

VÝUKA PYTHONU NA ÚBMI FEKT VUT

Kateřina Jurečková¹

¹Vysoké učení technické v Brně, Brno

ABSTRAKT

Předmět Algoritmizace a programování je vyučován v 1. ročníku bakalářského studia na ÚBMI, FEKT VUT. Jeho cílem je osvojení si programovacích návyků, seznámení s programovacím jazykem Python a řešení jednoduchých algoritmizačních úloh. Dále také předmět studenty seznamuje s verzovacím nástrojem Git, který je pak zejména klíčový při vybraných bodovaných aktivitách během semestru jako jsou např. domácí úkoly nebo půlsestrální test. V rámci nich je Git využíván pro vytváření revizí a odesílání revizí na platformu GitHub, kde následně probíhá ohodnocení daného úkolu pomocí automatických testů. Student tak ihned získává zpětnou vazbu k řešení jeho úkolu a učiteli odpadá nutnost všechny úkoly ručně opravovat. V tomto příspěvku bude představena osnova předmětu Algoritmizace a programování a také systém pro automatické ohodnocení odevzdaného úkolu studenty.

KLÍČOVÁ SLOVA

výuka, algoritmizace, programování, Python, GitHub Classroom

ÚVOD

Předmět Algoritmizace a programování je vyučován v 1. ročníku bakalářského studijního programu Biomedicínská technika a bioinformatika na Ústavu biomedicínského inženýrství, Fakultě elektrotechniky a komunikačních technologií, Vysokého učení technického v Brně. Předmět je koncipován jako úvod do problematiky algoritmizace a programování v Pythonu. V minulosti byl předmět vyučován v prostředí Matlab, ovšem v akademickém roce 2019/2020 došlo ke změně a od té doby výuka probíhá v programovacím jazyce Python. Od následujícího akademického roku došlo také v rámci bakalářského studijního programu ke změnám i v ostatních vyučovaných předmětech, které postupně přešly v rámci počítačových cvičení z programového prostředí Matlab na Python.

Pro výuku se využívá Python verze 3.10 a vyšší, integrované vývojové prostředí PyCharm Community Edition, verzovací nástroj Git a webová platforma GitHub. Všechny zmíněné softwary lze získat bezplatně a každý student má tedy možnost je využívat na svém osobním počítači. Obdobně na platformě GitHub je možné zřídit si účet zdarma a jeho základní funkcionality jsou také bezplatně dostupné.

V rámci tohoto příspěvku bude představena osnova předmětu Algoritmizace a programování, dále jednotlivé bodované aktivity, které probíhají během semestru. Blíže bude představen systém pro automatické testování konkrétních úloh využívající knihovnu pytest, verzovací nástroj Git, resp. online platformu GitHub.

OSNOVA PŘEDMĚTU ALGORITMIZACE A PROGRAMOVÁNÍ

Již z názvu předmětu vyplývá, že je jeho obsah rozdělen do dvou částí. V první polovině semestru jsou studenti seznámeni obecně se základy programování. Během jednotlivých týdnů se postupně přidávají i základní datové typy a struktury Pythonu a také základní operace s nimi. V sedmém týdnu semestru pak probíhá půlsestrální test. V druhé polovině semestru je pak kladen důraz na základy algoritmizace, analýzu časové náročnosti algoritmů a aplikování získaných znalostí na jednoduché algoritmizační úlohy. Celkové znalosti z předmětu jsou pak ověřeny během zápočtového testu na konci semestru. Podrobná osnova předmětu je uvedena v Tab. 1.

