

Naučiti prvake Python

Marko Mišić, Sanja Delčev, Tamara Šekularac
Univerzitet u Beogradu – Elektrotehnički fakultet, Beograd, Srbija
{marko.misic, sanja.delcev, tasha}@etf.bg.ac.rs

Rezime: Kursevi programiranja predstavljaju važan segment u obrazovanju inženjera elektrotehnike i računarstva, a već duži niz godina u akademskoj sredini postoji polemika o tome koji programski jezik bi najpre trebalo da se uči. U okviru ovog rada će biti predstavljena iskustva stečena u okviru dobrovoljne, eksperimentalne grupe studenata prve godine na kojoj se izučava programski jezik Python. Biće opisana motivacija za njegovo uvođenje, sadržaj kursa, način njegovog sprovođenja, uspeh studenata i prodiskutovani potencijalni problemi u vezi sa uvođenjem ovog programskog jezika za sve studente prve godine.

Ključne reči: učenje programiranja; metodologija nastave; programski jezici; Python.

I. Uvod

Nastava programiranja na Elektrotehničkom fakultetu Univerziteta u Beogradu (ETF-UB) se na prvoj godini osnovnih studija sprovodi kroz predmete Programiranje 1 i Programiranje 2, kako na zajedničkoj prvog godini na Elektrotehničkim odsecima (ER), tako i na Odseku za softversko inženjerstvo (SI). Programiranje 1 se drži u zimskom, a Programiranje 2 u letnjem semestru. Oba predmeta predstavljaju uvodne kurseve i obuhvataju gradivo potrebno za sticanje osnovnih programerskih veština. Na predmetu Programiranje 1 se izučava programski jezik Pascal kao primer jednostavnog programskog jezika, dok se na predmetu Programiranje 2 izučava programski jezik C koji se koristi u profesionalnim primenama. Pored programskih jezika, oba predmeta obuhvataju i različite opšte koncepte iz domena računarstva, kao što su elementi algoritama i struktura podataka, arhitekture računara, Bulove algebre, predstavljanje celih i realnih brojeva i sl.

Predmeti su masovni, a svake godine ih pohađa više od 900 studenata na ER odsecima i više od 200 studenata na Odseku SI. Zbog velikog broja studenata, ograničenih laboratorijskih kapaciteta i nedostajućeg nastavnog kadra, nastava se uglavnom izvodi kroz tradicionalna predavanja i auditorne vežbe. Praktičan rad se ostvaruje kroz odgovarajuće predmete Praktikum iz programiranja 1 (PP1) i Praktikum iz programiranja 2 (PP2). Ovi predmeti su obavezni na Odseku SI, dok su na ER odsecima izborni i pohađa ih u proseku oko 100 studenata po semestru.

Predznanje, kao i motivacija studenata za pohađanje ovih kurseva značajno variraju. Neki poznaju veći broj programskih jezika, dok neki nemaju gotovo nikakvog iskustva. Takođe, zahtevi različitih odseka koje studenti upisuju nakon završene prve godine ER odseka se u mnogome razlikuju. U tom smislu, nije lako odabrati

programski jezik za izučavanje na prvoj godini osnovnih studija koji bi zadovoljio potrebe svih odseka na ETF-UB, jer potrebe variraju od profesionalnih primena u sistemskom i poslovnom domenu do prototipskog programiranja i modelovanja u različitim simulacijama.

Kako je programski jezik Pascal odavno prestao da bude značajan u industrijskim i akademskim okvirima, nametnula se potreba za alternativom, a prve ozbiljnije diskusije su povedene tokom 2013. godine. Nakon pažljivog razmatranja trendova koji se mogu videti u [1] [2] odlučeno je da se krene sa uvođenjem programskog jezika Python u nastavu. Ideja o uvođenju programskog jezika Python potekla je iz više izvora koji uključuju kako nastavno osoblje, tako i studente. Ovaj jezik se na ETF-UB već koristi kao podrška na nekim predmetima iz oblasti automatike, elektronike, teorije kola, energetike, kao i većeg broja matematičkih predmeta. Takođe, i veliki broj studenata računarstva ga koristi u svojim samostalnim projektima, a često i na projektnim zadacima kod kojih nije propisana upotreba nekog konkretnog programskog jezika.

