

Da li se dovoljno govori o slobodnom softveru u našim školama?

Bojana Sataric

Nacionalni prosvetni savet, Beograd, Srbija

bojana.sataric@gmail.com

Rezime: Biti slobodan je osnovno ljudsko pravo, a slobodan softver je samo korak ka tome. Reforma školstva, započeta uvođenjem informatike kao obaveznog predmeta od petog razreda osnovne škole, doprinela je da obrazovanje bude usmereno ka širenju novih znanja. Za programske jezike Scratch i Python ima tutorijala, priručnika, zbirki zadataka na srpskom jeziku, što olakšava rad nastavnika i učenika za nastavnu temu programiranje kada se traži vizuelni ili tekstualni programski jezik. Ovaj rad ima za cilj da odgovori na pitanje: Da li se u školama u Srbiji dovoljno govori o slobodnom softveru i šta on stvarno jeste?

Ključne reči: škola; slobodan softver; ishodi predmeta; računarstvo i informatika; Scratch; Python.

I. Uvod

Osnivač Zadužbine slobodnog softvera, Richard Stallman, je rekao: “slobodan softver nije pitanje cene, već ljudskih sloboda”. Biti slobodan je osnovno ljudsko pravo, a slobodan softver je samo korak ka tome. Kod nas pojam slobodnog softvera nije poznat dovoljno. Koliko se slobodan softver primenjuje u školama, a koliko se o njemu govori i zna?

Slobode, po Richardu Stallmanu, koje definišu slobodan softver su [1]:

0. sloboda da se program pokrene i koristi kako mi se prohte.
1. sloboda da se program proučava kako radi i menja po mojoj volji.
2. sloboda da se program deli sa prijateljima.
3. sloboda da se deli program koji sam ja izmenio.

II. Stanje u školama

Reforma školstva, započeta uvođenjem informatike kao obaveznog predmeta od petog razreda osnovne škole školske 2017/2018. godine, doprinela je da obrazovanje bude usmereno ka širenju novih znanja. Osposobljavanje učenika za upravljanje informacijama, bezbednu komunikaciju u digitalnom okruženju, kreiranje digitalnih sadržaja i računarskih programa za rešavanje različitih problema u društvu koje se razvojem digitalnih tehnologija brzo menja.

Program nastave i učenja usmerava nastavnika da nastavni proces koncipira u skladu sa definisanim ishodom, odnosno da planira kako da učenici ostvare ishode, i da izabere odgovarajuće metode, aktivnosti i tehnike za rad sa učenicima – što u slučaju informatike podrazumeva odabir softvera. Definisani ishodi pokazuju nastavniku i koja su to specifična znanja i veštine koja su

učeniku potrebna za dalje učenje i svakodnevni život. Da bi svi učenici ostvarili predviđene ishode, potrebno je da nastavnik upozna specifičnosti načina učenja svojih učenika i prema njima planira i prilagođava nastavne aktivnosti. Nastavnik treba da osmisli aktivnosti tako da uključuju praktičan rad uz primenu IKT-a (IKT – informacione i komunikacione tehnologije), povezivanje različitih sadržaja iz drugih tema unutar samog predmeta, kao i sa drugim predmetima. Učenik treba da ima dovoljno prostora za inicijativu i kreativnost. Kroz diskusiju sa učenicima potrebno je odabirati adekvatne alate, koncepte i strategije za realizaciju određenih aktivnosti. U toku realizacije planiranih aktivnosti, poželjno je raditi na uspostavljanju i negovanju navika i ponašanja kao što su postupnost, analitičnost, istrajnost, samostalnost u radu, ali i spremnost na saradnju i odgovoran pristup timskom radu.

Dostizanje definisanih ishoda može se ostvariti uz određeni stepen slobode nastavnika kako u izboru metoda rada, programskih alata i tehnologija, tako i u redosledu i dinamici realizacije elemenata različitih tematskih oblasti. Na internetu i u literaturi se mogu naći primeri dobre prakse koje, uz prilagođavanje uslovima rada i poštovanje autorskih prava, treba koristiti u nastavi i učenju.

U osnovnoj školi i gimnazijama, nastavnicima se preporučuje radi razvijanja međupredmetnih kompetencija i ostvarivanja korelacije sa drugim predmetima, da realizuju sa učenicima nekoliko projektnih zadataka koji obuhvataju teme i iz drugih predmeta. Pri realizaciji programa data je prednost projektnoj, problemskoj i aktivno orijentisanoj nastavi, kooperativnom učenju, izgradnji znanja i razvoju kritičkog mišljenja.

