

NAPOJ – proti aktivni skupnosti učiteljev računalniških predmetov

Gregor Anželj
Gimnazija Bežigrad
Peričeva ulica 4
Ljubljana

Andrej Brodnik
UL FRI
Večna pot 113
Ljubljana

Matija Lokar
UL FMF
Jadranska ulica 19
Ljubljana

Gregor.Anzelj@gimb.org

Andrej.Brodnik@fri.uni-lj.si

Matija.Lokar@fmf.uni-lj.si

POVZETEK

V prispevku so opisane dejavnosti, katerih namen je z oblikovanjem aktivne učne skupnosti (*Community of practice*) učiteljev računalniških predmetov v srednjih šolah pripomoči k izboljšanju pouka računalništva in informatike v slovenskem šolstvu. Dejavnosti so potekale v sklopu dveh projektov NAPOJ in NAPOJ2.

V okviru projekta NAPOJ smo povezali tri sisteme, ki jih učitelji uporabljajo pri poučevanju programiranja: e-učbenik, spletno učilnico in sistem za samodejno preverjanje pravilnosti programov Tomo. Projekt NAPOJ2, ki je v teku, pa se osredotoča na uporabo fizičnega računalništva z uporabo sistema *Raspberry Pi* kot motivacijskega elementa za poučevanje osnov programiranja.

Izbrana je bila skupina učiteljev imenovanih *mojstri učitelji*, ki so v obeh projektih v sklopu večdnevne delavnice pripravili izhodiščno zbirko učnih gradiv. Temu so pri projektu NAPOJ sledile regijske delavnice, izvedene po vsej Sloveniji, na katerih so mojstri učitelji predstavili gradivo in idejo skupnosti.

V sklopu projekta NAPOJ2 bodo pripravljena gradiva uporabljena na večjem številu šol, čemur bo sledilo več dogodkov. Na njih se bodo srečali tako učitelji kot dijaki, predstavili svoje projekte in si izmenjevali izkušnje glede uporabe fizičnega računalništva kot motivacije in medija za učenje programiranja.

Kategorije in predmetne oznake

K.3.1 [**Computers and education**]: Computer Uses in Education: Computer-assisted instruction.

K.3.2 [**Computers and education**]: Computer and Information Science Education: Computer science education.

Splošni pojmi

Measurement, Experimentation, Languages, Human Factors, Verification

Ključne besede

poučevanje računalništva in informatike, poučevanje programiranja, motivacija

1. UVOD

Predmet Informatika je v slovenskih gimnazijah obvezen eno leto, v treh letih pa je ponujen kot izbirni predmet. Učni načrt je na srečo napisan zelo odprto, pokriva vsa štiri leta in daje učiteljem precej svobode pri izbiri tem. Ta odprtost omogoča, da lahko naredimo

določene spremembe proti še bolj kakovostnemu izobraževanju na področju računalništva in informatike (RIN) [1, 2]. Prvi korak v tem procesu je bil nastanek novega e-učbenika za predmet [3, 4] in določene spremembe pri Maturi [5]. Ob vpeljavi teh sprememb je bilo med učitelji kar nekaj nelagodja glede tega, kaj bodo te novosti pomenile za samo poučevanje. Glede na neuradne razgovore, sporočila v spletnih forumih, razprave na različnih dogodkih in z osebnimi stiki smo ugotovili, da je eno glavnih področij nelagodja to, kako pristopiti k poučevanju programiranja. Prav tako so učitelji izrazili mnenje, da jim manjka primernih učnih gradiv s področja programiranja. Kot kaže kar obširna literatura s tega področja [6 – 12] je eden od možnih odgovorov na tovrstne pomisleke Učna skupnost (*Community of practice*). V tem prispevku poročamo o poskusu, kako smo gradili tako učno skupnost z namenom, da povečamo zaupanje učiteljev in na ta način izboljšamo znanje dijakov s področja programiranja.

