvrdých i měkkých dovedností s využitím multioborového a multigeneračního prostředí.

1 TULTech workshopy

1.1 Důvod vzniku

TULTech workshopy vznikly jako reakce na prosbu paní učitelky ze základní školy. Požádala rodiče, aby přišli povyprávět o svém zaměstnání. Ve škole již byla maminka, která pracuje jako zdravotní sestřička a dětem povídala o své práci v nemocnici. Podobně jako u první pomoci, tak i u programování je lepší méně mluvit a radši rovnou zkoušet. Jako rodič, který pracuje v oboru IT, jsem proto pozvala

celou třídu na naši univerzitu a připravila pro ně workshop. Během něj jsme spolu s kolegy představili, co na univerzitě děláme, a zároveň jsme děti učili základy programování.

1.2 První workshop

Tématem našeho prvního workshopu se stala bionická ruka. Kolegové na univerzitě experimentují s různými materiály a mechanismy v této oblasti. Workshop začal příběhem mladé dívky Tilly Lockey⁹, která v dětství po prodělání meningitidy přišla o obě ruce. Dnes je sebevědomou influencerkou a modelkou propagující futuristické bionické ruce, díky nimž zvládne i takové věci jako líčení. Kolega představil různé prototypy částí rukou a materiálů. Na všechno si děti mohly sáhnout. Poté jsme se pustili v blokovém jazyku Scratch do programování ovládání bionické ruky včetně takzvaného “health check”, kterým se všechny úchopy ruky kontrolují. Věděla jsem, že nemůžu učit programovat třicet dětí sama. Proto jsem tuto část workshopu realizovala společně se svými studenty během cvičení předmětu Algoritmizace a programování. Studenti měli za sebou již dva měsíce výuky, a tak byli už všichni schopni své znalosti předat dál. Učení druhých je nejlepší formou, jak si ověřit vlastní porozumění. Vyšlo to skvěle - na jednoho studenta připadli vždy dva žáci. Žáci vzhlíželi ke svým novým učitelům, a i slabší studenti získali sebevědomí, že už něco svedou. Workshop jsme zakončili využitím Makey Makey¹⁰, což je malé zařízení, pomocí něhož můžeme vytvořit touchpad nebo klávesnici téměř z čehokoliv, co alespoň trochu vede elektrický proud - banán, člověk, mince, sklenice s vodou. Žáci propojili svůj program s papírově-aluminiovým modelem hodinek, které umožňují postiženému člověku provést změnu úchopů. Podobné workshopy jsme pak realizovali víckrát a pojmenovali jsme je TULTech workshopy. Byly na různá témata, ale vždy měly stejnou strukturu: teorie s ukázkami reálných projektů na TUL, navazující programování v jazyce Scratch a propojení s fyzickým světem (physical computing). Zvažovali jsme i programovací jazyk Python pro starší děti, ale velkou výhodou jazyka Scratch je jeho grafické uživatelské rozhraní, do kterého lze rychle cokoli přidat nebo stáhnout.

1.3 Rozšířená verze

Díky projektu Nadace Preciosa jsme do workshopů zapojili i studenty biomedicínského inženýrství (BI), kteří připravili medicínskou teoreticko-praktickou část a studenti IT pak realizovali návaznou počítačovou část. Jedno z témat byla například elektřina v lidském těle. Děti nejdřív objevovaly princip a způsob měření EKG, učily se rozpoznat základní tvar křivky. Následně určily hodnoty této křivky, uložily je do pole a ve Scratchi křivku na pozadí obrázku skutečného EKG přístroje vykreslily. Workshopy zapadaly do obou předmětů. Netradiční formou umožnily IT studentům zopakovat si a aplikovat na jednoduchých příkladech základní principy programování. Studentům BI pro změnu pomáhaly se lépe připravit na svou mezioborovou roli, která zahrnuje i vysvětlování odborných věcí srozumitelným způsobem lidem, kteří se v oboru nepohybují. Během workshopů objevíme i studenty, které baví učit a nabízíme jim pak spolupráci v dalších aktivitách pro veřejnost. Některé z nich jsme inspirovali i k založení vlastního kroužku programování pro děti.

2 Semestrální multioborový inovační projekt

2.1 Motivace

Ideathony a hackathony jsou známé v IT světě jako nástroje podpory inovačního procesu. Jsou to obvykle 24 hodinové soutěže, při nichž týmy vývojářů, designérů a dalších odborníků intenzivně a společně pracují na zadaném problému. Pokud je výsledkem rozpracovaný nápad nebo myšlenka, mluvíme o ideathonu. Pokud je to prototyp softwarového nebo hardwarového produktu, jde o hackathon.