Ni jedan od programa nije obavezujući. Postoje preporuke u vidu: “vizuelni programski jezik”, “tekstualni programski jezik”, “rad sa tekstem”, “tabelarne kalkulacije”... te se u školama u Srbiji od slobodnog softvera koriste: Scratch, Python, Jupyter, Pygame, Ubuntu, Libre Office, Notepad++, Inkscape, Gimp, Geogebra, Octave, R, Audacity, Euler, jEdit... U glavnom, programi koji su neophodni u svakodnevnom korišćenju računara i u nastavnom procesu.

Na žalost, učenicima se ne priča dovoljno o slobodnom softveru. Ovaj proces je prepušten individualnoj volji, znanjima i interesovanjima nastavnika. Ako govorimo o menjanju koda slobodnog softvera, tj prilagođavanju izvornog koda, tada su nastavnici i učenici samo konzumenti slobodnog softvera. Ali, kada je reč o korišćenju slobodnog softvera kao alata, onda učenici kreiraju programe u nekom od programskih jezika, koriste ih, menjaju, dele sa drugima... Učenici su generacija Z - “digitalni

urođenici“. Njihov uzrast, radoznalost, širina i velika znanja ponekad nadmašuju znanja nastavnika. Učenici su motivisani da istražuju, kreiraju i koriste raznovrsne softverske alate.

Nije retkost da nastavnici na pogrešan način razumeju pojam slobodnog softvera i ne shvataju njegove prednosti. Često poistovećuju sa pojmom “besplatan softver“ te se ta dva pojma preklapaju. U zavisnosti od predmeta koje predaju, nastavnici koriste samo one softvere koji “zadovoljavaju“ potrebe nastave. Bez obzira da li su i u kojoj meri nastavnici zainteresovani da govore o slobodnom softveru, oni ga u redovnoj nastavi primenjuju.

U poslednjih nekoliko godina distributeri hardvera instaliraju operativni sistem otvorenog koda u računare koje prodaju. Sada su korisnici (nastavnici) u obavezi da se upoznaju i da sebe nateraju da nauče npr. Ubuntu.

Ako se slobodan softver posmatra kroz pisanje programa u programskom jeziku, onda su mnogi nastavnici i kreatori nastavnog materijala. Može se reći da je program nastave i učenja tako koncipiran da podstiče kreativne nastavnike da sami kreiraju deo nastavnog sadržaja kako bi učenicima olakšali učenje. Svakodnevna iskustva govore da je potrebno stimulisati nastavnike, učenike, i drugo stručno osoblje u obrazovnim institucijama da kreiraju nastavni sadržaj kako bi se omogućilo lakše savladavanje nastavnog gradiva i kako bi realizacija inkluzije bila uspešnija uz poštovanje individualizacije i diferencijacije u nastavi. U tome veliku ulogu ima Fondacija Petlja, (<https://petlja.org/>) koja je za dve godine postojanja kreirala nastavne sadržaje od početnih do naprednih nivoa programiranja. [2]

III. Nastavni sadržaji

Glavni cilj obaveznog predmeta informatika je unapređivanje algoritamske pismenosti u Srbiji u opštem obrazovanju. To se postiže adekvatnim načinom učenja programiranja u osnovnim i srednjim školama. Za efektivno unapređivanje učenja programiranja, polazna tačka i osnova treba da budu kvalitetni materijali za učenje na koje se dalje naslanjaju i obuke nastavnika, rad u učionici i samostalan rad đaka. U skladu sa tim, jedna od glavnih aktivnosti Fondacije Petlja je razvoj besplatnih interaktivnih materijala koji su prilagođeni uzrastu učenika, ali koji su im i korisni i zanimljivi. Osim toga, ti materijali izrađeni su tako da predstavljaju i podršku nastavnicima osnovnih i srednjih škola koji ove materijale mogu koristiti prilikom osmišljavanja i izvođenja nastave.

Scratch je vizuelni programski jezik koji se izučava u petom razredu osnovne škole i služi za savladavanje osnova programiranja uz sticanje osnova algoritamske pismenosti svih učenika. Da bi učenik tog uzrasta napravio svoju sopstvenu kompjutersku igru, animiranu priču, uputstvo ili prezentaciju, to mu omogućava programski jezik Scratch, lokalizovan na srpski jezik i ćirilčno pismo. I najduža putovanja počinju prvim

korakom. U svet programiranja učenici petog razreda uplovljavaju koristeći programski jezik otvorenog koda.

Oni znatizeljni mogu pogledati izvorni kod Scratch pomoću pregledača i drugih alata unutar okruženja programa Squeak. Scratch je napisan na programskom jeziku Squeak, modernom, open-source kodu, sa punim performansama implementacije moćnog, nekada veoma popularnog Smalltalk programskog jezika i okruženja.