Python se u poslednjih nekoliko godina nalazi u vrhu popularnosti i u profesionalnim primenama, o čemu svedoči treća pozicija u okviru TIOBE indeksa koju je Python dostigao u septembru 2018. godine [3]. Učenje programskog jezika Python kao prvog jezika ima i veliki broj pobornika u svetu [4], a prevashodno se navode pedagoški razlozi, kao što su njegova jednostavnost, lakoća usvajanja sintakse i algoritamskog načina razmišljanja i direktna primenljivost na drugim poljima kao što su matematika, obrada podataka i sl. Na kraju, programski jezik Python je otvorenog koda i veoma često je slobodni softver napisan upravo na njemu, što su prakse koje je poželjno promovisati u okviru kurseva iz oblasti programiranja. Te ideje su sadržane i u zvaničnom logotipu jezika koji se može videti na Slici 1.



Slika 1: Python logo (izvor: <https://www.python.org/community/logos/>)

Programski jezik Python je u nastavu programiranja uveden u okviru specijalne, eksperimentalne grupe predmeta PP1 na Odseku SI u školskoj 2013/2014 godini. Prvi rezultati su bili veoma ohrabrujući [5], ali je zbog dosta čestih kadrovskih promena kurs ostao na nivou specijalne grupe tokom proteklih pet školskih godina.

U nastavku rada, u okviru Poglavlja II, će biti dato više detalja o organizaciji samog kursa i iskustvima u njegovom sprovođenju. U Poglavlju III će biti razmatrani

potencijalni problemi u vezi sa uvođenjem ovog jezika na kompletnim studijama Odseka ER i SI na ETF-UB, a na kraju je dat kratak zaključak.

II. Sadržaj i organizacija kursa

Programski jezik Python je u nastavu programiranja uveden u okviru specijalne, eksperimentalne grupe predmeta PP1 na Odseku SI. Broj učesnika u okviru specijalne grupe je bio uglavnom ograničen na između 25 i 30 studenata, što je definisano kapacitetom računarskih učionica u kojima se izvodila nastava i željom da nastava bude u dovoljnoj meri interaktivna. Studenti prve godine su se dobrovoljno prijavljivali za slušanje ove forme kursa, a u godinama kada je broj prijavljenih studenata premašivao navedene brojeve, vršeno je njihovo testiranje i intervjuisanje, nakon čega je vršen odabir.

Postoji nekoliko razloga zašto je pristupljeno ovakvom, postupnom načinu uvođenja novog programskog jezika. Najpre, studenti Odseka SI su zbog prirode odseka značajno više motivisani za učenje programskih jezika i uopšteno oblast računarstva i dolaze na ETF-UB sa znatno boljim predznanjem. Drugo, namera je bila da se nastava izvodi u relativno malim grupama, u skladu sa principima kontinuiranog učenja i interaktivne nastave. Takođe, rad u maloj grupi studenata omogućava značajno interaktivnije i detaljnije ispitivanje, kao i njegovo sprovođenje kroz veći broj samostalnih aktivnosti studenata, poput domaćih i projektnih zadataka i sl. Na kraju, specijalna grupa u okviru predmeta PP1 je odabrana kao način da se jednom broju izuzetnih studenata kojima je klasično gradivo iz predmeta Programiranje 1 nezanimljivo, ponudi zanimljiva i izazovna alternativa. Sa druge strane, nastavnici i saradnici su smatrali da će kroz navedenu organizaciju nastave lakše steći iskustvo u nastavi novog programskog jezika.