2. UČNE SKUPNOSTI

Računalništvo in informatika je relativno precej novo področje v srednješolskem izobraževanju ne le pri nas, ampak tudi v svetu nasploh. Zato se pri samem načinu poučevanja računalništva in informatike srečujemo z mnogimi izzivi. Kot poročajo v [9], se učitelji računalništva in informatike pogosto počutijo osamljene, kar je pogosto združeno tudi s pomanjkanjem zaupanja v svoje učne prakse [11]. Z vzpostavitvijo učnih skupnosti lahko presežemo ta občutek osamljenosti. V [12] je dana naslednja definicija učne skupnosti: »Učna skupnost je skupina ljudi, ki delijo skrb ali strast za določeno področje svojega delovanja in z druženjem dosežejo, da se izboljšajo na tem področju«. Pomemben del učne skupnosti, poleg skupnega zanimanja, je, da so člani praktiki, ki delijo izkušnje, zgodbe, orodja in načine naslavljanja težav. Tako Fincher and Tenenberg v [13, 14] poročata o projektu *Disciplinary Commons*, v okviru katerega se skupina učiteljev s področja računalništva in informatike mesečno sestaja in deli gradiva, izkušnje s področja poučevanja in dokumentira svojo učno prakso. Ni in ostali [9] poročajo o tem, kako je bil tak pristop uspešen pri vzpostavitvi skupnosti srednješolskih učiteljev računalništva in informatike. Eden od najpomembnejših rezultatov tega projekta je bil ta, da so praktično vsi udeleženi učitelji poročali, da so dobili občutek pripadnosti skupnosti. Prav ta pripadnost jim je omogočila, da so ustrezno ocenili svoje delo in izboljšali zaupanje vase kot učitelja računalništva. Zelo uspešne učne skupnosti so se zgradile tudi v sklopu gibanja *Computing At School* (CAS) v Veliki Britaniji. Kot poroča Sentence v [11], projekt CAS prav posebno skrb posveča prav izgradnji lokalnih učnih skupnosti in s tem podpira učitelje pri uvajanju novega učnega načrta v učilnicah.

3. Projekt NAPOJ

V sklopu Googlevega razpisa CS4HS (*Computer Science for High School*) smo dobili sredstva za projekt, katerega cilj je bil doseči večje sodelovanje med učitelji računalništva in informatike, medsebojno izmenjavo gradiv in predvsem izmenjavo izkušenj pri poučevanju računalniških predmetov, sprva s poudarkom na učiteljih v gimnazijah, kasneje pa tudi na ostalih srednjih šolah in osnovnih šolah.

Projekt smo poimenovali NAPOJ (*Načrtovanje poučevanja Algoritmov in Programiranja ter Organizacijske skupnosti*, Slika 1).



Slika 1: Logotip projekta.

3.1 Cilji

Osnovni cilj projekta NAPOJ je bil vzpostavitev aktivne skupnosti učiteljev računalništva in informatike ter jih opremiti s potrebnim gradivom in orodji. Nekoliko ožji cilj je bil osredotočenje predvsem na poučevanje začetnih korakov v programiranju.

V projektu smo se posvetili tudi mnenju učiteljev, da nimajo dovolj znanja, da bi učili prve korake v programiranju in jim pokazati, da z aktivno udeležbo v učnih skupnostih lahko pridobijo samozavest, ki jim po njihovem občutku manjka.

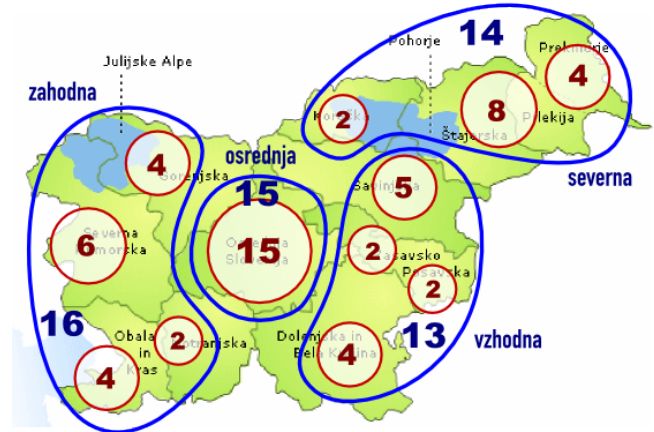
V ta namen smo izvedli tri korake. Najprej smo združili različna gradiva, ki jih učitelji uporabljajo, ali bi jih lahko uporabljali, pri poučevanju programiranja. V drugem koraku smo izbrali skupino mojstrov učiteljev (več o tem v nadaljevanju) in z njimi izvedli večdnevno delavnico, kjer smo skupaj razvili sklop učnih gradiv za neposredno uporabo v razredu pri pouku programiranja. V tretjem koraku so mojstri učitelji izvedli vrsto delavnic po vsej Sloveniji. Z lokalno izvedenimi delavicami smo želeli ne le prilagoditi delavnice potrebam lokalnih skupnosti, ampak tudi vzpostaviti močnejše stike med sodelujočimi v posameznih regijah.

3.2 Mojstri učitelji

Prva ciljna skupina pri gradnji skupnosti so bili vsi profesorji, ki v Sloveniji v gimnazijah učijo predmet Informatika. Glede na poročana spoznanja, predvsem v [9, 11, 13, 14] in delno sledeč modelu, ki je bil uporabljen že v obdobju projekta Računalniško opismenjevanje leta 1995 [15], smo uporabili model *Mojstrov učiteljev (Master Teachers)* [11, 16]. Mojstri učitelji so učitelji, ki so odlično strokovno podkovani na področju računalništva in informatike ter imajo tudi sposobnost in željo, da svoje znanje prenesejo na druge učitelje. Ta model se osredotoča na prenos znanja na lokalni in osebni ravni znotraj stroke. Mojstri učitelji izvajajo izobraževanja za druge učitelje v svojem okolju in nudijo podporo lokalni skupnosti učiteljev. Tako tudi znižujemo stroške,

saj se vse dogaja na lokalni ravni. Še pomembneje pa je, da taka lokalna razdelitev omogoča osebni stik med sodelujočimi.