⁹ How I lost my hands | My survival story: <https://www.youtube.com/watch?v=5LaVx7Le7E8>

¹⁰ <https://makeymakey.com/>

Akce bývá zakončena prezentací výsledků, která zahrnuje i představení ekonomického a tržního potenciálu daného řešení.

Účast studentů na podobných akcích pro ně přináší mnoho výhod. Především rozvíjí kreativní myšlení, což je dovednost, která je často opomíjená ve výuce zejména na technických školách. Dále podporuje týmovou spolupráci, což je opět aspekt, kterému není věnována taková pozornost ve školním prostředí, kde studenti často pracují a jsou hodnoceni individuálně. Situace, kdy mají studenti před sebou problém, který chtějí vyřešit a k dispozici zkušené mentory je ideální motivací pro učení se nových věcí. Přispívá k tomu i fakt, že zadané problémy vycházejí z potřeb skutečných zákazníků a studenti tak cítí větší smysl své práce. V neposlední řadě si zejména technicky zaměřeni studenti uvědomí, že pokud by vyvíjeli komerční produkt, nestačí k tomu jen dobře zpracované průmyslové řešení.

V Libereckém kraji organizují ideathony¹¹ a hackathony pro studenty nejčastěji organizace libereckého inovačního ekosystému jako Agentura regionálního rozvoje, CzechInvest nebo Liberecký podnikatelský inkubátor Lipo.ink. Jako univerzita, která akce mezi studenty propaguje, jsme si všimli, že studenti váhají s účastí. Důvody mohou být různé, např. obava z nedostatku znalostí a přílišné náročnosti, strach z neznámého, neschopnost najít si tým, nedostatečná motivace apod. Na druhou stranu je počet míst na každé akci omezený, což znamená, že naopak při velkém zájmu se nebudou moci zúčastnit všichni uchazeči. Do soutěže přicházejí většinou již vytvořené týmy kamarádů, které ale bývají jednooborové.

2.2 Cíl projektu

Abychom víc podpořili tento typ aktivit, navrhli a uskutečnili jsme inovační projekt, který trval celý semestr a byl přímo zaintegrovaný do výuky, a tak dostupný všem studentům. Multioborovost jsme docílili zapojením studentů dvou předmětů ze dvou fakult: studentů 2. ročníku oboru IT v předmětu Databázové systémy (dále studenti IT) a 5. ročníku oboru Biomedicínské inženýrství v předmětu Lékařské systémy (dále studenti BI). Pilotní projekt se uskutečnil v zimním semestru v roce 2021. Zúčastnilo se jej 9 studentů BI a 26 studentů IT. Zároveň jsme chtěli zachovat propojení na praxi, proto jsme požádali o spolupráci Liberecký podnikatelský inkubátor (dále LipoInk) a místní firmy.

2.3 Přípravná fáze: Hledání zadavatelů, výzev a mentorů

V přípravné fázi bylo potřeba najít vhodné zadavatele a výzvy. S ohledem na zaměření předmětů jsme jako zastřešující téma vybrali zdravotnictví. Oslovili jsme zdravotnická zařízení, organizace působící ve zdravotnictví i firmy vyvíjející zdravotnický zaměřený software. Získali jsme různorodé výzvy týkající se např. zdravotnických odpadů, dalšího vzdělávání lékařských a nelékařských povolání, nebo plánování a sdílení znalostí při ortopedických operacích. Je vhodné najít zadavatele, kteří mají skutečný zájem o řešení své výzvy a jsou ochotni se projektu alespoň online ve vybraných časech aktivně účastnit. Případně je možné hledat témata, která se dotýkají samotných studentů. Ti se pak dokážou lépe vcítit do potencionálních zákazníků. Dále bylo nutné získat mentory se zkušenostmi v oblasti zdravotnictví, businessu a technické odborníky z univerzitního i firemního prostředí. Osvědčilo se nám oslovit i bývalé studenty, kteří ochotně nabídli své odborné znalosti. Mentori ocenili možnost zapojit se do projektu i hybridně - online nebo offline.

2.4 Realizační fáze

Protože projekt probíhal napříč dvěma předměty a bylo nutné dodržet předepsané sylaby, byl zvolen harmonogram zaintegrovaný do 14. týdenního semestru, který kombinoval samostatné i společné aktivity v obou předmětech a obsahoval povinné i nepovinné části. Pro kolaborativní a kreativní aktivity je vhodné využít i prostory mimo školu. V našem případě jsme pro většinu aktivit zvolili zázemí LipoInku. Jejich prostory disponují modulárním nábytkem, ze kterého lze vytvářet různé komfortní

¹¹ Liberec Ideathon 2023: <https://1012plus.cz/cs/projekt/liberec-ideathon-2023>

seskupení pro týmovou i samostatnou práci a zároveň tam na studenty inspirativně působí startupové prostředí. Projekt byl rozdělen do 6 aktivit, jak ukazuje tabulka 1.