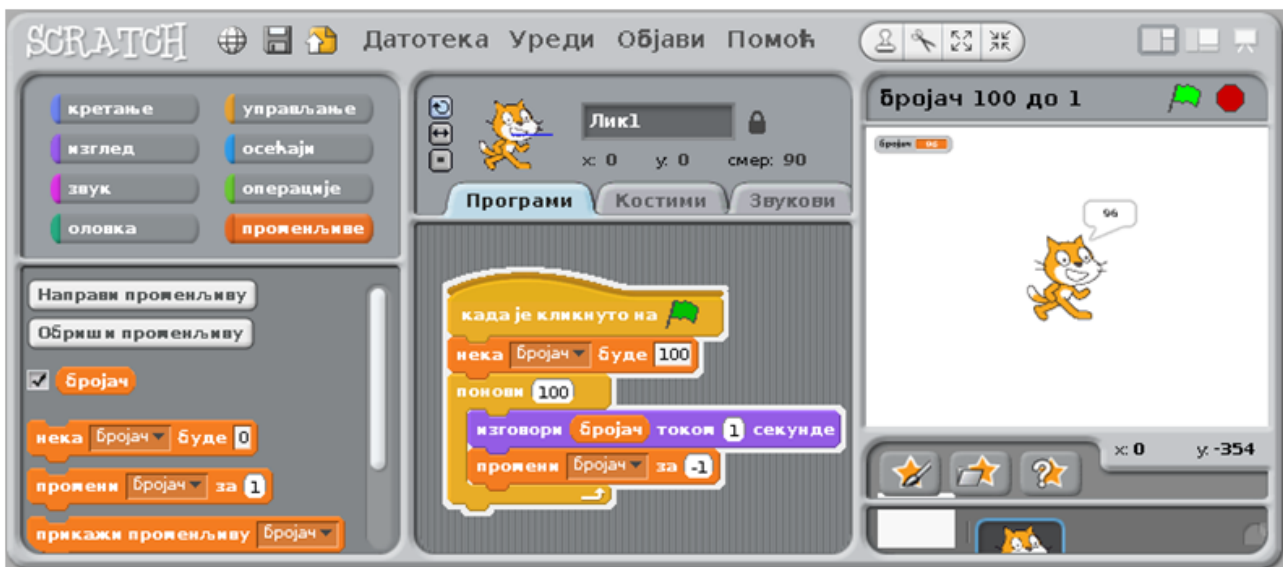
Scratch je takođe i naziv za grafičko okruženje za razvoj programa pisanih u programskom jeziku Scratch. Osnovna verzija razvojnog okruženja izvršava se u pregledaču interneta. Ovo razvojno okruženje omogućava direktno objavljivanje kreiranog programa kao i saradnju sa drugim Scratch programerima. Postoje verzije ovog razvojnog okruženja za Linux, Mac OS X i MC Windows operativne sisteme. Scratch razvojno okruženje je otvorenog koda. Kompletno okruženje i programski jezik su prilagođeni kroz lokalizaciju. [3], [9]

Prelazak sa Skreča na klasičan programski jezik, nije lak za učenika. Treba da ovlada apstraktnim mišljenjem. Pri tome treba naći način da prelazak bude što bezbolniji i da učenik ne bude suviše opterećen tehničkim detaljima koje nosi klasičan programski jezik i realno razvojno okruženje. U zavisnosti od toga koliko se uspešno može spustiti nivo tog dodatnog opterećenja koje nosi klasičan programski jezik, zavisi koliko će uspešno šira populacija mlađeg uzrasta savladati početke programiranja u Python-u. Python je tekstualni programski jezik čija je sintaksa orijentisana ka strukturiranom programiranju i omogućava elegantno izražavanje algoritama. Gvido van Rosum je programer koji je autor programskog jezika Python. Na portalu Fondacije Petlja postoji interaktivan priručnik iz koga učenik može da radi primere u okviru veb pregledača u kontekstu lekcije, tako da čitanjem, gledanjem video-sadržaja i interaktivnim radom postepeno počinje da koristi programski jezik. [4]

SCRATCH



Slika 1: Maskota programa Scratch, By The Scratch team - <https://en.scratch-wiki.info/wiki/File:ScratchCat-Small.png#filelinks>, CC BY-SA 4.0, <https://commons.wikimedia.org/w/index.php?curid=75559628>.
By Scratch Team - Scratch, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=47438844>.



Slika 2: Izvršenje Scratch programa.