Tab. 1 Osnova předmětu po jednotlivých týdnech v semestru. Modrá část: úvod do programování v Pythonu, zelená část: úvod do algoritmizace, oranžová část: bodované aktivity

Týden semestru	Náplň cvičení	
1.	Úvod do Pythonu	Příkazová řádka, PyCharm, Python konzole, deklarace proměnných, logické operace, vývojové diagramy
2.	Textové řetězce a datové typy	Textové řetězce, základní operace a metody nad řetězcem, indexace, základní datové typy, převody mezi datovými typy
3.	Seznamy a větvení programu	Seznamy a vnořené seznamy, základní operace a metody nad seznamem, větvení pomocí podmínek, vnořené podmínky
4.	Cykly a příklady I	Cyklus for, vyhrazené příkazy pro řízení cyklu, n-tice a jejich iterátory, vnořené cykly, praktické příklady
5.	Cykly a příklady II	Cyklus while, praktické příklady
6.	Funkce a moduly	Deklarace funkce, volání funkce, globální a lokální proměnná, vstupní a výstupní parametry, dokumentační řetězec, moduly
7.	Půlsestrální test	
8.	Základy GIT, práce se soubory	Vytváření revizí, větvení, spolupráce (GitHub), práce s textovými soubory, CSV soubory a JSON soubory
9.	Datové struktury	Pole (NumPy array), množina, slovník, základní operace a metody nad nimi; výhody a nevýhody všech datových struktur
10.	Asymptotická složitost, vyhledávání	Analýza složitosti algoritmu, časová náročnost, lineární vyhledávání, vyhledávání vzorů, binární vyhledávání
11.	Asymptotická složitost, řazení	Řazení v Pythonu, selection sort, bubble sort, insertion sort
12.	Rekurze, rozděl a panuj	Rekurze, rozděl a panuj, binární vyhledávání rekurzivně
13.	Zápočtový test	

HODNOCENÍ PŘEDMĚTU

Předmět je zakončen klasifikovaným zápočtem. Z osnovy z Tab. 1 je patrné, že v průběhu semestru studenti musí zvládnout půlsestrální a zápočtový test. V prvním akademickém roce 2019/20, kdy byl zaveden Python, studenti navíc vypracovávali skupinový projekt. Bohužel vypracování projektu se příliš neosvědčilo mimo jiné také kvůli epidemii COVID-19.

Od následujícího roku byl skupinový projekt nahrazen celkem deseti domácími úkoly, které studenti každý týden vypracovávají. Vzhledem k tomu, že v osnově předmětu je zařazen i verzovací nástroj Git, bylo rozhodnuto o využití platformy GitHub pro zadávání úkolů a konkrétně její podčásti Classroom, která je určena právě ke vzdělávacím účelům.

Tato platforma byla vybrána, protože se studenti naučí pracovat se systémem Git a jeho online variantou GitHub. GitHub Classroom podporuje automatické testování zadaných úloh, a tedy studenti mají ihned zpětnou vazbu, zda je jejich řešení správné. V případě, že student má problémy s řešením úkolu, učitel může nahlédnout do jeho aktuálního řešení nebo repozitář naklonovat a vyzkoušet kód lokálně. Systém úlohy automaticky ohodnotí bez nutnosti ručního hodnocení učitelem.

JAK FUNGUJE AUTOMATICKÉ HODNOCENÍ ÚLOH?

Na začátku semestru je vytvořena třída obsahující seznam všech studentů. Každý student je dán příjmením a jménem. Během první hodiny pak každý student prováže svůj GitHub účet s konkrétním studentem ve třídě viz Obr. 1

The screenshot displays the GitHub Classroom interface for an assignment named 'hw_01'. At the top, it shows 'Classrooms / slytherins-hub-classroom-23 / hw_01'. The assignment is marked as 'Individual assignment', 'Deadline Passed', and 'Active'. A URL 'https://classroom.gi' and buttons for 'Edit' and 'Download' are visible.

