

ULOGA PROGRAMIRANJA U POUČAVANJU STEM PODRUČJA

Dr. sc. Jelena Nakić

Prirodoslovno-matematički fakultet Split

jelena.nakic@pmfst.hr

Sažetak

Danas u svijetu, pa tako i u Hrvatskoj, nedostaje velik broj računalnih stručnjaka, a visoka razina digitalnih vještina zahtijeva se i u mnogim drugim zanimanjima. Učenje programiranja i računalnog razmišljanja nužno je za stjecanje kompetencija za ova zanimanja, ali je i mnogo više od toga. U ovom izlaganju naglasit će se razlika između programiranja i računalnog razmišljanja, te će se pokazati da se poticanjem računalnog razmišljanja kod učenika razvijaju i brojne druge sposobnosti, kao što su kreativno razmišljanje, sposobnost rješavanja problema i razvoj apstraktnog mišljenja. Ove su sposobnosti od velike pomoći učenicima u svladavanju mnogih nastavnih predmeta iz područja tehničkih i prirodnih znanosti, a također su važne i u svakodnevnom životu. U radu se također predstavljaju primjeri dobre prakse u poučavanju programiranja i računalnog razmišljanja od najranije dobi, kao što su vizualno programiranje, programiranje kroz igrice, programiranje robota, 3D virtualnih svjetova, itd. Konkretnije se opisuju natjecanja u programiranju (*Croatian Makers Liga*) i računalnom razmišljanju (natjecanje *Dabar*), koja pridonose razvoju navedenih vještina, te aktivnosti vezane uz početno učenje programiranja na *Festivalu znanosti* tijekom školske godine 2016./17. na Prirodoslovno-matematičkom fakultetu u Splitu.

Ključne riječi: STEM, programiranje, računalno razmišljanje, *Croatian Makers Liga*, *Dabar*

1. Uvod

Prema podacima sa 16. CARNetove korisničke konferencije (CUC 2014), održane u studenom 2014. godine u Zagrebu, istaknuto je kako u Europskoj uniji vlada velika potražnja za računalnim stručnjacima [1]. U proteklih desetak godina najtraženije je zanimanje računalnog programera. Nastavi li potražnja rasti ovim tempom, predviđa se da bi do 2020. godine u Europskoj uniji moglo nedostajati oko 900.000 računalnih stručnjaka. U Hrvatskoj se primjećuje isti trend, programeri su vrlo traženi, a oni iskusni odlično su plaćeni. Kao posljedica takvog trenda, sve više mladih računalnih stručnjaka pronalazi posao u Hrvatskoj, a postoji i izvjestan broj onih koji se nakon nekoliko godina rada u inozemstvu odlučuju na nastavak karijere u domovini.

Pojam IT profesionalca danas podrazumijeva ne samo programera, nego čitav niz zanimanja vezanih uz informacijsko-komunikacijsku tehnologiju (IKT). Osim toga, ta je tehnologija sve više prisutna i u drugim zanimanjima koja za svoj primarni cilj nemaju ni razvoj ni upotrebu IKT-a. Tako, primjerice, liječnik danas više ne piše nalaze rukom niti koristi pisači stroj, nego koristi računalo s posebnim aplikacijama koje su namijenjene isključivo njegovim potrebama. Takve specijalizirane aplikacije u pravilu imaju za cilj digitalizaciju društva na dobrobit pacijenata, odnosno šire zajednice. Na sličan način, zbog istih društvenih potreba, predviđa se da će informacijsko-komunikacijska tehnologija postati sastavni dio mnogih zanimanja u kojima danas uglavnom nije prisutna, kao što su, na primjer, zidari, frizeri i slično. Osim postojećih zanimanja, vrlo je izvjesno da nam skora budućnost donosi čitav spektar novih zanimanja koja će neminovno uključivati korištenje IKT-a na načine koje danas vjerojatno ne možemo ni zamisliti. Točnije, svjetska istraživanja predviđaju da će 65 posto učenika koji danas upisuju prvi razred osnovne škole već nakon završetka svojeg školovanja raditi poslove koji danas ne postoje. Danas nemamo predodžbu koja će to biti zanimanja niti koje će se kompetencije zahtijevati da bi se ti poslovi mogli uspješno obavljati, ali možemo biti sigurni da će uključivati napredno korištenje suvremenih tehnologija.

