



Sveučilište u Rijeci
University of Rijeka
<http://www.uniri.hr>

Polytechnic: Journal of Technology Education, Volume 5, Number 1 (2021)
Politehnika: Časopis za tehnički odgoj i obrazovanje, Volumen 5, Broj 1 (2021)



Politehnika
Polytechnica
<http://www.politehnika.hr/journal>
cte@uniri.hr

DOI: <https://doi.org/10.36978/cte.5.1.2>

Prethodno priopćenje
Preliminary note
UDK: 37.018.43:004.42
004.42:37.018.43

Uloga korištenja alata za vizualizaciju na razumijevanje algoritama sortiranja u online nastavi

Monika Mladenović, Lucija Udiljak, Ivica Boljat

Prirodoslovno-matematički fakultet, Sveučilište u Splitu

Ruđera Boškovića 33, 21000 Split

monika.mladenovic@pmfst.hr, lucija.udiljak@skole.hr, boljat@pmfst.hr

Sažetak

Učenje i poučavanje algoritama sortiranja je vrlo zahtjevno kako za učitelja tako i za učenike. Problem može biti još veći kada se nastava u cijelosti provodi u online okruženju što je bio slučaj u Republici Hrvatskoj od sredine ožujka do kraja školske godine 2019/20 zbog COVID-19 pandemije. Korištenje alata za vizualizaciju može pomoći u spuštanju razine apstrakcije pri poučavanju složenih koncepata kako bi učenici lakše razumjeli rad algoritama. U svrhu ispitivanja utjecaja korištenja alata za vizualizaciju tijekom online nastave provedeno je istraživanje u travnju i svibnju 2020. godine u dva treća razreda (n=52) jedne Prirodoslovno-matematičke gimnazije. U eksperimentalnoj grupi koristili smo alate za vizualizaciju VisuAlgo i Python Tutor za poučavanje algoritama sortiranja i implementaciju istih u Pythonu. Osim toga su ispitani i stavovi učenika o aktualnom provođenju online nastave. U ovom radu su opisani provedba i rezultati navedenog istraživanja.

Ključne riječi: *Vizualizacija; online nastava; algoritmi sortiranja; selection sort; informatika.*

1 Uvod

Poučavanje programiranja općenito je od samih početaka izazovno za učenike tako i za učitelje. Programiranje zahtijeva visoku razinu apstrakcije (Koppelman & Van Dijk, 2010; Statter & Armoni, 2016, 2017). koja kod mlađih učenika često još uvijek nije dovoljno razvijena. Kako onda poučavati programiranje mlađem uzrastu? Osnovna ideja je spustiti razinu apstrakcije kako bi učenicima omogućili „lakši pristup“ apstraktnim pojmovima. Takav pristup se naziva „od konkretnog prema apstraktnom“ (Dann & Cooper, 2009). Jedan način koji bi mogao unaprijediti njihovo razumijevanje je uvođenje vizualizacija u nastavu kako bi apstraktne koncepte učinili konkretnijima (Korhonen & others, 2003).

Jedna od osnovnih poteškoća u učenju koncepata programiranja je nedostatak ispravnih mentalnih modela (Sorva, 2013). Nedostatkom mentalnih modela učenici često program doživljavaju kao „crnu kutiju“ u kojoj se događa nešto što njima nije vidljivo (Mladenović, Žanko & Aglič Čuvić, 2020). Vizualizacija može pomoći učenicima u razjašnjavanju i ispravljanju mentalnih modela pojmovnog stroja (Cunningham, Blanchard, Ericson, & Guzdial, 2017) te „crnu kutiju“ učiniti „staklenom“ na način da učenici „vide“ što se događa u programu (Mladenović et al., 2020).

Algoritmi sortiranja se prema kurikulu u prirodoslovno-matematičkim gimnazijama obrađuju u trećem razredu (Ministarstvo znanosti i obrazovanja, 2018). Kako se nastava od ožujka do kraja školske godine 2019/2020 u svim školama u Republici Hrvatskoj zbog pandemije održavala u potpunosti online (Ministarstvo znanosti i

obrazovanja Republike Hrvatske, 2020), tako su se nastavne teme vezane uz algoritme sortiranja održale u online obliku, što dodatno otežava razumijevanje ali i poučavanje ovako kompleksnog gradiva.