Obstajajo različni načini, kako prepoznati in izbrati mojstre učitelje. Mi smo kandidate izbrali individualno, na osnovi povabil in 14 med njimi je postalo Mojstrov učiteljev za računalništvo in informatiko. Pri izboru smo želeli doseči tudi ustrezno regijsko pokritost. Tako smo Slovenijo razdelili na štiri območja (gl. Slika 2), ki okvirno vsebujejo približno enako število gimnazij.



Slika 2: Delitev na štiri področja glede na število Gimnazij.

3.3 Gradiva in orodja

V zadnjih letih se je v slovenskem šolskem prostoru pojavilo več spletnih storitev, ki so neposredno uporabne za poučevanje računalništva in informatike, oziroma ožje, za poučevanje programiranja. Za naš projekt je pomembnih dvoje gradiv. Prvo je e-učbenik za informatiko v gimnaziji [3], skupinsko delo več avtorjev, ki bo v kratkem dobil tudi status uradnega učbenika (gl. Slika 3).



Slika 3: E-Učbenik Računalništvo in informatika.

Več o samem učbeniku in razlogih za njegov nastanek v [4].

Poleg tega so na Fakulteti za matematiko in fiziko Univerze v Ljubljani razvili spletno storitev Tomo ([17, 18]) kot pomoč pri poučevanju in učenju programiranja (Slika 4). Storitev omogoča po eni strani takojšnjo povratno informacijo o pravilnosti rešitev programerskih nalog, kot tudi spremljanje napredka sodelujočih.

Izkazal se je kot učinkovit pripomoček pri izvajanju praktičnih vaj iz programiranja. Omogoča, da učenci samostojno rešujejo naloge, katerih pravilnost rešitve preveri sistem sam. S tem dejansko nudi učitelju možnost, da se posveti posameznemu dijaku in ga usmerja pri reševanju. Na ta način lahko posamezen učenec napredujejo v svojem tempu. Zaradi odprtosti in proste dostopnosti jo uporablja že preko 20 šol in fakultet.



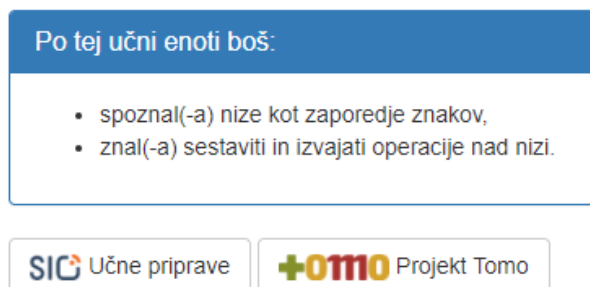
Slika 4: Spletna storitev Projekt Tomo.

Ocenili smo, da bomo ta gradiva najlažje med učitelje razširili tako, da bomo na omrežju SIO, ki ga učitelji že dobro poznajo, v spletni učilnici Moodle pripravili poseben predmet Projekt NAPOJ [19].



Slika 5: Predmet Projekt NAPOJ na SIO [19].

Ta naj bi vseboval različne učne priprave, hkrati pa naj bi bil odskočna deska do ustreznih poglavij v učbeniku in dela zbirke nalog v sistemu Tomo. Prav tako smo želeli ustrezno povezati sistem Tomo in učbenik. Tako smo v učbenik vgradili povezavi (Slika 5)

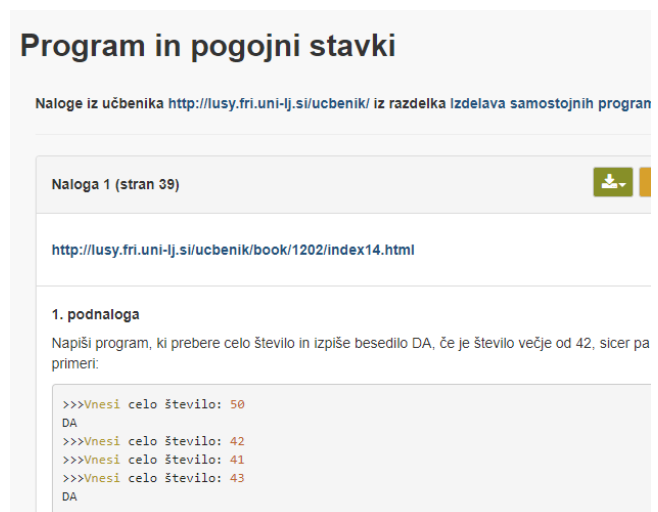


Slika 6: Navezava na spletni storitvi v e-učbeniku.

do spletne učilnice s pripravami in do ustreznega predmeta na spletni storitvi (Slika 6 in Slika 7),



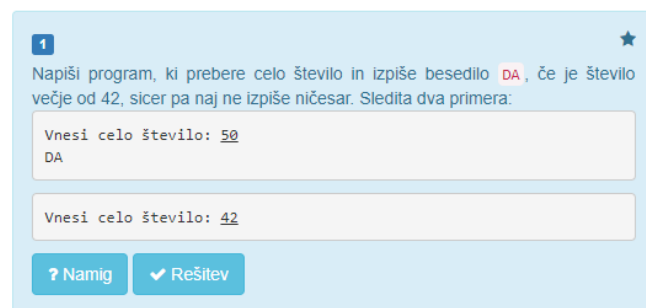
Slika 7: Predmet z nalogami iz učbenika.