Svi ovi trendovi postavljaju velike izazove pred nastavnike i sve ostale sudionike obrazovnog procesa. Obrazovanje za budućnost počinje danas iako ne znamo kako će poslovi budućnosti izgledati. Stoga ne možemo više ciljano obrazovati stručnjake za pojedina zanimanja, pogotovo ne možemo garantirati da ćemo ih opskrbiti svim potrebnim znanjima, nego moramo osposobiti učenike za kontinuirano učenje, koje će se vrlo vjerojatno protezati tijekom čitavog njihova radnog vijeka. Primjerice, liječnik će dobiti novu, poboljšanu aplikaciju za pisanje nalaza i izdavanje recepata. Nadalje, suvremena će dijagnostika koristiti tehnologije s naprednim mogućnostima, što će dovesti do prilagodbe sučelja tih uređaja kako bi bili što učinkovitiji za korištenje. Zatim dolazi do potrebe objedinjavanja usluga u jedinstveni sustav, što našem liječniku donosi nove zahtjeve za prilagodbom, i tako dalje. Uspješno obavljanje svakodnevnih zadataka u mnogim poslovima bit će izravno povezano s učinkovitim korištenjem tehnologije. Istovremeno je potrebno razvijati kognitivne vještine koje su u sadašnjem obrazovanju značajno zanemarene, kao što su kritičko mišljenje i kreativnost u rješavanju problema. Veliki doprinos ovom cilju može dati upravo učenje programiranja. Istraživanja u obrazovanju pokazala su još prije više od trideset godina da učenje programiranja potiče temeljito razmišljanje, precizno izražavanje u formulaciji problema, raščlambu problema na manje dijelove, izradu plana rješavanja, uočavanje sličnosti s poznatim problemima, mogućnosti primjene već poznatih rješenja na rješavanje novog problema, te kritički osvrt na vlastito rješenje i uočavanje mogućnosti za njegovo poboljšanje [2]. Sve su ove dobrobiti koje donosi učenje programiranja vrlo vrijedne i učenici koji imaju razvijene ove kvalitete lakše i temeljitije usvajaju ne samo sadržaje iz STEM predmeta, nego

jačaju sposobnosti učenja i razumijevanja svih ostalih nastavnih sadržaja, kao i izazova svakodnevnog života u svim njegovim aspektima.

Ovaj rad sastoji se od pet poglavlja. U idućem poglavlju objašnjava se što je to STEM i na koji način programiranje pridonosi poučavanju u tom području. U poglavlju koje slijedi definira se pojam programiranja, te se izlažu programski jezici i okruženja koja su prikladna za početno učenje programiranja. U nastavku se opisuju natjecanja u programiranju i računalnom razmišljanju koja je autorica provodila u svojem neposrednom radu s učenicima u svrhu promidžbe učenja programiranja od najranije dobi. Spominju se i aktivnosti provedene na *Festivalu znanosti* kojima se također nastoji približiti pojam programiranja kroz edukativne igre namijenjene predškolicima i mlađim učenicima osnovne škole. Posljednje poglavlje donosi zaključna razmatranja i smjernice budućih aktivnosti.

2. Programiranje u poučavanju STEM područja

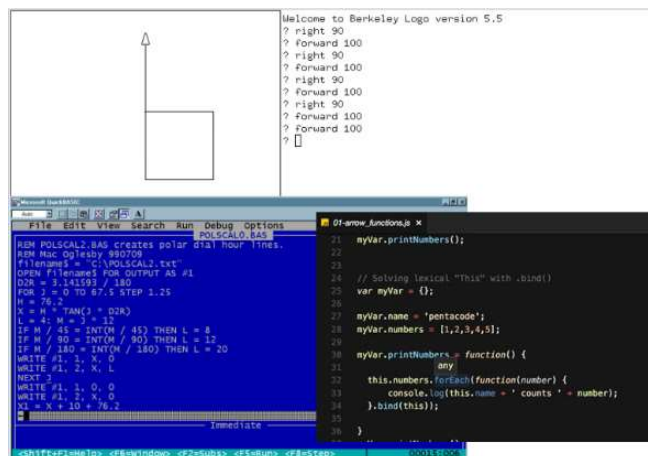
STEM (Science, Technology, Engineering, Mathematics) kratica je za grupu disciplina koje se odnose na znanost, tehnologiju, inženjerske discipline i matematiku. STEM zanimanja danas su najtraženija na tržištu rada i vrlo su dobro plaćena. Velike informatičke tvrtke koje se danas u Hrvatskoj, pa tako i u Splitu, razvijaju i rastu velikom brzinom sve više ulažu u jačanje istraživačkih timova koji generiraju nove spoznaje i trendove u istraživanju i razvoju. Takvi su timovi redovito interdisciplinarni i okupljaju stručnjake i znanstvenike iz svih STEM disciplina. Osim toga, stručnjaci koji imaju razvijene STEM kompetencije vrlo često u svojim zanimanjima iskazuju sklonost interdisciplinarnom pristupu i po potrebi se vrlo jednostavno prilagođavaju novim zahtjevima tržišta, te su sposobni uklopiti se u nova zanimanja. U vremenu kad ne znamo koje poslove donosi čak ni skora budućnost možemo s velikom vjerojatnošću pretpostaviti da će sposobnost brze prilagodbe na nova zanimanja postati jedan od imperativa konkurentnosti na tržištu rada.