U ovom radu je opisano istraživanje provedeno u dva treća razreda jedne prirodoslovno-matematičke gimnazije tijekom online nastave u školskoj godini 2019/2020.

2 Poučavanje algoritama sortiranja

Algoritmi sortiranja se provode u okviru domene Računalno razmišljanje i programiranje u drugim i trećim razredima prirodoslovno-matematičkih gimnazija, gdje se za treće razrede naglašava uspoređivanje različitih algoritama sortiranja (Ministarstvo znanosti i obrazovanja, 2018). Algoritmi sortiranja koji se obrađuju su: sortiranje izborom najmanjeg elementa (engl. *selection sort*), sortiranje zamjenom susjednih elemenata (engl. *bubble sort*) i sortiranje umetanjem (engl. *insertion sort*), dok su složeniji algoritmi primjerice sortiranje spajanjem (engl. *merge sort*) i brzo sortiranje (engl. *quick sort*).

Poučavanju algoritama sortiranja se može pristupiti na više načina ovisno o samom algoritmu koji se poučava, dobi učenika te razini obrazovanja. Od učenika prirodoslovno-matematičkih gimnazija se očekuje da razumiju, opišu i primijene neke algoritme sortiranja. Za spuštanje razine apstrakcije se mogu koristiti metode vizualizacije u koje spadaju aktivnosti kao što su skiciranje (Cunningham et al., 2017), korištenje animacija, te korištenje različitih alata za vizualizaciju algoritama i/ili koda (Mladenović et al., 2020). Prema navedenom, pri poučavanju algoritama sortiranja se osim samog kodiranja u nekom programskom jeziku mogu koristiti statičke slike (ispisom stanja elemenata niza nakon svakog prolaza sortiranja), te animacije ili simulacija. Računalna implementacija algoritama sortiranja može zahtijevati primjenu struktura podataka, korištenje ugniježđenih petlji, rekurzivne postupke i sl. Zbog toga je njihovo kodiranje mentalno zahtjevan izazov za učenike, pa korištenje vizualizacijskih alata koji prikazuju izvršavanje algoritma korak po korak može biti jako korisno u poučavanju (Bernát, 2014).

3 Vizualizacija u nastavi

Vizualizacija može učiniti jasnim značenje nekog apstraktnog entiteta, što bi inače bilo teško otkriti nekim drugim sredstvima (Gómez-Henríquez, 2001).

Softverska vizualizacija za poučavanje programiranja sastoji se od dvije domene: programska vizualizacija (PV) i algoritamska vizualizacija (AV) (Price, Baecker, & Small, 1993). PV

se odnosi na vizualizaciju izvršavanja algoritma korak po korak, dok se AV odnosi na vizualizaciju stanja struktura podataka za vrijeme izvršavanja algoritma.

Prema prvoj opsežnoj meta-analizi po pitanju korištenja AV u nastavi (Hundhausen, Douglas, & Stasko, 2002) zaključeno je da je najučinkovitije korištenje AV tehnologije u slučaju kada njeno korištenje aktivno uključuje učenike u proces učenja algoritama. To bi primjerice bile aktivnosti kao što su: „što ako“ analize ponašanja algoritama, vježbe predviđanja, vježbe programiranja i sl. Istraživanje (Jarc, Feldman, & Heller, 2000) je pokazalo da interaktivno predviđanje najviše pomaže u težim pitanjima i onima koji uključuju animaciju umjesto čistu vizualizaciju.

Navedena istraživanja su pokazala pozitivan utjecaj vizualizacije u slučaju kada ona potiče aktivno učenje. Upravo je aktivno učenje temeljno načelo kognitivnog konstruktivizma. Ovaj pristup tvrdi da pojedinci konstruiraju svoje vlastito individualno znanje iz svojih subjektivnih iskustava o svijetu. Vizualizacija je učenicima izvor za konstrukciju znanja, a aktivan angažman s njome omogućava konstruiranje vlastitog razumijevanja (Hundhausen et al., 2002).