Slika 8: Primer naloge iz učbenika v spletni storitvi Tomo.

pri samih nalogah pa smo vgradili povezavo (glej besedilo naloge na Slika 7) do pripadajoče strani v e-učbeniku (Slika 8).

Naloge



Slika 9: Naloga v e-učbeniku.

3.4 Dejavnosti

Še pred koncem šolskega leta 2015/16 smo s kandidati za mojstre učitelje izvedli prvo delavnico, na kateri smo predstavili sam projekt, njegovo idejo ter jih podrobneje spoznali tako z e-učbenikom, kot sistemom Tomo. Dogovorili smo se tudi za dejavnosti preko poletnih počitnic. Glavna dejavnost je bila prenos nalog, ki so v učbeniku, v posebni predmet v sistemu Tomo ter priprava zbirke nalog v sistemu Tomo, primernih za uporabo v srednji šoli.

Na podlagi dejavnosti v tem obdobju smo med kandidati izbrali 14 Mojstrov učiteljev. Z njimi smo konec avgusta izvedli delavnico. Glavni namen delavnice je bil pripraviti učne priprave za prvih šest učnih enot poglavje Programiranje v e-učbeniku. Pripravili smo torej učne priprave za učne enote: *Osnovni koncepti programiranja, Izdelava samostojnih programov in pogojni stavki, Zanke, Tabele, Funkcije in Nizi.*

Predmet Projekt NAPOJ na SIO [19] smo razdelili v skladu z omenjenimi učnimi enotami (Slika 10).



Slika 10: Razdelek v spletni učilnici.

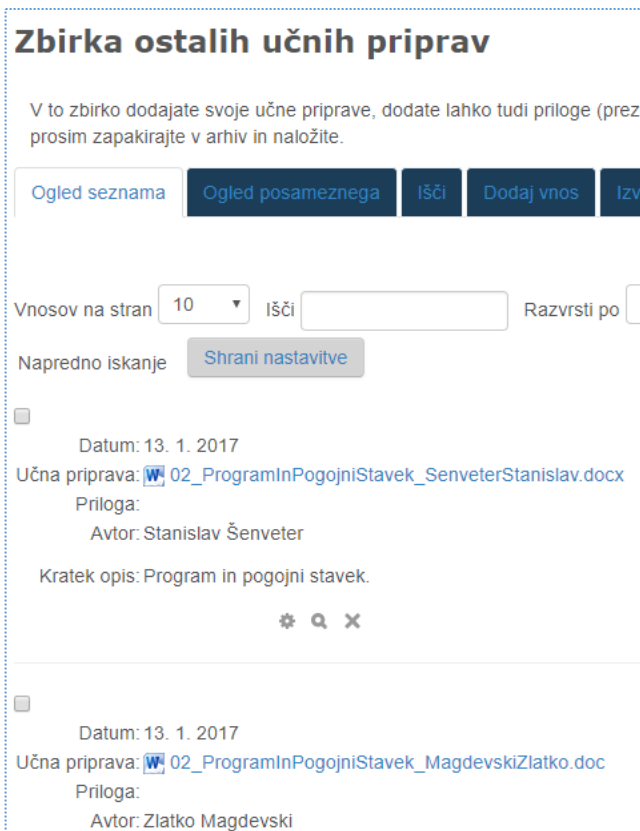
Vsak razdelek v SIO vsebuje povezavo do ustreznega poglavja v učbeniku in do ustreznega sklopa nalog v sistemu Tomo. Prav tako je tam Forum, katerega osnovni namen je pogovor o različnih vidikih poučevanja tega učne enote. Žal Forumi še niso dovolj zaživi, kar kaže, da bomo pri gradnji aktivne skupnosti morali narediti še več korakov.