Tabulka 1: Harmonogram projektu

Týden	Aktivita	Způsob	Místo
1. týden	Aktivita 1: Lego4Scrum - projektové řízení	samostatně BI, IT	TUL
2. týden	Aktivita 2: Ideathon	BI	LipoInk
4. týden	Aktivita 3: GetTogether	BI + IT	LipoInk
8. týden	Aktivita 4: Hackathon (nepovinné)	IT (+ BI)	LipoInk
12. týden	Aktivita 5: Prezentace - pitch	BI + IT	LipoInk
14. týden	Aktivita 6: Prezentace technických řešení	IT	TUL

2.5 Aktivita 1: Lego4Scrum

V rámci této aktivity se studenti nejdříve seznámili s principy agilního projektového řízení a vývoje a následně si jej v rámci interaktivní hry Lego4Scrum prožili. Lego4Scrum (Krivitsky, 2019) je simulace používaná k výuce a tréninku agilního frameworku Scrum s využitím stavebnice LEGO. V průběhu simulace studenti tvoří LEGO produkt, který odpovídá požadavkům zákazníka (tzv. product ownera). V případě studentů BI se například jednalo o stavbu nového nemocničního zařízení. Zákazníka (náročného) hraje učitel, který postupně přináší do stavby různé často nepředvídatelné události jako “úraz člena stavebního týmu”, požadavek na další oddělení apod. Studenti se při hře musí řídit pravidly Scrumu, jako např. nejvyšší prioritou je uspokojit zákazníka, proto je potřeba se zákazníkem intenzivně komunikovat a být připraven adaptovat se na i na změny jeho požadavků. Dále pak, že produkt se vyvíjí v krátkých iteracích (tzv. sprintech). Po každé z nich následuje zpětná vazba od zákazníka a retrospektiva, na základě které by studenti měli přehodnotit svůj způsob práce. Během simulace se studenti učí, jak efektivně komunikovat a plánovat práci v samoorganizujícím se týmu a jak dosáhnout vysoké produktivity a kvality vývoje. Zmíněné zručnosti se jim hodí v dalších fázích projektu. Se simulací Lego4Scrum máme na TUL už dlouholeté zkušenosti. Její velká výhoda je, že se dá přizpůsobit délkou (do jaké hloubky se půjde) i počtem studentů. Otestovali jsme ji se studenty technických i netechnických oborů, s dětmi ze základních škol, i s profesory na zahraniční univerzitě.

2.6 Aktivita 2: Ideathon

Navazující aktivitou je ideathon. Hlavním cílem Ideathonu je přijít s novými nápady nebo neobvyklými řešeními daného problému nebo výzvy, které by mohly být později rozvíjeny do reálných projektů nebo podnikatelských iniciativ. Důraz je kladen na tvořivost řešení tedy originalitu, ale zároveň i užitečnost. Ve škole se většinou nevyžadují netradiční řešení a ani se studenti nezamýšlí nad prospěšností toho, co vytvoří. Na začátku aktivity zadavatelé představili studentům výzvy. Následovala diskuze a rozdělení do týmů spolu s výběrem výzvy. Na jedné výzvě mohlo pracovat i víc týmů. Pak už probíhala samotná identifikace konkrétního problému a hledání inovativního řešení. Účastníci bývají často na začátku ztraceni. V rámci výzvy se snaží řešit velký, příliš všeobecný problém, místo aby identifikovali jeden uchopitelný konkrétní problém. Spokojí se s jedním z prvních řešení, které není nijak inovativní. Bojí se vystoupit ze zajetých kolejí. Proto je vhodné ukázat jim techniky jako design thinking, principy brainstormingu a dalších metod pro podporu generování nápadů. Aktivita končila krátkou prezentací (1 min) každého týmu. Studenti zde představili svůj nápad/produkt formou 5W (Who, Why, What, How, Why, When) jako odpověď na následující typy otázek: Kdo jsou Vaši zákazníci? Jaký mají problém? Co je Váš nápad/produkt? Jak jim Váš nápad/produkt daný problém řeší?