Ipak, pokazalo se da vizualizacija nije uvijek učinkovita te treba voditi računa o načinu primjene u nastavi što ovisi o složenosti programskog koncepta koji se obrađuje, dobi učenika te razini školovanja. Utvrdilo se da neki studenti kada je interaktivno predviđanje prisutno smatraju učenje igrom, te se fokusiraju na zabavu, a ne na učenje (Jarc et al., 2000). Iako vizualizacija može spustiti razinu apstrakcije potrebnu za razumijevanje poučavanog pojma te tako olakšati učenicima savladavanje istog, može se dogoditi suprotan efekt gdje učenici u tom slučaju ulažu manji kognitivni napor zbog čega se usvojeno znanje ne zadržava tijekom vremena (Mladenović et al., 2020).

Ključnu ulogu bi mogla imati aktivnost učenika tijekom korištenja vizualizacije. (T. L. Naps et al., 2002) definiraju Taksonomiju uključenosti (engl. Engagement Taxonomy) kojom opisuju razinu aktivnosti učenika pri korištenju vizualizacije. Definirali su šest različitih formi učeničke uključenosti u vizualizacijsku tehnologiju, a (Myller, Bednarik, Sutinen, & Ben-Ari, 2009) su proširili predloženu taksonomiju s četiri nove aktivnosti. Prijelazi među kategorijama se ne smatraju strogo definiranim, tj. ovih deset formi uključenosti ne čine jednostavnu hijerarhijsku strukturu. U učenju se teži postizanju što više formi uključenosti.

Korištenje vizualizacija u obrazovanju može biti imati pozitivan utjecaj na usvajanje učenih pojmova, jer studenti mogu brže i dublje razumjeti osnove te lakše doseći više dimenzije kognitivnog procesa (Grissom, McNally, & Naps, 2003). Ključna zapreka

usvajanju vizualizacije od strane predavača je vrijeme potrebno za naučiti, instalirati, razviti vizualizaciju i tada je integrirati u kolegij. (Boljat & Pletikosa-Grgić, 2013) navode opiranje nastavnika promjenama koje pomiču fokus učenja s nastavnika na učenika i želju da zadrže autoritet kao uzrok nekonzistentnosti njihovih pozitivnih reakcija prema inovacijama. (T. Naps et al., 2003) smatraju da se neke od glavnih zapreka za korištenje vizualizacije u poučavanju mogu prevladati omogućavanjem visoko kvalitetnih pomoćnih materijala za instruktore. Dostupnost dobrih pomoćnih materijala vodi prema zadovoljstvu nastavnika, što posljedično može utjecati na šire korištenje vizualizacije.

4 Metodologija istraživanja

Predmet ovog istraživanja je utjecaj korištenja vizualizacije na poučavanje u online nastavi. Cilj je utvrditi učinkovitost primjene vizualizacije u poučavanju algoritama sortiranja, s naglaskom na *selection sort*. Osim navedenog cilja je utvrditi stavove učenika općenito o provođenju online nastave. U skladu s navedenim ciljevima postavljeno je istraživačko pitanje: „Kakav je stav učenika prema trenutnom provođenju online nastave?“

Osim istraživačkog pitanja postavljamo i hipotezu H1: „Učenici koje se poučava primjenom vizualizacije ostvarit će bolje rezultate u aktivnostima online nastave u kojima se ispituje usvojenost *selection sort*-a u odnosu na učenike koji se poučavaju na klasičan način, bez vizualizacije“.