Današnje obrazovanje učenika za STEM područja mora uvažavati sve navedene činjenice i zbog toga je ključno razmotriti sve mehanizme koji mogu pridonijeti povećanju kvalitete obrazovanja za STEM. Očito je da učenje programiranja koje slijedi konstruktivistički pristup tu ima izuzetno veliku ulogu jer pridonosi dubljem razumijevanju koncepata, njihova konteksta i mogućnosti primjene. Kako naglašavaju akademik Leo Budin i suradnici [3], usvajanje novih koncepata u STEM područjima skladno se nadopunjuje rješavanjem problema korištenjem programiranja. Ako učenik treba napisati program kojim će riješiti matematički ili tehnički problem, on najprije mora detaljno proučiti postupak rješavanja tog problema u njegovoj osnovnoj disciplini. Zatim pristupa rješavanju problema programiranjem, odnosno analizira problem, raščlanjuje ga na manje dijelove i definira algoritam za rješavanje zadatka korak po korak. Nakon što je napisao program, učenik testira njegovo izvršavanje da se uvjeri u točnost izvršavanja zadatka. Ljepota rješavanja problema programiranjem

leži u činjenici da računalo velikom brzinom izvršava složene algoritme, što učeniku štedi vrijeme prilikom svakog novog izračuna. Tako sada učenik može korištenjem različitih skupova ulaznih podataka vrlo brzo i učinkovito istražiti kontekst problema i moguće aspekte njegove primjene. Upravo opisani doprinosi programiranja u poučavanju STEM područja mogu se sažeti u dva aspekta: prvo, izbjegli smo „bubanje“ formula jer je učenik kroz razmišljanje o načinu rješavanja problema usvojio postupak rješavanja tog problema u njegovoj osnovnoj disciplini (*learning by doing*), i drugo, sagledavanjem konteksta i mogućnosti primjene ovog postupka uvidio je potrebu za njegovim učenjem, što značajno podiže interes učenika za pojedinu temu, kao i motivaciju za učenjem.