2.7 Aktivita 3: GetTogether

Po Ideathonu v rámci hodin studenti BI provedli rešerši a pracovali na IT analýze. Učili se extrahovat a specifikovat požadavky, vytvářet use case diagramy, definovat priority a určit MVP (Minimal Viable Product) pro prototypování. Následně kreslili prototyp uživatelského rozhraní na papír nebo v nástroji Figma. Komunikovali online se zadavateli, aby si práci validovali. Na společné aktivitě GetTogether pak odprezentovali svoje řešení studentům IT. Následovaly kulaté stoly, kde vždy seděl jeden zadávající tým a studenti IT se mohli postupně doptávat na podrobnosti a zvolit si, ke kterému týmu a teda i řešení se připojí. Multioborový tým si pak nasdílel dokument s IT analýzou, nad kterou se pak spolu bavili a vylepšovali ji. Studenti BI se naučili svou odbornost podat způsobem blízkým vývojářům a studenti IT se naučil doptávat na odborné detaily. Zvolili jsme práci v multioborovém týmu, tj. snažili jsme se ukázat, že každý v týmu může mít podle odbornosti svoji roli a dohromady pak tým dokáže řešit problém komplexněji z odborného i technického hlediska. Je vhodné připravit pár teambuildingových her, aby se studenti měli šanci lépe poznat a to jim usnadnilo vzájemnou komunikaci.

2.8 Aktivita 4: Hackathon

Další aktivitou byl Hackathon ve formě osmihodinové intenzivní práce na technické stránce zvoleného řešení. Tato aktivita byla nepovinná, protože není oficiálně v sylabu. Ten ale zároveň obsahuje hodiny pro vypracování semestrální práce. Studenti IT si tedy mohli zvolit, jestli budou pracovat samostatně, nebo využijí hackathon, kde měli k dispozici učitele předmětu, online i offline konzultační schůzky s mentory i zadavateli, studenty BI na telefonu, pohodlné prostředí LipoInku, drobné občerstvení a pizzu. Zúčastnilo se 92 % studentů.

2.9 Aktivita 5: Presentace - pitch

Konec hackathonu patřil workshopu o tom, jak si připravit dobrou prezentaci (tzv. pitch), která přesvědčí zadavatele/investory o užitečnosti a businessovém potenciálu řešení. Osvědčilo se nám navázat na popis, co má prezentace obsahovat i skutečnou ukázkou. Pro techniky je většinou tato forma prezentace dost překvapivá, protože jsou zvyklí o své práci mluvit hlavně z funkčního a technického hlediska. Musí se umět překlopot z otázek typu “co to dělá”, na otázky “co to přinese”, nebo “vyplatí se to”. Zde studenti narazili i nato, že by se jim do týmu hodil ještě někdo s ekonomickými znalostmi.

2.10 Aktivita 6: Presentace technických řešení

Tato aktivita se týkala pouze studentů IT. V rámci zapojeného předmětu Databázové systémy se učili během semestru navrhovat datovou vrstvu, pracovat s daty a tvořit databázové aplikace. Protože byl projekt součástí předmětu, kontroloval vyučující i samotné technické řešení, které se pak započítalo do zápočtu.

2.11 Výstupy

V projektu získali studenti novou širokospektrální zkušenost a zároveň aplikovali nabyté znalosti z daných předmětů. V některých případech pokračovali na projektu v dalších univerzitních pracích. Stalo se tak i u prototypu na evidenci zdravotnického odpadu. V rámci bakalářské práce a TAČR projektu se z něj stala plnohodnotná aplikace¹², kterou mohou nyní využít menší zdravotnická zařízení nebo provozovny na vytváření podkladů pro průběžnou evidenci odpadů. Někteří studenti využili nově navázaných kontaktů a domluvili si další spolupráci. Pro jiné byl projekt inspirací a povzbuzením k účasti na podobných akcích i mimo univerzitu. V rámci TUL na projekt plynule navazuje soutěž Start-up TUL¹³, jejímž cílem je kreativním způsobem přiblížit studentům možnosti založení vlastní firmy

¹² Aplikace Jispen: <https://jispen.inisoft.cz/>

¹³ Studentská soutěž Start-up TUL: <https://sbc-tul.cz/soutez/>

nebo řízení vlastního projektu. V zimním semestru 2023 se chystáme projekt rozšířit a zapojit do něj i studenty Ekonomické fakulty, kteří obohatí týmy právě o ekonomickou expertízu.

Závěr

V tomto článku byly představeny dva netradiční projekty využívané na Technické univerzitě v Liberci. TULTech workshopy přibližují akademické prostředí a jeho výzkum světu základního vzdělávání. Semestrální inovační projekt zas propojuje multioborový svět univerzity s praxí a budoucností. Oba podporují zvědavost a tvořivost napříč obory a generacemi.

Literatura

[1] Krivitsky, A. (2019). *lego4scrum: A complete guide. A great way to teach the Scrum framework and Agile thinking*. ISBN 978-1520729169.

Kontaktní údaje

Ing. Jana Vitvarová, PhD.