3.1 Kontekst istraživanja

Istraživanje je provedeno kao kvazi-eksperiment (Cohen, Manion, & Morrison, 2013) za vrijeme online nastave u dva treća razreda jedne prirodoslovno-matematičke gimnazije tijekom školske godine 2019./2020. Jedan razred se poučavao uz primjenu vizualizacijskih alata (Eksperimentalna grupa), a drugi na klasičan način (Kontrolna grupa). Poučavanje se temeljilo na videolekcijama koje su za obje grupe izrađene u skladu s pristupima poučavanja. Videolekcije u Eksperimentalnoj grupi opisuju algoritme sortiranja uz primjenu vizualizacijskih alata, dok ih u Kontrolnoj grupi opisuju na klasičan način, opisivanjem rada algoritama i zapisivanjem stanja elemenata niza. Alati korišteni za vizualizaciju algoritama sortiranja u video-lekcijama su Visualgo¹ za algoritamsku vizualizaciju (AV) i Python Tutor² (Guo, 2013) za programsku vizualizaciju (PV).

Svi ostali sadržaji, kao što su tekstualni sadržaji, pitanja, zadaće i kviz su za obje grupe bili

ekvivalentni. Zbog nedostatka vremena i brojnih obaveza učenika u online nastavi, obrađena je primjena algoritama samo za nizove brojeva. Materijali koji su bili predviđeni za obradu su prvo pripremljeni kao tekstualni dokument u kojem su bile dostupne poveznice na izrađene videolekcije, objašnjenja gradiva i zadaci, te su kao takve pregledale dvije nastavnice informatike. Na temelju pripremljenih i odobrenih materijala organizirane su aktivnosti na platformi Loomen. Materijali su organizirani u obliku lekcija i raspodijeljeni po rasporedu nastave informatike. Nastava je u konačnici organizirana kroz četiri aktivnosti. Nakon izvršavanja svih aktivnosti učenici su dobrovoljno mogli ispuniti anketu za ispitivanje stavova o provedenoj nastavi. U tablici 1 opisana je svaka od provedenih aktivnosti.

Aktivnost	Naziv	Vrsta aktivnosti	Opis
A1	Sortiranje 1	Lekcija	<i>Selection sort</i> - 3 kraće videolekcije, kratka tekstualna objašnjenja, pitanja.
A2	Zadaća - zadaci	Zadaća	2 zadatka za predaju, forum za diskusiju.
A3	Sortiranje 2	Lekcija	Ostali algoritmi - 1 videolekcija, kratka tekstualna objašnjenja, pitanja.
ZT	Provjera – kviz	Aktivni kviz	Završni test - provjera u realnom vremenu.

Tablica 1 Opis aktivnosti

U aktivnostima A1 i A3 su se koristile videolekcije.

U tablici 2 se nalazi pregled korištenih videolekcija za kontrolnu (K) i eksperimentalnu (E) skupinu.

Video lekcija	Opis	E grupa - poveznice na videolekcije	K grupa - poveznice na videolekcije
A1-V1	Objašnjenje koncepta algoritma <i>selection sort</i> .	V1-E - video	V1-K - video
A1-V2	Objašnjenje implementacije <i>selection sort</i> a u Pythonu.	V2-E - video	V2-K - video
A1-V3	Postupak rješavanja zadatka.	V3 - video	
A3-V4	Objašnjenje koncepta ostalih algoritama.	V4-E - video	V1-E - video

Tablica 2 Opis video-lekcija

Zbog online nastave te redovnog rasporeda učenika nisu sve aktivnosti održane u isto vrijeme za sve grupe što je moglo utjecati na rezultate.

¹ <https://visualgo.net/en>

² <http://pythontutor.com/>

Istraživanje je provedeno od 20. 4. 2020. do 30. 4. 2020. ukupno u 5 termina (blok satovi). Aktivnost A1 se obrađivala u dva dana unutar iste grupe/razreda (podgrupe EA i EB, KA i KB). Zbog toga su pitanja unutar lekcija neznatno promijenjena kako bi se smanjila mogućnost prepisivanja. U analizi navedene aktivnosti su se dodatno uspoređivali rezultati podgrupa. U tablici 3 su prikazani datumi provođenja aktivnosti.