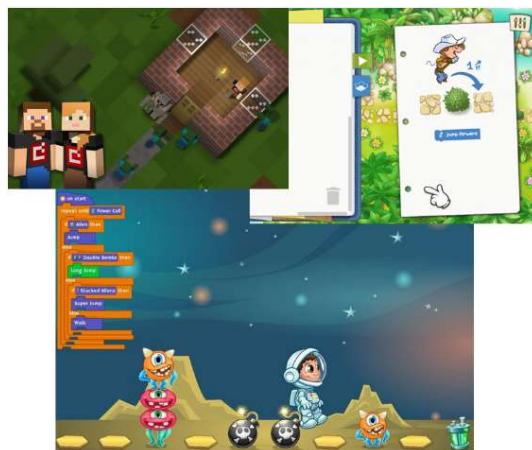
3. Programiranje i programski jezici

Programiranjem nazivamo postupak pisanja računalnog programa, a računalni program je niz naredbi čijim se pokretanjem izvršava neki složeni zadatak. Program se piše u nekom programskom jeziku, a svaki programski jezik ima svoju sintaksu i semantiku. Slika 1. prikazuje najčešću percepciju računalnog programa i postupka programiranja kao kucanja koda na računalu. Program se sastoji od niza naredbi koje je potrebno napisati korištenjem tipkovnice. Jedan pogrešan znak, znak viška ili manjka, pa čak i razmak na pogrešnome mjestu, često dovodi do toga da program ne radi. Programski jezici kao što su Logo i QBASIC dugo su se koristili kao prvi jezici za učenje programiranja. U usporedbi s nekim moćnijim programskim jezicima kao što su C ili Java, ovaj je odabir opravdan. Ipak, učenje programiranja korištenjem ovih programskih jezika vrlo se često svodi na rješavanje zadataka iz matematike ili fizike, u najboljem slučaju geometrije. Učenje sintakse jezika u početnom programiranju često je mukotrpno i samo programiranje je većini učenika osnovnoškolske dobi vrlo apstraktno i stoga teško pojmljivo. Primjena na zadatke koji su dio nastave drugih predmeta također učenicima nije posebno zanimljiva, barem ne u nastavi informatike. Stoga je motivacija učenika za učenjem programiranja općenito na vrlo niskoj razini, s izuzetkom manjeg broja učenika koji imaju izražen interes za programiranje i nastavnika koji znaju kako rješavanje problema učiniti dinamičnim i uzbudljivim.



Slika 1. Programska okruženja za pisanje programa u programskim jezicima Logo, BASIC i JavaScript

U novije vrijeme pojavljuju se programski jezici koji ne zahtijevaju učenje sintakse i znatno su prikladniji za početno učenje programiranja. To su vizualni programski jezici. Slika 2. prikazuje kako izgleda postupak programiranja u nekim od takvih okruženja namijenjenih ranom učenju programiranja. Tipkovnica više nije osnovni alat za pisanje naredbi, nego se naredbe dovlače mišem na mjesto na kojem trebaju biti da bi program izvršavao dani zadatak. Umjesto učenja sintakse i pisanja naredbi, učenik gotovo sve vrijeme provodi u razmišljanju o postupku rješavanja problema, testiranju svojeg rješenja i kritičkom osvrtu na postignuto. S lakoćom uviđa mogućnosti za poboljšanje napisanog programa i vrlo jednostavno implementira izmjene u svoje rješenje. Dobar izbor okruženja prikladnih za rano učenje programiranja može se naći, primjerice, na [4]. Osim što nisu opterećeni sintaksom, dodatna prednost vizualnih programskih jezika nalazi se u promjeni paradigme koja je potrebna za svladavanje ishoda učenja. Naime, više nam nije cilj naučiti učenike kako napisati program za izračunavanje prosjeka zadanog niza brojeva, nego isti obrazovni ishod, npr. razumijevanje programske petlje, postizemo programiranjem lika lijevo-desno na sceni na kojoj se odvija neka aktivnost. Tako zadaci za rano učenje programiranja redovito podrazumijevaju izradu računalnih igara primjerenih dobi predškolaca ili učenika nižih razreda osnovne škole. Dobar primjer je aplikacija kroz koju učenici stvaraju vlastitu verziju poznate igre *Flappy bird* [5]. Takva promjena paradigme u kojoj je cilj programiranja napraviti igru kao prirodnu aktivnost djeteta te dobi dovela je do vidljive promjene programskih okruženja u kojima i sam postupak izrade programa, odnosno programiranje, postaje igra. Uvjereno da se igra, dijete zapravo vrlo učinkovito usvaja osnovne koncepte programiranja i primjenjuje ih u sve složenijim zadacima.



Slika 2. Programska okruženja za početno učenje programiranja za najmlađe: *Minecraft*, *Trči Marko!* i *Tynker*

Ipak, za ozbiljniji pristup učenju programiranja koriste se programski jezici kao što su, primjerice, Scratch i Alice, čije je okruženje prikazano na Slici 3. Scratch [6] je danas najpopularniji jezik za početno učenje programiranja, dok je Alice [7] jedan od primjera jezika primjerenijih početnicima nešto starije dobi. Učenici stvaraju igre i priče na različite teme, potiče se kreativnost i kritičko promišljanje.