Fakulta mechatroniky, informatiky a mezioborových studií (FM)

Technická univerzita v Liberci (TUL)

Studentská 1402/2, 461 17 Liberec

e-mail: jana.vitvarova@tul.cz

ROZŠÍŘENÉ POJETÍ VÝUKY DEVOPS NA FIT ČVUT

COMPREHENSIVE CONCEPT OF DEVOPS TEACHING AT FIT CTU

Tomáš Vondra

Abstrakt

DevOps inženýrství je poměrně nová disciplína nepříliš pokrytá vysokým školstvím. Firmy zavádějí interní školicí střediska, aby nedostatek absolventů suplovaly. Jde o široké spektrum postupů a technologií od agilního vývoje softwaru, přes testování a bezpečnost, sestavení a release management, až po automatizaci nasazení a monitoringu aplikací. Na FIT ČVUT jsme proto spojili síly kateder softwarového inženýrství a počítačových systémů a expertů z praxe a vytvořili bakalářský předmět navazující na jedné straně na Git a na druhé na Operační systémy, jehož absolventi si na interních cloudových prostředcích prakticky vyzkouší základy všech technologií a sestaví si pipeline kontinuální integrace od kompilace po nasazení.

Klíčová slova: DevOps, výuka, git, kontinuální integrace, docker, cloud

Abstract

DevOps is a relatively new discipline not covered well by higher education. Companies resort to internal training to compensate for the lack of graduates. It is a spectrum of processes and technologies from agile development, testing and security, build and release management, to automation of deployment and application monitoring. At FIT CTU, the departments of Software Engineering and Computer Systems and experts from practice created a bachelor's course with prerequisites of Git and Operating Systems, whose students practically try out the basics of all technologies on internal cloud resources and a continuous integration pipeline comprehensively from compilation to deployment.

Keywords: DevOps, teaching, git, continuous integration, docker, cloud

Úvod

Název DevOps se sestává ze zkratk Dev pro vývoj softwaru a Ops pro operations čili provoz počítačových systémů. Proto bylo třeba na FIT ČVUT navázat úzkou spoluprací mezi Katedrou softwarového inženýrství, která se zabývá návrhem a vývojem software, a Katedrou počítačových systémů, která zkoumá architektury a správu počítačů.

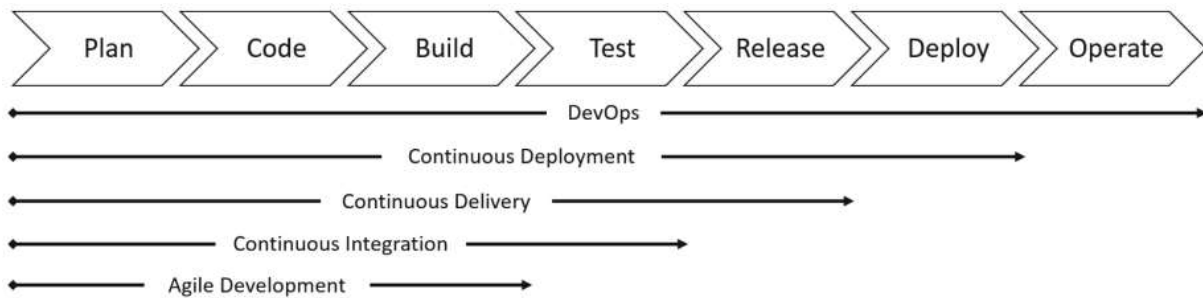
Ve skutečnosti má DevOps tři aspekty, které se navzájem propojují a podporují. Jsou jimi: (Jones, 2018)

1. Moderní kultura vývoje, tedy agilní vývoj, evidence lístků s prací (tickets), verzování a recenzování kódu.
2. Sestavovací (build) engineering, kvalita kódu, automatické testy a CI/CD (Continuous Integration/Delivery|Deployment).
3. Infrastruktura jako kód (IaC) a automatizace konfigurace serverů.

Společná pro celý DevOps je snaha o automatizaci, která zabezpečí opakovatelné procesy a díky nim lepší kontinuitu provozu a jednotnost prostředí pro vývoj, testování a produkci. Cílem je zefektivnit práci, zkrátit dobu mezi napsáním kódu a jeho nasazením do produkce (což

podporuje snahy agilního vývoje) a předcházet lidským chybám. K veškerým automatizačním skriptům se chováme taktéž jako ke kódu, čímž se přibližuje náplň práce administrátorů k vývojářům.

Primitivní představy manažerů, že DevOps znamená, že vývojáři se stanou také administrátory, se ve většině případů nevyplnily nebo vyústily ve snížení dostupnosti, protože vývojáři většinou nemají vztah k pozdějším fázím životního cyklu software (Obr. 1). Lepší představa je, že DevOps se promítá do práce více profesí, které se na tomto cyklu podílejí. Filosofie DevOps nestojí jen na technologiích, ale hlavně na užší spolupráci.