Summary statistics for the assignment:

- Students total: 166 (166 Rostered, 0 Added students)
- Accepted assignments: 133 (133 Students)
- Assignment submissions: 133 (124 Submitted, 9 Not submitted)
- Passed students: 107 (107/133 Passed)

Below the summary, there are filters for 'Search for an assignment', 'Filter by unlinked accounts', 'Filter by accepted', 'Filter by passing', and 'Sort'. The 'Classroom roster' section lists four students who have submitted their work:

Student	Status	Time	Commits	Repository	Feedback
Ondrej	Submitted	On-time	11 commits 2/2	Repository	Feedback
Iaroslava	Submitted	On-time	5 commits 2/2	Repository	Feedback
Marek	Submitted	On-time	4 commits 2/2	Repository	Feedback
Dominik	Submitted	On-time	1 commit 2/2	Repository	Feedback

Obr. 1 Ukázka GitHub Classroom s přehledem odevzdaných úkolů pro daný domácí úkol

Každý týden jsou pak studentům zadávány domácí úkoly a to následujícím způsobem. Učitel vytvoří vzorový repozitář na GitHubu. V tomto repozitáři se obvykle nachází soubor README.md, který obsahuje zadání domácího úkolu, soubor `assignment_x_1.py`, do kterého studenti píšou vlastní kód, a soubor `test_assignment_x_1.py`, který obsahuje automatické testy. Tento vzorový repozitář, pak využívá GitHub Classroom pro vygenerování privátního repozitáře pro každého studenta, který úlohu přijme. Dále učitel vytvoří v GitHub Classroom úlohu a k ní přiřadí vzorový repozitář, nastaví čas odevzdání a nastaví automatické testy tzn. určí, který soubor obsahuje automatické testy a má být tedy spuštěn při nahrání nové revize do repozitáře.

Samotný student pak domácí úkol přijme otevřením odkazu, který je zveřejněn na e-learningové stránce předmětu. Po kliknutí na odkaz a přihlášení do GitHub účtu, se provede individuální vygenerování repozitáře pro daného studenta: nakopírování obsahu vzorového repozitáře do nového pod jeho jménem, nastavení automatického testování a jeho prvotní spuštění. Z pohledu studenta jde v GitHub repozitáři vidět aktuální verzi jeho řešení, všechny dosavadní revize a výsledky automatického testování (viz Obr. 2). Při hodnocení domácího úkolu pak rozhoduje poslední verze odevzdaná včas.

The screenshot shows a GitHub repository page for a student. The repository is named "hw_01" and is a private repository generated from a template. The page displays a commit history with a recent commit "b3bbc7e" on Feb 9 with 16 commits. A notification banner states "All checks have passed" for a GitHub Classroom Workflow / Autograding (push). The README.md file is visible, titled "Domácí úkol č. 1", with instructions for the assignment. The right sidebar shows repository statistics: 0 stars, 0 watching, 0 forks, and 0 releases.

Obr. 2 Ukázka GitHub repozitáře studenta včetně všech souborů a výsledku automatického testování

Studentům je hned na první hodině vysvětlen postup, jak se tímto repozitářem pracovat. Nejprve si každý student naklonuje svůj repozitář do svého počítače. V rámci své lokální verze provádí úpravy podle zadání domácího úkolu, vytváří revize v systému Git a odesílá revize do svého repozitáře na GitHub. Platforma GitHub pak zajistí spuštění automatického testování domácího úkolu při obdržení nové revize. Student si následně může prohlédnout během asi jedné minuty výsledky testování včetně podrobného logu o průběhu testování. Značná část studentů během semestru tyto logy pozorně čte a využívá je pro odhalení chyb ve vlastním kódu. Obdobně pak také využívají možnosti nahlížet a pracovat se souborem s automatickými testy.

Z pohledu učitele lze pak u konkrétní úlohy v GitHub Classroom vidět, kteří studenti úkol přijali, zda ho včas odevzdali a také výsledky automatického testování. Učitel má právo přistupovat ke všem repozitářům od studentů a může tak provádět i ruční kontrolu.

ZÁVĚR

Výuka Algoritmizace a programování probíhá na ÚBMI FEKT VUT v 1. ročníku a slouží jako úvod do problematiky programování v Pythonu s aplikací na jednoduché algoritmizační úlohy. Pro hodnocení získaných znalostí jsou každý týden semestru zadávány studentům domácí úkoly k procvičení učiva daného týdne. Pro kontrolu domácích úkolů se využívá platforma GitHub Classroom, která umožňuje spuštění automatických testů, poskytování zpětné vazby studentům a eliminuje ruční opravování úloh učitelem.