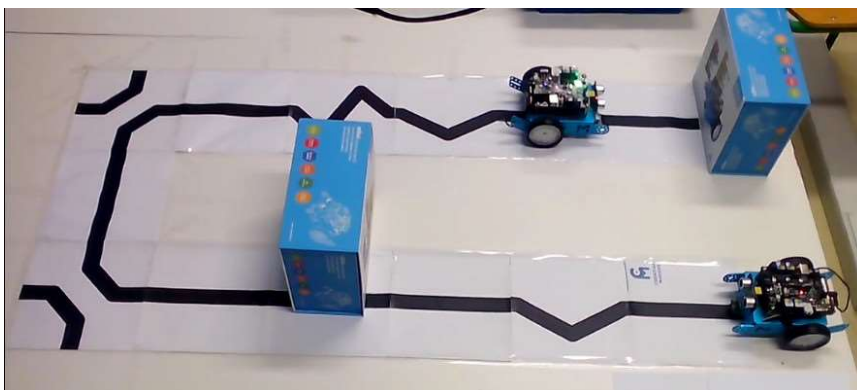


Slika 3. Primjeri kodiranja u programskim jezicima Scratch i Alice

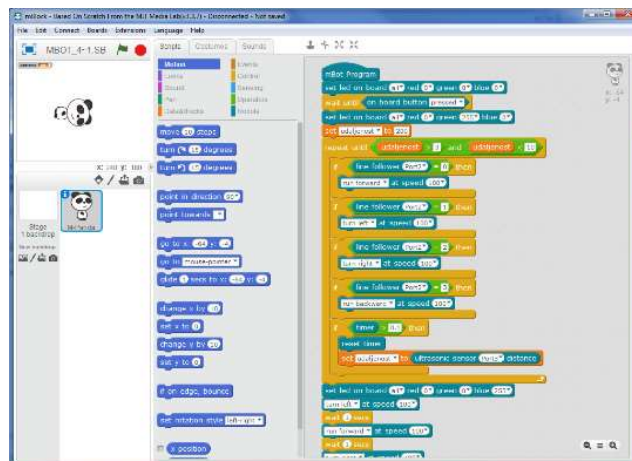
4. Natjecanja kao motivator za početno učenje programiranja

Budući da se programiranje u javnosti percipira kao neka vrlo specifična vještina koja je većini ljudi nepotrebna i uz to se vrlo teško uči, potrebni su veliki naponi da se ideja programiranja kao rješavanja problema prihvati kao sastavni dio svakodnevnog načina razmišljanja i učenja. Veliku ulogu u osvještavanju o važnosti programiranja imaju natjecanja koja se provode među mladima. U ovom će se radu posebno izložiti neka od natjecanja koja su specifično namijenjena najmlađoj populaciji učenika, a to su *Croatian Makers liga* i natjecanje *Dabar*.

Croatian Makers liga je ekipno natjecanje iz robotike za učenike osnovne škole [8]. Inicijativu je 2015. godine pokrenula udruga *Institut za razvoj i inovativnost mladih*, koja intenzivno radi na razvoju digitalne i znanstvene pismenosti, tehnoloških i ostalih kompetencija u okviru STEM područja za mlade u Hrvatskoj i drugim zemljama. *Croatian Makers liga* uključuje 360 škola i udruga kojima je donirano 1800 robota u svrhu edukacije. Učenici najprije sastavljaju svojeg robota, a zatim ga programiraju kako bi izvršavao dane zadatke. Naglasak je na edukativnom radu s djecom, a programiranje se odvija u programskom jeziku temeljenom na Scratchu. Slika 4. prikazuje kako roboti izvršavaju zadatak tijekom petoga kola natjecanja *Croatian Makers lige* u školskoj godini 2016./17. Robot se kreće prateći crtu, uočavanjem prepreke napušta crtu da bi obišao prepreku, a zatim nastavlja pratiti crtu dok ne dođe do druge prepreke. U tom trenutku šalje drugom robotu poruku da krene, te drugi robot ide istom putanjom i zaustavlja se na kraju staze iza prvog robota. Videosnimka ove vožnje, kao i drugih zadataka s natjecanja, može se pogledati na poveznici [9]. Kako izgleda programski kod jednog od robota koji izvršava ovaj zadatak, može se vidjeti na Slici 5.



Slika 4. Roboti izvršavaju zadatak u 5. kolu natjecanja *Croatian Makers Lige* 2016./17.