Obrázek 1: DevOps pokrývá životní cyklus software. Zdroj: (Jones, 2018)

Z původní profese DevOps inženýra se postupně oddělily různé větve podle toho, kde v životním cyklu aplikace je těžiště jejich znalostí:

- SCRUM Master a Product Owner jsou profese vyplývající z agilního vývoje a spadají do fáze Plan. Často mají DevOps inženýry v rámci vývojového týmu a musí tudíž jejich práci rozumět.
- Build and Release inženýr je člověk, který ovládá nástroje pro kontinuální integraci (CI) a je schopen pro aplikace sestavit a spravovat pipeline (posloupnost kroků, kde výstup jednoho je vstupem následujícího, a selhání kroku způsobí zastavení procesu). Inzeráty hledající lidi na DevOps nejčastěji míní tuto profesi.
- DevSecOps inženýr je člověk, který má základy mimo vývoje software také v počítačové bezpečnosti. Pracuje vedle klasického testera ve fázi Test a má za úkol doplnit do pipeline kroky jako statická analýza kódu a detekce zranitelností, které zaručí kvalitu kódu z pohledu bezpečnosti ještě dříve, než se dostane do produkce.
- SRE (System Reliability Engineer) je člověk, který má základy hlavně v administraci, ale je schopen efektivně používat cloud pro zvýšení dostupnosti a výkonu aplikací a psát automatizační skripty IaC (Infrastructure as Code). Také je zodpovědný za monitoring. Pracuje ve fázích Deploy a Operate.
- FinOps inženýr je člověk, který rozumí nákladové struktuře cloudu, je zodpovědný za přeučtování spotřeby zdrojů na jednotlivé aplikace v rámci firmy a je schopen odhadnout finanční náklady na provoz aplikace již během vývoje. Pracuje ve fázích Operate a Plan.

Z chování firem je zřejmé, že české vysoké školy neprodukují dostatek absolventů se znalostmi vhodnými pro zaplnění těchto rolí. Pracovní místa pro seniory jsou velmi dobře placená a firmy často najímají juniory vzdělané v softwarovém inženýrství s tím, že je sami do DevOps zaučí, jako příklad může sloužit (LinkedIn, 2023).

1 DevOps v rámci studijního programu Informatika

Vzhledem k tomu, že DevOps je velmi široká disciplína, rozhodli jsme se ji zařadit ve formě přehledového předmětu Úvod do DevOps již do třetího semestru bakalářské etapy studia a včlenit jej jako povinný pro několik specializací. Cílem je naučit studenty základní znalosti z oboru, vytvořit u nich povědomí o obsahu výše zmíněných profesí a pomocí praktických cvičení je naučit aplikovat vybrané nástroje, až na ně v praxi narazí.

Prerekvizitami v 1. ročníku společném pro všechny specializace jsou:

- Předmět Unixové operační systémy, ve kterém se učí práce v příkazové řádce a, který byl prezentován na této konferenci (Zemánek a Muzikář, 2022).
- Předmět Technologie pro vývoj SW se zkratkou BI-GIT, kde se studenti naučí práci s verzovacím nástrojem Git jak na lokálních, tak na vzdálených repozitářích. Předmět končí tématem použití větví, na které náš předmět navazuje.
- Počítačové sítě, ze kterých jsou potřeba znalosti jako co je to IP adresa, port TCP nebo UDP a doménové jméno.

Nepříjemností je, že v rámci předmětů Programování a algoritmizace 1 a 2 studenti v 1. ročníku pracují pouze s jazykem C a C++, který se typicky nepoužívá pro tvorbu informačních systémů a webových portálů. Zvolili jsme proto pro demonstrační program jazyk Java, který mají povinný alespoň studenti oboru Softwarové inženýrství (Bílá kniha, 2021a) v souběžném předmětu Technologie Java.

Dalšími obory, pro který je předmět DevOps povinný, jsou Webové inženýrství (Bílá kniha, 2021b), kde si studenti mohou vybrat jazyk PHP nebo Python, a obor Počítačové systémy a virtualizace (Bílá kniha, 2021c), který je zaměřený na administraci a další jazyky povinně neobsahuje.

Nicméně časným zařazením předmětu si otevíráme možnost na jednotlivá témata navazovat v pozdějších předmětech různých specializací. Pro první dva obory je to klíčový předmět Softwarové inženýrství, který se zabývá primárně návrhem software, ale také testováním, řízením konfigurací, projektovým řízením a agilním vývojem. Prakticky se mohou u softwarových inženýrů znalosti z DevOps promítnout v předmětech Softwarový týmový projekt 1 a 2.