Osnovni ciljevi kursa su definisani tokom prve godine održavanja specijalne grupe [5]: (1) osposobljavanje za samostalno analitičko mišljenje i rešavanje problema; (2) upoznavanje sa programskim alatima; (3) upoznavanje sa osnovnim konceptima programiranja i osnovnim algoritimima; (4) upoznavanje sa dobrim praksama u programiranju; (5) integrisanje nastavnih sadržaja, kako kroz kurs kao nezavisni entitet, tako i sa drugim kursovima.

Nastava se odvija u računarskoj učionici, pri čemu svaki student radi za zasebnim računarom. Studenti su u prilici da prezentovane sadržaje i programe pokrenu na računaru i izvrše modifikacije na licu mesta. Pored klasičnih predavanja, na delu časova je primenjena interaktivna nastava, gde predavač zajedno sa studentima na licu mesta rešava problem i piše programski kod ili studenti to čine samostalno. Takođe, tokom prvih godina je eksperimentisano i sa samostalnim radom studenata u parovima, što su studenti dosta dobro prihvatili. Navedenim metodama nastave se forsiraju principi aktivnog učenja [6]. Kod aktivnog učenja, studenti aktivno učestvuju u usvajanju znanja, umesto tradicionalnog pasivnog prenošenja znanja posmatranjem

i slušanjem. Aktivnosti na kursu su osmišljavane prateći te principe [7].

Plan predmeta obuhvata glavne aspekte rada sa programskim jezikom Python, uz osvrt na generalne programerske i algoritamske prakse. Sadržaj uključuje glavne teme kao što su: (1) promenljive, vrednosti, dodele, tipovi i operacije, primitivni tipovi; (2) kontrola toka - uslovno izvršavanje; (3) upoznavanje sa razvojnim okruženjima (4) unos podataka i čitanje podataka sa standardnog ulaza; (5) rad sa ciklusima, brojanje (*range*), ponavljanje do zadovoljenja uslova; (6) vidljivost imena i blokovska struktura programa; (7) definisanje i upotreba funkcija, rekurzija; (8) liste, pojam, definisanje, operacije, iteriranje kroz liste; (9) rad sa datotekama; (10) obrada i analiza teksta; (11) tip *n*-torke i primena; (12) regularni izrazi. Sadržaj kursa je evoluirao tokom vremena, pa je povremeno uključivao i funkcionalno-orijentisane i objektno-orijentisane elemente programskog jezika Python, rad sa heš mapama i rečnicima, generisanje pseudoslučajnih brojeva i sl. Nastava se odvijala kroz sedam dvočasa.

Može se uočiti da je plan predmeta obiman, ali temeljno izabran tako da se prikažu svi važni aspekti rada sa ovim jezikom. Sa druge strane, navedeni koncepti su na programskom jeziku Python mnogo jednostavniji za usvajanje nego na jeziku poput programskog jezika C [4], što se može videti i iz jednostavnog primera u Tabeli 1.

Tabela 1: Usporedni prikaz Hello World programa na programskim jezicima C++ i Python, zasnovano na [4]

C++	Python 3.6.6
<pre>#include <iostream.h> void main() { cout << "Hello world" << endl; }</pre>	<pre>print("Hello world")</pre>

Tematske celine kursa su ilustrovane odgovarajućim programskim primerima, kao što su rešavanje kvadratne jednačine, konverzija veličina između mernih sistema (stepeni Farenhajta u stepene Celzijusa), ali naprednijim primerima poput formiranja Fibonačijevog niza, računanja faktorijela broja, ispitivanja palindroma, računanja NZD i sl. Rad sa stringovima, datotekama i regularnim izrazima je ilustrovan na primeru jednostavne analize sadržaja HTML stranica.