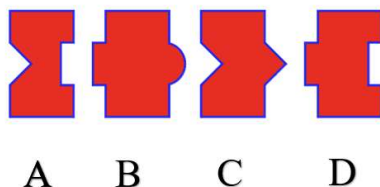


Slika 5. Programski kod jednog od dvaju robota koji u koordinaciji izvršavaju zadatak sa slike 4

Natjecanje *Dabar* [10] drugo je važno natjecanje namijenjeno poticanju interesa učenika za programiranje. Natjecanje je rezultat međunarodne inicijative *Dabar* (Bebras) koja se održava u 58 zemalja svijeta, a od 2016. godine i u Hrvatskoj. Promiče informatiku i računalno razmišljanje među učenicima osnovnih i srednjih škola, njihovim učiteljima i u široj javnosti. O velikom interesu učenika i nastavnika u Hrvatskoj za ovo natjecanje govori činjenica da je u prvoj godini natjecanja sudjelovalo 5900 učenika, što je bilo iznad svih očekivanja, te da se već 2017. uključilo više od 15.000 učenika. Kao članica organizacijskog odbora, autorica se pridružuje rastućem timu volontera koji sudjeluju u pripremi i provedbi ovog natjecanja.

Zadaci na natjecanju *Dabar* nisu ni matematički ni strogo programerski, nego su osmišljeni tako da potiču logičko razmišljanje i razvoj različitih strategija rješavanja problema. Te vještine posebno nastojimo razvijati u poučavanju STEM područja. Računalno razmišljanje je znatno širi pojam od programiranja, iako metodološki često uključuje vrlo slične korake kao i programiranje. Međutim, osim samog postupka iznalaženja rješenja u obliku algoritma, računalno razmišljanje često zahtijeva sagledavanje odnosa među elementima i uočavanje prostornih zakonitosti [11]. Natjecatelji se dijele u pet kategorija ovisno o uzrastu. Zadaci se rješavaju na računalu, a odmah nakon završetka testa učenik dobiva povratnu informaciju o osvojenom broju bodova. Za najmlađe učenike, koji tek svladavaju vještinu čitanja, uveden je audio-vizualni prikaz svakog zadatka. Slika 6. i Slika 7. prikazuju primjere zadataka namijenjenih učenicima prvog i drugog razreda osnovne škole.

Dabrovi slažu slagalicu. Dijelovi slagalice različitih su oblika. Dva dijela mogu se složiti u odgovarajući par.



Koja dva para mogu biti složena istovremeno?

Odaberite jedan odgovor:

- ☐ A i B, C i D
- ☐ A i C, B i D
- ☐ A i D, B i C

Slika 6. Zadatak iz kategorije *Mikro Dabar* (prvi i drugi razred osnovne škole) tipa višestrukog izbora

Četiri dabrova počinu plivati na različitim mjestima označenima na slici. Plivaju prema naprijed i uvijek prate strelice.



Koliko će dabrova stići do jagode?
U prostor za odgovor upiši samo broj.

Odgovor:

Slika 7. Zadatak iz kategorije *Mikro Dabar* (prvi i drugi razred osnovne škole) u kojem se traži unos broja

Za razliku od tradicionalnih školskih natjecanja, primjerice iz matematike, fizike ili stranog jezika, koja se provode hijerarhijski na nekoliko razina (najčešće školsko, županijsko, državno pa međunarodno), natjecanje *Dabar* u svojim osnovnim postavkama ima za cilj biti dostupno svima, a ne samo nadarenima. Ova ideja u skladu je s temeljnim postavkama poučavanja za STEM, gdje se radi na razvoju generičkih vještina koje su prijeko potrebne za uspješno ostvarivanje ishoda učenja u svim školskim predmetima.

U nastojanjima za buđenjem interesa za STEM područje kod mladih, osim natjecanja, vrlo su korisne i druge aktivnosti popularizacije znanosti kao što su *Festival znanosti*, *Noć istraživača* i slične manifestacije. Slika. 8 i Slika 9. prikazuju aktivnosti na *Festivalu znanosti* 2017. na splitskom PMF-u. Učenici svih uzrasta sudjelovali su u igranju edukativne igre *Trči Marko!* koja u obliku animirane slikovnice poučava učenike kako programirati animirani lik Marka da prijeđe zadani put u potrazi za blagom. Cilj je ove radionice da se učenicima približi pojam programiranja i učenja programiranja, koje se uvelike razlikuje od učenja u školi.



Slika 8. Učenici u računalnoj učionici PMF-a na Festivalu znanosti 2017.



Slika 9. *Trči Marko!* – igra za početno učenje programiranja na *Festivalu znanosti* 2017.