Datum	Eksperimentalna grupa	Kontrolna grupa
20. 4. 2020.	A1 (grupa A)	A1 (grupa A)
22. 4. 2020.	A1 (grupa B) A2	A1 (grupa B) A2
23. 4. 2020.	A3	A3
29. 4. 2020.	-	ZT
30. 4. 2020.	ZT	-

Tablica 3 Raspored provođenja aktivnosti

U postavkama lekcije Sortiranje 1 (A1) ograničeno je određivanje lekcije na jedan pokušaj i maksimalno vrijeme obrade lekcije je postavljeno na 90 minuta. Ipak, nakon primjedbi učenika na vremensko ograničenje, zbog sporijeg interneta kod nekih, kao i sporijeg rada Loomena kao posljedica opterećenja, postavljeno je proizvoljno vrijeme odrade sljedeće lekcije i dva pokušaja. Iste postavke su vrijedile i za lekciju Sortiranje 2 (A3). U otvorenom Forumu za diskusiju o eventualnim pitanjima o zadacima Zadaće (A2) nije nitko sudjelovao. Završni test je zbog nemogućnosti zajedničkog pisanja proveden u dva dana.

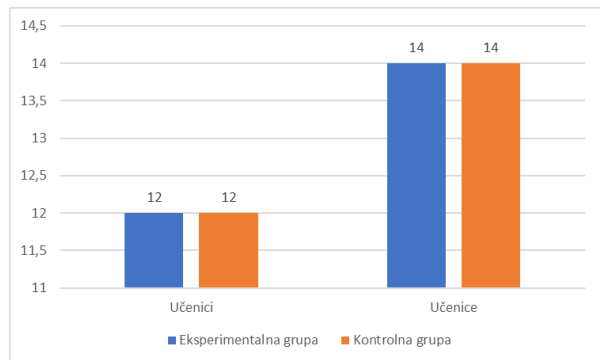
Uz nastavne aktivnosti provedene su dvije ankete tijekom cijelog procesa. Ankete provedene u ovom istraživanju izrađene su pomoću Google obrasca. Učenici su anketu ispunjavali dobrovoljno ali ne anonimno. U anketama je učenicima zajamčena povjerljivost i zaštita podataka, što je način zaštite sudionikova prava na privatnost (Cohen et al., 2013). Svaka od grupa je ispunjavala dvije ankete, jednu nakon odrade aktivnosti A1, a drugu nakon odrade A3. Obje grupe su odgovarale na jednaka pitanja, uz dodatna pitanja za Eksperimentalnu grupu, a koja se odnose na korištene vizualizacijske alate. Pitanja se mogu podijeliti u nekoliko skupina: pitanja koja se odnose na princip rada *selection sort-a*, njegovu implementaciju u Pythonu, koncept rada ostalih algoritama, stavovi o programiranju općenito, stavovi o provođenju online nastave, te jedno dodatno pitanje otvorenog tipa za Eksperimentalnu grupu. Od pitanja iz navedenih skupina, ankete su sadržavale 11 zajedničkih pitanja, te 17 međusobno različitih pitanja.

Kako ne bi ugrozili uspjeh učenika, u svrhu etičnosti ispitivanja (Cohen et al., 2013) učenici nisu bili ocjenjivani za navedene aktivnosti i testove.

3.2 Sudionici

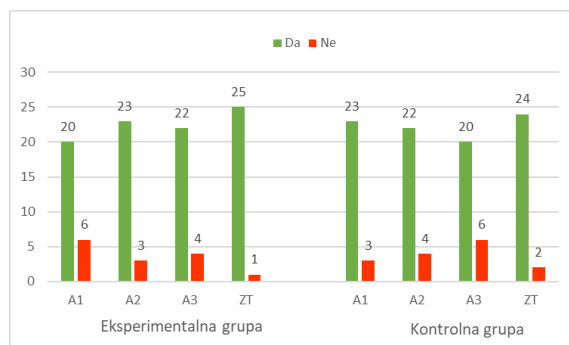
Sudionici ovog istraživanja su učenici trećih razreda jedne Prirodoslovno-matematičke gimnazije u Splitu koji su sudjelovali u online nastavi Informatike. Budući da se istraživala ciljana skupina populacije sudionici čine prigodni, neslučajni uzorak.

Ukupan broj sudionika je 52, od čega 24 čine učenici, a 28 učenice. Obje grupe imaju po 26 učenika. Na slici 1 prikazana je raspodjela učenika.