Ispitivanje studenata je sprovedeno kroz dve vrste domaćih zadataka i ispitni zadatak. Ispitni zadatak se radi u ispitnom roku, a imperativ je da rešenje proradi i testira se na licu mesta. Ispitni zadatak nosi 30% ocene, tako da je naglasak na ispitivanju tokom semestra koji nosi 70% ocene.

Postoji pet „velikih“ domaćih zadataka koje studenti rešavaju samostalno. Zadaci nose po 12 poena, pri čemu se prvi zadatak odnosi na gradivo iz oblasti *picoComputer*-a (jednostavnog pseudo-assemblyera) koji je isti za sve studente PP1 kursa u okviru koga se specijalna grupa odvija. Odbrana domaćih zadataka se obavlja u računarskoj učionici pred studentima-demonstratorima. Studenti-demonstratori imaju dvostruku ulogu: da ispitaju i ocene rešenje i znanje studenata, ali i da budu edukatori kroz interakciju sa studentima. Takođe, na odbranama

studenti dobijaju i „modifikacije“ programa na licu mesta koje predstavljaju dodatne zahteve koje treba implementirati u ograničenom vremenu. Time se naglašava integracijski aspekt aktivnog učenja.

Redovan rad tokom nastave se vrednuje kroz pet „malih domaćih zadataka“ koji se zadaju nakon održanog časa iz predmeta. Ti zadaci imaju za svrhu da pomognu usvajanje obrađenog gradiva, osmišljeni su tako da je potrebno direktno primeniti koncepte koji su pokazani na času i za njihovo rešavanje nije potrebno izdvojiti mnogo vremena. U sumi nose 10 poena. Studenti rešenja zadataka predaju elektronskim putem do unapred definisanog datuma.

Za razliku od „malih“ domaćih zadataka, „veliki“ domaći zadaci zahtevaju nešto veći napor i angažovanje studenata za rešavanje. Neki problemi su formulisani i u vidu širih tekstualnih opisa, odnosno studentima se zadatak predstavlja kroz potencijalno realan problem, kako bi ih motivisali da rešenje isprojektuju na osnovu analize teksta i pokazali da programiranje može da se primeni u najrazličitijim situacijama.

Eksperimentalna priroda grupe, kao i velika zainteresovanost studenata za zahtevnijim zadacima, dovela je do toga da se studentima daje mogućnost da biraju između „lakšeg“ (osnovnog) i „težeg“ (dodatnog) problema. „Teži“ problem od studenata ponekad iziskuje bolje poznavanje algoritama i struktura podataka. Te probleme mahom biraju studenti sa boljim predznanjem iz oblasti programiranja. Nastavnici na predmetu su svesni da to nije slučaj sa većinom studenata i da ovakav koncept nije održiv za masovnije grupe, ali takođe žele da takvim studentima održe interesovanje i pruže priliku da razvijaju svoje veštine.

Nakon prve godine držanja kursa, u školskoj 2013/2014 godini je napravljena anketa kojom je ispitivano zadovoljstvo studenata novouvedenom specijalnom grupom i izučavanjem programskog jezika Python. Anketa čiji su rezultati detaljno obrađeni u [5] je pokazala veliko zadovoljstvo među studentima koji su pohađali kurs, tako i među studentima-demonstratorima koji su učestvovali u njegovom sprovođenju. Jedan od glavnih zaključaka ankete je da studenti koji su se prijavili za izučavanje programskog jezika Python kroz specijalnu grupu pretežno imaju kvalitetnu osnovu za učenje programiranja, formiranu bilo kroz školski program, bilo kroz samostalno učenje i praksu.

Takođe, većina studenata koji su pohađali kurs je navela da ga aktivno koriste ili da će nastaviti da ga koriste i za potrebe rešavanja problema na drugim predmetima ili u samostalnoj praksi. Stiče se utisak kroz razgovor sa studentima i nastavnim osobljem da je takva praksa zadržana kod mnogih studenata tokom poslednjih pet godina držanja specijalne grupe i da je procenat onih koji su upoznati sa ovim programskim jezikom sve veći.