Razdelek vsebuje tudi primer vzorčne učne priprave za obravnavano učno enoto (Slika 11),

POTEK UČNE URE			
UVODNI DEL: UVAJANJE			
ČAS	UČITELJ	UČENEC	UČNE OBLIKE, METODE, TEHNIKE, UČNI PRIPOMOČKI
7 min	<p>Napoved obravnave nove učne snovi.</p> <p>PNOVITEV Učitelj na tablo napiše problem: Napiši program, ki izpiše prvih 10 naravnih števil, vsako število v svojo vrstico. <i>Učitelj ima možnost, da z učenci ponovi množice števil.</i></p> <p>MOTIVACIJA Učitelj postavlja vprašanja: »Kako bi izpisal prvih 1000, 10000 naravnih števil?« »Kaj se dogaja v vašem programu z ukazom print?« »Kaj se dogaja s številom?« »Kako bi lahko to naredili na krajši način?«</p>	<p>Učenci poslušajo.</p> <p>Učenci rešujejo nalogo na računalniku.</p> <p>Učenci razmišljajo in podajajo rešitve.</p> <p>»Ukaz se ponavlja.« »Števila se povečujejo za 1.« »S programom, ki ponavlja ukaze.«</p>	<p>Frontalno, individualno</p> <p>Pogovor, samostojno delo, razlaga</p>

Slika 11: Izsek iz priprave.

in možnost, da sodelujoči v zbirko dodajajo svoje učne priprave (Slika 12)



Slika 12: Zbirka ostalih učnih priprav.

Naslednji korak projekta je bil izvedba regijskih delavnic. Te so potekale tako, da so Mojstri učitelji, ki so bili zadolženi za posamezno regijo, organizirali lokalno delavnico, na katero so povabili druge učitelje iz lokalnega okolja. Delavnice so potekale v dveh srečanjih. V prvem srečanju so Mojstri učitelji kolegom predstavili vsa orodja (e-učbenik, Tomo in spletno učilnico) ter učne priprave. Nato je sledilo dvo- ali tri-tedensko obdobje, v katerem so udeleženci posamezne regijske delavnice izvedli učne ure s pomočjo pripravljenih učnih priprav. To je pomenilo tudi, da so s svojimi učenci pri pouku uporabljali tudi e-učbenik in spletno storitev Projekt Tomo za takojšnje pridobivanje povratnih informacij pri dejavnosti programiranja.

Obdobju uvajanja in preizkušanja orodij pri pouku je sledilo drugo srečanje. Namen drugega srečanja je bilo vrednotenje uporabe orodij pri pouku s stališča motiviranosti učencev in večjega razumevanja učne snovi programiranja ter posledično boljšega in trajnejšega znanja.

Večina udeležencev regijskih delavnic je menilo, da takšen način poučevanja pomembno pripomore k boljšemu in trajnejšemu znanju dijakov. Pripravljene in v praksi preizkušene učne priprave pripomorejo, da se učitelji, ki občutijo nelagodje pri poučevanju programiranja, kljub temu lažje odločijo za poučevanje programiranja.

Kot dejavnost, ki naj bi pripomogla h gradnji aktivne učne skupnosti, smo se večkrat tudi srečali v ARNESovih VOX konferencah, kjer smo se pogovarjali o različnih temah v povezavi s poučevanjem programiranja in računalništva nasploh.

Ocenjujemo, da so tovrstne redne spletne konference lahko izredno dobrodošel način h gradnji aktivne skupnosti. Težava je v tem, kako

zagotoviti njihovo redno izvajanje. Seveda pa je še vedno potrebno srečevanje v živo, saj vsi v literaturi navedeni primeri gradenj skupnosti [6, 7, 9, 10, 11, 13, 14] navajajo pomembnost osebnega srečevanja članov skupnosti.

4. NAPOJ 2

Uspešen zaključek projekta NAPOJ nas je spodbudil, da smo se v naslednjem letu (torej za obdobje 2017/18) ponovno prijavili na razpis podjetja Google. Osnovni cilj projekta je ostal enak, le da smo tokrat želeli dijake motivirati za programiranje na drugačen način. Odločili smo se, da bi učiteljem pomagali navduševati dijake za programiranje tako, da bi jih opremili z gradivi s področja fizičnega računalništva (Slika 14). Nadaljevali smo v obliki, ki smo jo vzpostavili že v projektu NAPOJ. Še vedno pa je bil prisoten naš cilj gradnje aktivne skupnosti učiteljev.



Slika 13: Logotip projekta NAPOJ2.

Nadaljevali smo torej tam, kjer smo se ustavili pri projektu NAPOJ in sicer kako z različnimi pristopi doseči, da bi dijaki na programiranje gledali kot na privlačno in zanimivo dejavnost. V pogovorih je bila večkrat omenjena možnost uporabe računalnika *Raspberry Pi* (malina) s sklopom vhodno/izhodnih naprav. Sam projekt smo pripravili v povezavi z Zavodom 404 (<https://404.si/>), ki je tudi preskrbel potrebne complete strojne opreme in sodeloval pri izvedbi delavnic.

Skupina mojstrov učiteljev je ostala več ali manj nespremenjena. Le namesto tistih kolegov, ki zaradi različnih razlogov niso mogli sodelovati v NAPOJ2, se nam je pridružilo nekaj novih učiteljev.