V oboru Počítačové systémy a virtualizace navazuje předmět Virtualizace a datová centra, kde se studenti dozvědí více o infrastruktuře uvnitř cloudu, se kterým se poprvé setkali v předmětu o DevOps. V předmětu Správa sítí a služeb si mohou vzpomenout na úvod do certifikátů a certifikačních autorit, který děláme pro všechny obory v rámci DevOps. Konečně, v předmětu Administrace webového a DB serveru společným s oborem Webové inženýrství se zase navazuje na webové servery, reverzní proxy a vyvažovače zátěže, které jsou nutně součástí nasazení aplikace v cloudu.

2 Struktura předmětu a pokryté technologie

V rámci předmětu jsme museli skloubit požadavky dvou skupin oborů. Ty softwarově inženýrské potřebovaly zejména, aby si studenti na cvičení vyzkoušeli připravit automatizované nasazení nějaké vzorové webové aplikace. V rámci CI by se měla aplikace nejprve sestavit, spustit automatické testy, statickou analýzu kódu a po úspěšném provedení těchto kroků aplikaci nasadit. Měli by mít možnost vyzkoušet správu nastavení pro různá prostředí (DEV,

TEST, PROD). Do produkce by se nemělo nasazovat zcela automaticky bez manuálního potvrzení. Celý proces by měl být řízen posláním kódu do odpovídající větve v Gitu.

Systémový obor pak potřeboval pro studenty udělat úvod do kontejnerizace a naučit je aplikaci zabalit do formátu Docker. Studenti musí umět použít cloudy dvou vrstev, IaaS (Infrastructure as a Service) a PaaS (Platform as a Service), a to ručním i automatizovaným způsobem. Měli by být schopni připravit pro aplikaci cloudové prostředí, do kterého se bude aplikace pomocí pipeline CD automaticky nasazovat. V provozu je potřeba aplikaci monitorovat a dodávat doporučení ohledně potřebných prostředků, nebo toto ponechat na automatickém škálování.

Výuku by ale nebylo vhodné rozdělovat do dvou předmětů, protože je třeba, aby pro užší spolupráci v rámci DevOps obě profese rozuměly náplni té druhé. Požadavkem kultury DevOps je, aby vývojáři byli zodpovědní za výkonnost a spolehlivost své aplikace v provozu v drželi za ni případné noční služby (což dříve bylo výhradní povinností provozního oddělení). Proto musí znát způsob nasazení aplikace na cloud. Naopak administrátoři by měli znát architekturu aplikace a její testovací pipeline, aby dokázali vývojářům poskytnout nutné nástroje pro CI a podporu v provozu.

Protože žádný vyučující by nedokázal kvalitně pokrýt celou škálu technologií, rozdělili jsme si přednášky na dvě poloviny. První má na starost Katedra softwarového inženýrství, druhou Katedra počítačových systémů. Předávacím bodem je fáze Release z životního cyklu. Přednášek je celkem 12. Se dvěma technologiemi (testování SW a IaC) jsme neměli dostatečné zkušenosti, a proto jsme oslovili externí kolegy z praxe, kteří předmětu dodali svůj pohled na věc a zapojili se do výuky i jako cvičící.

Praktických cvičení je během semestru také 12, takže můžeme jejich obsahem navazovat na přednášky. Jejich autorství je také rozděleno na poloviny, ale každý cvičící provází své studenty celým semestrem. Ke cvičením jsou připraveny podrobné návody, aby podle nich mohl cvičící studenty vést a aby oni pak případně mohli úlohy dodělat z domova, pokud by je během hodiny nestihli. Mimo cvičícího studenty semestrem provází také vzorová webová aplikace, pro kterou se krok za krokem sestavuje pipeline od kompilace po automatické nasazení.

Hodnocení předmětu je ze 60% založeno na praktické stránce, která se testuje dvěma praktickými testy (vždy po čtyřech cvičeních), kde mají studenti samostatně provést úlohy velmi podobné těm, co byly předváděny v rámci návodů na cvičení. Testuje se zde hlavně praktická schopnost technologie použít a základní pochopení tím, že jsou úlohy drobně obměněny. V běžné praxi DevOps se často kopíruje z dokumentace, i proto je při testech povoleno používat návody na cvičení a vlastní poznámky a kód ze cvičení.

Ve formě cvičení a hodnocení jsme se museli odchýlit od předchozí praxe, kde se DevOps často zařazoval jako magisterský volitelný předmět, kde pak logicky byl větší nárok na hlubší pochopení a návrh pipeline jako samostatnou či týmovou semestrální práci. Studentů ale byly jednotky a od předmětu odstupovali pro náročnost přípravy (Capozucca, 2018). U nás jsou jich stovky v povinném předmětu v bakalářské etapě studia.