Velika većina studenata koji su pohađali kurs tokom prethodnih pet godina je isti završila sa ocenom 10. To se može pripisati velikoj motivaciji i odličnim predznanjem studenata Odseka SI koji su u njoj učestvovali, ali i kvalitetnom selekcijom i načinom izvođenja nastave. U

tom smislu podaci o uspehu studenata tokom godina nisu sasvim relevantni, pa stoga i nisu izloženi.

III. Python kao uvodni programski jezik

Već duži niz godina u akademskoj sredini se vode polemike o upotrebi programskog jezika Python kao prvog programskog jezika za studente elektrotehnike i računarstva. Objavljuju se rasprave [4], porede se različiti jezici [8] i sprovode ankete [9]. Nakon pet godina održavanja specijalne grupe, stekao se utisak da Python nailazi na dobar prijem kod studenata, zbog svoje jednostavnosti, implicitnosti i jasnoće, što je na tragu onoga na šta ukazuju i drugi autori [9].

Python je slabo tipizirani, dinamički, skriptni jezik i stoga se može veći akcenat staviti na razmatranje zanimljivih, realnih problema, učenje o tehnikama dekompozicije problema, osnovnih algoritama i tehnika otklananja grešaka u programu. Sa druge strane, takvim pristupom se studenti više usmeravaju ka programiranju prototipova i modela, dok se se u aplikativnom i sistemskom programiranju koje podrazumeva pisanje efikasnog koda na značaju daje i konceptima poput upravljanja memorijom, pokazivačima, tipovima podataka, upravljanju greškama, optimizaciji i sl. Ti koncepti i mehanizmi se prvenstveno vide kroz programske jezike kao što su C ili C++.

Iako se postavlja pitanje da li ih je neophodno izučavati na prvom kursu programiranja, kod studenata svakako treba izgraditi svest da je Python pre svega skriptni jezik, što sa sobom nosi određena ograničenja za primenama. Sa druge strane, izučavanje programskog jezika Python kao uvodnog jezika bi svakako bilo od koristi na predmetima starijih godina sa drugih odseka, gde gore pomenuti koncepti nisu od ključne važnosti, već se teži upotrebi računara za rešavanje konkretnih inženjerskih problema, kao što su različite simulacije. Na tim predmetima je uglavnom zastupljen programski paket MATLAB, a Python predstavlja odličnu, besplatnu alternativu sa dobrom bibliotečkom podrškom.

Na žalost, specijalna grupa nije mogla da se iskoristi kao realno merilo postignuća studenata u smislu stečenih ocena nakon održanog kursa, zbog velike većine studenata koji su položili ispit sa ocenom 10. Stoga je teško proceniti kako bi se identičan način rada mogao preslikati na kurs Programiranja 1 sa preko 1100 studenata na oba odseka ETF-UB. Iskustva u radu u maloj grupi uspešnih studenata se pokazuje kao izuzetno dobro. Međutim, uvođenje ovog programskog jezika kao obaveznog bi iziskivalo pažljivu analizu u smislu organizacije i metoda držanja nastave, ali i ispitivanja.

Izložena metodologija ima smisla samo ukoliko se ispoštuju svi njeni elementi, a za veliki broj studenata problem predstavljaju kako raspoloživi kapaciteti računarskih laboratorija, tako i nedostatak nastavnog kadra koji bi učestvovao u svim formama interaktivne nastave. Organizacija ispitivanja za toliki broj studenata takođe predstavlja problem, jer sa masovnošću na kursovima dolaze i problemi poput plagijarizma [10], što

može nepovoljno da utiče na kvalitet stečenog znanja studenata.