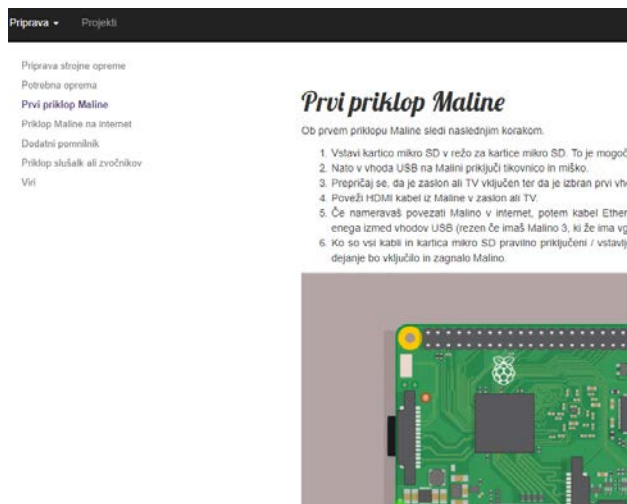
4.1 Dejavnosti pri NAPOJ2

V mesecu avgustu smo ponovno izvedli delavnico z mojstri učitelji. Na njej smo razvili vrsto gradiv, katerih del smo združili v e-učbenik ([20], naslovnico prikazuje Slika 15).



Slika 14: E-učbenik za uvod v fizično računalništvo.

V uvodnem delu (priprava) je napisano najnujnejše glede priprave strojne opreme in namestitve programske opreme (Slika 16).



Slika 15: Navodila za priključ Raspberry Pi.

Prav tako je na kratko predstavljeno fizično računalništvo kot tako, ter kako s programi v programskem jeziku Python uporabljamo Malinine GPIO nožice za krmiljenje elektronskih komponent, kot so na primer svetleče diode (LED), gumbi, brenčaci itd. (Slika 17)

Fizično računalništvo s pironom
GPIO nožice
Prižiganje LED diode
Prižiganje in ugaščanje LED diode
Utriganje LED diode
Uporaba gumba za vhodno podatke
Ročno krmiljenje LED diode
Izdelava stikala
Viri

Fizično računalništvo s pironom



Ta vodič te bo naučil, kako uporabljati Malinine GPIO nožice (angl. pins) na primer svetleče diode (angl. light-emitting diodes – LED), gumbi, brenčaci

Slika 16: Izsek iz učbenika.

Glavnino učbenika sestavlja opis vrste projektov. Pripravljena je tudi razpredelnica (Slika 18), ki kaže, katere učne teme s področja programiranja pokriva posamezen projekt.

Dejavnost	Projekt	Osnove	Vejitve	Zanke	Tabele	Funkcije	Nizi	Navezave
4-številčni 7-segmetna prikazovalnik	Igra Simon	✓		✓				generator naključnih števil in naključna števila
Črna različnost	Primeri signalizacija	✓	✓					komunikacije, vrste in oblike komunikiranja
Mišni zvonec	Glavna skema	✓	✓					zvočni zvok v računalniku, formati zvočnih datotek
Igra Simon	Igra Simon	✓	✓	✓	✓	✓		generator naključnih števil in naključna števila

Slika 17: Del seznama projektov.

Na sami delavnici (in tudi kasneje preko sodelovanja na daljavo) so mojstri učitelji pripravili vzorčne učne priprave, ki kažejo, kako s temi gradivi izvesti posamezno učno uro ali njihov sklop. Slika 19 in Slika 20 prikazujeta izseka iz priprav.

OSNOVNI PODATKI
Šola:
Letnik: 1. letnik
Datum:
Predmet: Informatika
Učna tema: Obdelava podatkov
Učna enota: Osnovni koncepti, vejitve, zanke, tabele, funkcije, nizi
Učne oblike:
- frontalno, individualno
Učne metode:
- razlaga, pogovor, demonstracija, reševanje problemov
Predznanje dijaka:
- programski jezik Python, dvojiški številski sistem
Operativni učni cilji
Ob koncu učne ure dijak:
- zna napisati funkcijo, ki pretvori desetiško v dvojiško število
- zna prestreči prekinitveno sistemsko funkcijo CTRL-C z drugo funkcijo
- zna napisati programske izbire
Učna sredstva:
- Učila: e-učbenik (http://lusy.fri.uni-lj.si/ucbenik/), Malina in piton (https://anzeljg.github.io/rpi/book/)
- Učni pripomočki: računalnik, projektor, projekcijsko platno, zveze pripadajočo opremo ...
Didaktične etape učnega procesa:
1. pripravljanje ali uvajanje

Slika 18: Izsek iz priprave za učno enoto osnovni koncepti, vejitve.

Projekt: Ugibanje števila

Avtor: Marina Trost

Datum: avgust, 2017

Letnik: 1. ali 2.