Pokud jde o technologie použité v předmětu, snažíme se používat otevřené varianty. Máme za to, že není třeba žádat o grant nějakého veřejného provozovatele cloudu a propagovat jeho produkt. Z otevřeného projektu se v praxi dá snadno přejít na komerční varianty různých výrobců.

Stěžejní pro předmět je GitLab provozovaný naším oddělením ICT, který mimo správy verzí kódu má vestavěno i plánování práce pomocí tiketů a recenzování kódu. Dále má funkce CI, na kterých se staví v průběhu předmětu pipeline. Vzorová aplikace je postavena na javovém frameworku Spring Boot se sestavením pomocí Maven. V rámci testování jsou použity testy pomocí JUnit a Mockito. Obor DevSecOps zastupuje nástroj pro statickou analýzu SonarQube

s detekcí zranitelností DependencyCheck. Studentům jsou ukázány automatické testy uživatelského rozhraní pomocí nástroje Selenium.

Druhá část předmětu staví na OpenNebule, což je cloud IaaS stavějící virtuální stroje (VM) dodaný rovněž oddělením ICT. Studenti se ho naučí i skriptovat z příkazové řádky. Na VM se aplikace nasazuje pomocí nástroje automatické správy konfigurací Ansible, pro což je třeba umět manipulovat s klíči SSH. Aplikace dosud existující ve formátu JAR se zabalí do formátu kontejneru nástrojem Podman. GitLab zde figuruje jako nástroj pro CD a je využit jako registr kontejnerových obrazů. Dále se představí cloud PaaS OpenShift nasazený v otevřené verzi vyučujícími nad školním IaaS. Toto je cloud automaticky sestavující z aplikací kontejnerů na klastr Kubernetes. Tím se završí pipeline krokem Continuous Deployment. Až po druhém testu se ještě představí nástroj Terraform pro IaC a Prometheus pro monitoring cloudu PaaS. Studenti si vyzkouší zátěžový test a automatické škálování.

Závěr

První běh předmětu proběhl v zimním semestru 2022/23 a včetně anglické a kombinované varianty na něj bylo zapsáno 197 studentů, z toho 166 předmět dokončilo. U prezenční varianty bylo rozdělení známek přibližně normální, u anglické spíše posunuté k nižším stupňům a u kombinované s velkým podílem A u lidí s praxí a F u lidí s nedostatkem času na domácí přípravu. V příštím semestru očekáváme 317 studentů. S technickým zajištěním vlastními zdroji fakulty se nevyskytly větší potíže, jen příští semestr vyměníme cloud OpenNebula za OpenStack spravovaný vyučujícími z důvodu nedostatku kvalifikovaných správců v oddělení ICT. Drobné chyby v aplikaci nebo obsahu cvičení vyřešíme podle ohlasů cvičících a ze studentské ankety.

Literatura

- Bílá kniha (2021a). [online], [2023-07-23]. Doporučený průchod: Doporučený průchod: Bc. specializace Softwarové inženýrství. Dostupné z: <https://bk.fit.cvut.cz/cz/pruchody/pr1217952639605.html>
- Bílá kniha (2021b). [online], [2023-07-23]. Doporučený průchod: Doporučený průchod: Bc. specializace Webové inženýrství. Dostupné z: <https://bk.fit.cvut.cz/cz/pruchody/pr1216107633905.html>
- Bílá kniha (2021c). [online], [2023-07-23]. Doporučený průchod: Bc. specializace Počítačové systémy a virtualizace. Dostupné z: <https://bk.fit.cvut.cz/cz/pruchody/pr1217952878505.html>
- Capozucca, A. (2018). Design of a (Yet Another?) DevOps Course. In Bruel, J., Mazzara, M., Meyer, B. (ed.), Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment, First International Workshop. Chateau de Villebrumier, France: Springer, Lecture Notes in Computer Science, 1–18.
- Jones, C. (2018). A Proposal for Integrating DevOps into Software Engineering Curricula. In Bruel, J., Mazzara, M., Meyer, B. (ed.), Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment, First International Workshop. Chateau de Villebrumier, France: Springer, Lecture Notes in Computer Science, 33–47.
- LinkedIn (2023). [online], [2023-07-23]. DevOps Incubator - Komerční banka. Dostupné z: <https://www.linkedin.com/jobs/view/3652243273/>
- Zemánek, P., Muzikář, Z. (2022). Koncepce výuky předmětů z oblasti operačních systémů. In Konference Informatika 2022. VŠP Jihlava, 59-62.

Kontaktní údaje

Ing. Tomáš Vondra, Ph.D.

Katedra počítačových systémů

Thákurova 7, 160 00 Praha 6 – Dejvice

e-mail: tomas.vondra@fit.cvut.cz