Rešenje se može tražiti u različitim oblicima kombinovanog učenja [11] koje bi uključivalo korišćenje platformi za elektronsko učenje poput *Moodle* i drugih sistema, kao što su *online judge* sistemi za automatizovano testiranje i ocenjivanje programskog koda. U okviru takvih platformi bi trebalo razviti i odgovarajuće pakete za samostalno učenje koji bi uključivali interaktivne nastavne materijale i set laboratorijskih vežbi i zadataka za samostalnu izradu. Takvi pokušaji već postoje i naišli su na odličan prijem kod studenata [12]. Naravno, takav model bi zahtevao i aktivniji samostalan rad samih studenata.

IV. Zaključak

Programski jezik Python svakako predstavlja budućnost nastave iz oblasti programiranja, kako na ETF-UB, tako i u svetu. Njegova jednostavnost, interaktivnost i široka primenljivost ga čine pogodnim za učenje i upotrebu, kako od strane studenata sa niskim predznanjem, tako i onih sa naprednijim znanjem i potrebama. Iako je Python objektno-orijentisani programski jezik, pokazao se pogodnim i u nastavi proceduralnog programiranja.

Nadamo se da će se u narednom periodu aktivnije ući u proces transformacije baznih predmeta iz oblasti programiranja na ETF-UB i da će programski jezik Python biti uveden kao uvodni programski jezik za sve studente osnovnih studija. U tom smislu, potencijalni problemi koji su izloženi u prethodnom poglavlju se više mogu posmatrati kao podsticaj, nego kao realne prepreke u ostvarivanju tog cilja koji bi doprineo poboljšanju obrazovanja iz oblasti programiranja za sve studente elektrotehnike i računarstva.

Literatura

- [1] PYPL Popularity of Programming Language. [Online] Dostupno na <http://pypl.github.io/PYPL.html> [pristupljeno sep. 09, 2018]
- [2] R. Mason, G. Cooper, M. de Raadt, "Trends in Introductory Programming Courses in Australian Universities – Languages, Environments and Pedagogy," *Proceedings of the Fourteenth Australasian Computing Education Conference (ACE2012)*, Melbourne, Australia, 2012
- [3] TIOBE Index for September 2018. [Online] Dostupno na <https://www.tiobe.com/tiobe-index/> [pristupljeno sep. 09, 2018]
- [4] Lorena A. Barba, "Why I push for Python", 2014. [Online] Dostupno na <http://lorenabarba.com/blog/why-i-push-for-python/> [pristupljeno sep. 09, 2018]
- [5] N. Vitorović, J. Protić, "Eksperimentalno uvođenje programskog jezika Python kao prvog programskog jezika za studente elektrotehnike i računarstva," *XX skup Trendovi razvoja: "Razvojni potencijal visokog obrazovanja"*, Kopaonik, 24. - 27. 02. 2014
- [6] I Ivić, A. Pešikan, S. Antić, *Aktivno učenje 2*, Institut za psihologiju, Beograd, 2001
- [7] L. Frank, *A Self-Directed Guide to Designing Courses for Significant Learning*, San Francisco, 2003
- [8] H. Fanghor, "A Comparison of C, MATLAB and Python as Teaching Languages in Engineering," *Computational Science - ICCS 2004: 4th International Conference*, 2004.
- [9] I. D. Sanders, and S. Langford, "Students' perceptions of Python as a first programming language at Wits," *ACM SIGCSE Bulletin*, vol. 40, no. 3, pp. 365-365, 2008
- [10] M. Mišić, J. Protić, and M. Tomašević, "Improving source code plagiarism detection: Lessons learned", *25th Telecommunication Forum (TELFOR)*, November, 2017. IEEE, pp. 1-8
- [11] F. L. Wang, J. Fong, and M. Choy, "Blended Learning for Programming Courses: A Case Study of Outcome Based Teaching & Learning", *Lecture Notes in Computer Science*, 2008, 4823/2008, pp. 606-617
- [12] A. Radenski, "Python First: A lab-based digital introduction to computer science," *ACM SIGCSE Bulletin*, vol 38, no. 3, pp. 197-201, 2006