Število ur: 2 uri

Predznanje:

- povezava lučk na RaspberryPi,
- prižiganje in ugašanje lučke s programskim jezikom Python na Ra
- vnos števila v programskem jeziku Python
- poznavanje spremenljivk v programskem jeziku Python

Učna sredstva:

- računalniki, projektor
- RaspberryPi s 3 led diodami (1 zelene barve) - za dva dijaka 1 ko

1. ura

Učna tema: vejitve in zanke

Naloga:

Računalnik si izmisli neko število npr. Od 0 do 32.

Napiši program za uganjanje tega števila, tako, da ena (višja) dioda posve

Slika 19: Izsek iz priprave za učno enoto vejitve in zanke.

Ponovno smo pripravili spletno učilnico ([21], Slika 21) z opisom projekta, gradivi in forumi za razprave. V njej dostopna gradiva so bila razvita v sklopu delavnice in samostojnega dela sodelavcev projekta. V to učilnico smo povabili vse učitelje.



Slika 20: Spletna učilnica projekta NAPOJ2 [18].

4.2 Nadaljnji koraki

Vsekakor v Sloveniji potrebujemo aktivno učno skupnost učiteljev računalniških predmetov. Podobne izkušnje v tujini dokazujejo velik napredek v kakovosti poučevanja, če je omenjena skupnost prisotna v šolskem prostoru (prim. CAS v Veliki Britaniji, CS4All v ZDA in podobno). Upamo, da bodo dejanskosti, ki smo jih že izvedli in ki jih še načrtujemo vsaj malo pripomogli k gradnji te skupnosti.

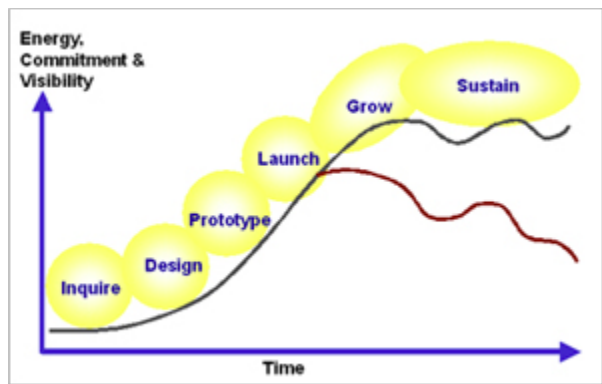
V septembru in oktobru bo delo potekalo po šolah, kjer se bodo dijaki učili programirati ob fizičnem računalništvu. Delo je nivojsko, saj se na osnovni ravni uporablja običajni računalnik, na napredni ravni pa Malinin sistem. Poleg tega delo poteka po skupinah, kjer vsaka skupina pripravlja svoj projekt.

V novembru načrtujemo Malinino društvo po regijah. Na teh druženjih dijaki predstavijo svoje delo in ob tem obiskujejo delavnice. Na delavnicah jim mentorji pomagajo pri delu na njihovih projektih (napredna raven) in jih seznanjajo s fizičnim računalništvom (osnovna raven).

Delo do meseca aprila poteka naprej po šolah in v mesecu aprilu ali maju bo zaključni dogodek, kjer bodo dijaki predstavili svoje projekte. Namen tega dogodka ni tekmovanje, ampak srečanje in predvsem izmenjava idej ter spretnosti.

5. ZAKLJUČKI

Dejansko smo pri gradnji aktivne skupnosti učiteljev računalništva in informatike prehodili šele prve korake. Kot lahko vidimo na sliki Slika 22, smo



Slika 21: Rast skupnosti ([22]).

trenutno v fazi, ko moramo zelo paziti, da naša nadaljnja pot ne bo šla v smeri rdečega dela grafa. Brez dvoma je pri tem ključna ne samo vloga vodij projekta, ampak predvsem samih učiteljev v skupnosti. Učitelji morajo sami pripoznati, da jim skupnost pomaga pri njihovem delu. Pri tem bodo odigrali ključno vlogo Mojstri učitelji.

6. ZAHVALA

Projekta NAPOJ in NAPOJ2 sta bila finančno podprta s strani podjetja Google v sklopu pobude Computer Science for High School (CS4HS). Projekta bi bila neuspešna brez mojstrov učiteljev: Gregor Anželj, Lidija Babič, Klemen Bajec, Anamarija Cencelj, Maja Čelan, Marko Kikelj, Melita Kompolšek, Karmen Kotnik, Gabrijela Krajnc, Urban Kržan, Nastja Lasič, Zlatko Magdevski, Andrej Peklar, Grega Rekel, Stanislav Šenveter, Marina Trost, Romana Vogrinčič in Matej Zdovc.