Slika 3: Python logo, By www.python.org - <https://www.python.org/community/logos/>, GPL, <https://commons.wikimedia.org/w/index.php?curid=34991637>

Veoma interesantan domen primene programiranja je izrada programa koji koriste grafičke mogućnosti računara, tj. mogućnost iscrtavanja oblika, prikaza slika, kreiranja animacija, simulacija i slično. Takve su, na primer, računarske igre koje obično podrazumevaju da korisnik mišem ili tastaturom upravlja nekim likom koji se prikazuje na ekranu i šeta se kroz virtuelni svet, savlađujući prepreke i boreći se protiv "neprijatelja". Programski jezik Python neke grafičke mogućnosti nudi u sklopu svoje standardne biblioteke. Jedna takva, pogodna za učenje programiranja računarske grafike je biblioteka PyGame koja učenicima sedmog razreda omogućava da uđu u svet grafike. [5], [11]



Slika 4: Pygame logo, By the Corruptor/pygame developers - https://github.com/pygame/pygame/blob/master/docs/pygame_logo.gif, LGPL, <https://commons.wikimedia.org/w/index.php?curid=76699336>

U osmom razredu se uče osnove obrade podataka, analize i prezentacije podataka uz korišćenje okruženja za interaktivna izračunavanja Jupyter. To je radno okruženje za programiranje koje se može upotrebljavati za nekoliko programskih jezika. Jedan od njih je Python. U tom okruženju se mogu kombinovati izvorni kod programa sa tekstom i crtežima koji opisuju šta program radi i kako, kao i izlazima programa koji mogu biti tekst i grafika.

Ime projekta Jupyter odnosi se na tri programska jezika koja podržava, a to su Julia, Python i R. [6]



Slika 5: Jupyter logo, By Cameron Oelsen - <https://github.com/jupyter/jupyter.github.io/blob/master/assets/main-logo.svg>, BSD, <https://commons.wikimedia.org/w/index.php?curid=68763478>

U gimnazijama učenici rade analizu i obradu podataka korišćenjem prilagođenih programskih okruženja (poput Jupyter/Python, Matlab, Octave, R). Sve vreme treba vršiti poređenje ovog pristupa sa rešavanjem istih zadataka pomoću programa za tabelarna izračunavanja i isticati prednosti i mane jednog u odnosu na drugi. Cilj ovog segmenta je upoznavanje učenika sa algoritmima koji leže u osnovi analize i obrade podataka i upoznavanje učenika sa njihovom implementacijom u klasičnom programskom jeziku, bez korišćenja gotovih

bibliotečkih funkcionalnosti. Kroz to učenik uči opšte tehnike dizajna i analize algoritama i programiranja. [7]

IV. Zaključak

Koliko je slobodan softver danas prisutan u školama u Srbiji? Na osnovu istraživanja Udruženja nastavnika informatike Srbije iz 2011. godine, [8] u Srbiji je tada slobodan softver koristilo svega pet posto škola i devet posto nastavnika. Možemo reći da sada sve osnovne škole u drugom ciklusu obrazovanja i mnoge gimnazije koriste slobodan softver u nastavi [3], [4], [5], [6], [7]. Postoji onlajn literatura na srpskom jeziku, javno dostupne zbirke, praktikumi i vežbe primenljive u nastavi. Pišu se i objavljuju udžbenici.

U bliskoj budućnosti se očekuje da će količina materijala za primenu slobodnog softvera u nastavi biti značajno uvećana i raznovrsna.

Literatura

- [1] Operativni sistem GNU: (dostupno na: <https://www.gnu.org/philosophy/free-sw.sr.html>)
- [2] Fondacija Petlja, <https://petlja.org/>

- [3] Službeni glasnik Republike Srbije - Prosvetni glasnik broj 9/2017
- [4] Službeni glasnik Republike Srbije - Prosvetni glasnik broj 3/2018: (dostupno na: <https://www.pravno-informacioni-sistem.rs/SlGlasnikPortal/viewdoc?uuid=57bda1cd-4eff-4ab3-9f0c-561ad2c01766>)
- [5] Službeni glasnik Republike Srbije - Prosvetni glasnik broj 3/2019: (dostupno na: <https://www.pravno-informacioni-sistem.rs/SlGlasnikPortal/viewdoc?uuid=a24406e1-a0df-4c0e-8d37-889324374c16>)
- [6] Službeni glasnik Republike Srbije - Prosvetni glasnik R.S. broj 11/2019 (dostupno na: <https://www.pravno-informacioni-sistem.rs/SlGlasnikPortal/viewdoc?regactid=428910&doctype=reg&findpdfurl=true>)
- [7] Službeni glasnik Republike Srbije - Prosvetni glasnik broj 8/2019: (dostupno na: <https://www.pravno-informacioni-sistem.rs/SlGlasnikPortal/viewdoc?uuid=3a36ddff-48e0-47c3-8950-de51925bdf27>)
- [8] Izvor: <http://blog.b92.net/text/25217/Slobodan-softver-u-nastavi/>
- [9] <https://scratch.mit.edu/>
- [10] <https://www.python.org/>
- [11] <https://www.pygame.org/news>
- [12] <https://jupyter.org/>