Zaključak

U radu su prikazane aktivnosti koje potpomažu poučavanje STEM područja, a provedene su tijekom školske godine 2016./17. na Prirodoslovno-matematičkom fakultetu u Splitu: natjecanje *Croatian Makers liga*, natjecanje *Dabar* te poučavanje programiranja na *Festivalu znanosti*. Osnovni je cilj svih navedenih aktivnosti omogućiti svim učenicima razvoj vještina koje su veoma važne za STEM područje, kao što su komunikacijske vještine, kreativnost, suradnja i kritičko razmišljanje (*Communication, Collaboration, Critical Thinking and Creativity, 4 Cs*). Danas se u svijetu sve više teži integriranom STEM-u, koji se najbolje ostvaruje kroz interdisciplinarni projektni pristup. To znači da učenik s ciljem razvoja složenog projekta, primjerice iz područja ekologije, proučava različite discipline (matematiku, kemiju, fiziku, programiranje...) kako bi stekao znanja koja su mu potrebna za ostvarenje projektnog zadatka. U razvoju ovakvog pristupa poučavanje programiranja i računalnog razmišljanja postaje nužan preduvjet za uspjeh. Veliki iskorak u tom smjeru događa se upravo sada kada se konačno uvodi informatika kao obavezni predmet u petom i šestom razredu osnovne škole počevši od školske godine 2018./19. Istovremeno započinje i eksperimentalna provedba kurikularne reforme u 46 osnovnih i 26 srednjih škola, koje polazi oko 8500 učenika i u kojima je zaposleno 1500 učitelja i profesora. Novim kurikulumima za predmete STEM područja, kao i kurikulumima za međupredmetne teme predviđaju se promjene paradigme poučavanja koje omogućavaju poticajno iskustvo učenja kroz aktivnosti koje potiču razmišljanje, istraživanje i stvaranje [12]. Očekuje se da će najavljene promjene u velikoj mjeri pridonijeti kako obrazovanju za STEM područja, tako i za sva druga područja ljudskog djelovanja.

Literatura

- [1] <https://cuc.carnet.hr/2014/>
- [2] Roy D. Pea, D. Midian Kurland. On the cognitive effects of learning computer programming. *New Ideas in Psychology*, Elsevier, 1984, 2(2), pp.137-168
- [3] Budin, Leo; Brođanac, Predrag; Markučić, Zlatka; Perić, Smiljana; Škvorc, Dejan; Babić, Magdalena. Računalno razmišljanje i programiranje u Pythonu, Zagreb: Element d.o.o., 2017
- [4] <https://www.allcancode.com>
- [5] <https://studio.code.org/flappy/1>
- [6] <https://scratch.mit.edu/>
- [7] <https://www.alice.org/>
- [8] <http://croatianmakers.hr>
- [9] <https://vimeo.com/220714400>
- [10] <http://ucitelji.hr/dabar/>
- [11] Čičin-Šain, Marina; Babić, Snježana; Kralj, Lidija. Ilustracija primjene novog Kurikuluma iz predmeta Informatika i to domene - Računalno promišljanje i programiranje na primjeru metode Početnica Mema za prvi razred osnovne škole. *Proceedings of the 39th International Convention on Information and Communication Technology, Electronics and Microelectronics* / Biljanović, P. (ur.). Rijeka: Croatian Society for Information and Communication Technology, Electronics and Microelectronics, 2016. str. 1258-1261
- [12] <https://mzo.hr/hr/rubrike/nacionalni-kurikulum>

Dr. sc. Jelena Nakić zaposlena je kao poslijedoktorandica na Odjelu za informatiku Prirodoslovno-matematičkog fakulteta Sveučilišta u Splitu. Diplomirala je profesoricu matematike i informatike stekla je na istom fakultetu, a titule magistricu znanosti i doktorice znanosti u području računarstva na Fakultetu elektrotehnike i računarstva Sveučilišta u Zagrebu. Izvodi nastavu iz nekoliko računalnih kolegija na PMF-u i drugim splitskim fakultetima. Njezini znanstveni interesi vezani su uz područje interakcije čovjeka i računala, u kojem posebno proučava aspekte korisničkog iskustva u obrazovnim tehnologijama, tj. u sustavima e-učenja i 3D virtualnim okruženjima za učenje. Sudjeluje u aktivnostima popularizacije znanosti te u organizaciji i provedbi natjecanja iz programiranja i računalnog razmišljanja kao što su *Croatian Makers liga* i natjecanje *Dabar*. Na PMF-u u Splitu pokrenula je *Code Club*, koji je dio svjetske mreže besplatnih volonterskih klubova programiranja za djecu.