7. LITERATURA

- [1] Tomazin, M., Brodnik, A.: Učni cilji pouka računalništva v osnovni šoli - slovenski in ACM K12 kurikulum. Organizacija: revija za management, informatiko in kadre, 2007, 40 (6), str. A173-A178
- [2] Mori, N., Brodnik, A., Lokar, M.: Development of CS curriculum for secondary schools through changes in external examination and textbooks. In: Proceedings of the IFIP TC3 Joint Conference Stakeholders and Information Technology in Education, str. 58. University of Minho, Guimares (2016), <http://saite2016.dsi.uminho.pt/wp-content/uploads/2016/06/Book-of-Abstracts.pdf>
- [3] Anželj, G., Brank, J., Brodnik, A., Bulič, P., Ciglarich, M., Dukič, M., Fürst, L., Kikelj, M., Krapež, A., Medvešek, A., Mori, N., Pančur, M., Sterle, P.: Računalništvo in informatika v2.10, E-učbenik za informatiko v gimnaziji <http://lusy.fri.unilj.si/ucbenik/>
- [4] Mori, N., Lokar, M.: A new interactive computer science textbook in Slovenia. V: Proceedings of Informatics in Schools: Improvement of Informatics Knowledge and

Perception: 9th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2016., str. 167–178,

https://link.springer.com/chapter/10.1007/978-3-319-46747-4_14

- [5] Anželj, G., Batagelj, V., Brodnik, A., Kikelj, M., Krapež, Lasič, N., Wechsterbach, R.: PIK: Predmetni izpitni katalog za splošno maturo 2017 – informatika, <http://www.ric.si/mma/2017%20M-INF-2017/2015083113004713/>
- [6] Chalmers, L., Keown, P.: Communities of practice and professional development. *International Journal of Lifelong Education* 2(25), str. 139–156 (2006).
- [7] Corso, M., Giacobbe, A.: Building communities of practice that work: a case study based research. In: *The Sixth European Conference on Organizational Knowledge, Learning, and Capabilities*, str. 17–19. Bentley College, Waltham (2005).
- [8] CUREE. Centre for the use of research evidence in education: Understanding what enables high quality professional learning (a report on the research evidence), [http://www.curee.co.uk/files/publication/\[site-timestamp\]/CUREE-Report.pdf](http://www.curee.co.uk/files/publication/[site-timestamp]/CUREE-Report.pdf)
- [9] Ni, L., Guzdial, M., Tew, A. E., Morrison, B., Galanos, R.: Building a community to support HS CS teachers: the disciplinary commons for computing educators. V: *Proceedings of the 42nd ACM technical symposium on Computer science education (SIGCSE 11)*, ACM, str. 553–558, (2011).
- [10] Sentance, S., Humphreys, S.: Online vs face-to-face engagement of computing teachers for their professional development needs. V: *Proceedings of the 8th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives (ISSEP 2015)* (2015), Springer International Publishing, str. 69–81.
- [11] Sentance, S., Humphreys, S., Dorling, M.: The network of teaching excellence in CS and master teachers. V: *Proceedings of the 9th Workshop in Primary and Secondary Computing Education*, ACM, str. 80–88 (2014).
- [12] Wenger-Trayner, E., Wenger-Trayner, B.: Introduction to communities of practice, A brief overview of the concept and its uses. Wenger-Trayner, 2015.
- [13] Fincher, S., Tenenberg, J.: Warren's question. In: *Proceedings of the Third International Computing Education Research Workshop*, ACM, str. 51–60, (2007).
- [14] Tenenberg, J., Fincher, S.: Opening the door of the computer science classroom: The disciplinary commons. In: *Proceedings of the 38th SIGCSE technical symposium on Computer science education*, ACM, str. 514–518, (2007).
- [15] Batagelj, V., Rajkovič, V., Namen, cilji in smernice programa Računalniško opismenjevanje – RO, november 1995, <http://www.educa.fmf.uni-lj.si/ro/izomre/novice/doc/vizija.htm>
- [16] CAS Master Teachers, Network of Excellence Computer Science Teaching, http://www.computingschool.org.uk/custom_pages/36
- [17] Tomo, samodejni sistem za preverjanje pravilnosti programskih nalog, 2015, <https://www.projekt-tomo.si>

- [18] Lokar, M., Pretnar, M.: A Low Overhead Automated Service for Teaching Programming, V Proceedings of the 15th Koli Calling Conference on Computing Education Research, Koli, Finland 2015, <http://doi.acm.org/10.1145/2828959.2828964>
- [19] Projekt NAPOJ (CS4HS), spletna učilnica, <https://skupnost.sio.si/course/view.php?id=9376>
- [20] Anželj G., Brodnik A., Capuder R., Lokar M., Malina in piton, E-učbenik za uvod v fizično računalništvo v1.0, <https://anzeljg.github.io/rpi/book/index.html>
- [21] NAPOJ-2: Fizično računalništvo, spletna učilnica, <https://skupnost.sio.si/course/view.php?id=9449>
- [22] Cambridge D., Kaplan S., Suter V., Communities of Practice Design Guide, Step-by-Step Guide for Designing and Cultivating Communities of Practice in Higher Education, 2005, <https://library.educause.edu/resources/2005/1/community-of-practice-design-guide-a-stepbystep-guide-for-designing-cultivating-communities-of-practice-in-higher-education>