Slika 1. Raspodjela učenika po grupama

Na slici 1 su prikazani svi učenici koji pohađaju razrede u kojima se provelo istraživanje. Nisu svi učenici odradili sve aktivnosti pa su pri analizi podataka uzeti u obzir samo podaci učenika koji su odradili određene aktivnosti. Slika 2 prikazuje odradenost aktivnosti po grupama.



Slika 2. Odradenost aktivnosti po grupama

4 Rezultati istraživanja

4.1 Analiza aktivnosti A1

Aktivnost A1, u kojoj je objašnjen koncept *selection sorta*, dovršilo je 20 učenika u Eksperimentalnoj i 23 učenika u Kontrolnoj grupi (slika 2). Za utvrđivanje

razlika u rezultatima ove aktivnosti (lekcije) između grupa korišten je Mann-Whitney U test. Neparametrijski Mann-Whitney U test nije pokazao statistički značajnu razliku među grupama za rezultate lekcije ($U=185.500$, $p=0.345$). Budući da su grupe A (EA, KA) obiju proučavanih skupina odrađivale A1 u ponedjeljak, a grupe B (EB, KB) u srijedu, u pitanjima su brojevi u nizovima bili promijenjeni, kako bi se reduciralo eventualno prepisivanje grupe u srijedu. Zbog toga su uspoređeni i rezultati podgrupa koje su odrađivale aktivnost istoga dana, kao i rezultate unutar samih grupa E i K. U tablici 4 su prikazani rezultati Mann-Whitney U testa za utvrđivanje razlika među podgrupama za aktivnost A1.

	EA-KA	EB-KB	EA-EB	KA-KB
Mann-Whitney U	47.000	42.000	40.500	48.500
Z	-0.806	-0.670	-0.715	-0.776
p	0.420	0.503	0.475	0.438

Tablica 4 Rezultati Mann-Whitney U testa za utvrđivanje razlika među podgrupama za aktivnost A1

Prema rezultatima Mann-Whitney U testa nema statistički značajnih razlika među testiranim skupinama. Učenici svih grupa su ostvarili prosječno jako dobre rezultate, između 87% i 89.04% pri čemu nema statistički značajnih razlika (tablica 3). Zbog razlike u terminima pristupanja testovima, gdje su podgrupe EA i KA pristupale lekciji dan prije grupa EB i KB značajno je navesti i dodatne podatke koji su prikazani u tablici 5.

Podgrupa	Prosječan broj bodova	Prosječno vrijeme	Najduže vrijeme	Najbrže vrijeme
EA	87.00%	39 min 54 s	1 sat 14 min	14 min 6 s
KA	88.11%	29 min 34 s	51 min 36 s	8 min 19 s
EB	86.78%	37 min 11 s	1 sat 18 min	12 min 43 s
KB	89.04%	23 min 38 s	37 min 57 s	13 min 54 s

Tablica 5 Deskriptivna statistika Lekcije Sortiranje 1

Videolekcije za grupu E su trajale ukupno oko 3 minute duže nego kod grupe K (zbog objašnjavanja korištenja vizualizacijskih alata). Ako i izuzmemo te tri minute, primjećujemo dulje prosječno vrijeme rješavanja ove Lekcije za podgrupe EA i EB u odnosu na KA i KB.

4.2 Analiza aktivnosti A3

U aktivnosti A3, kroz lekciju je na istom nizu brojeva opisan rad algoritama *bubble sort*, *insertion sort*, *merge sort* i *quick sort*. Ova aktivnost se provodila isti dan pa nije bilo potrebe za formiranjem podgrupa kao u aktivnosti A1. Aktivnost 3 je dovršilo 20 učenika u grupi E, odnosno 22 u grupi K. Na samom početku, prije prelaska na ostale algoritme, postavljeno je pitanje višestrukog odabira odgovora koje se odnosilo na *selection sort*. U grupi K su svi učenici odabrali točan odgovor. U grupi E je troje učenika odabralo pogrešan odgovor. Rezultat Mann-Whitney U testa je pokazao da postoji statistički značajna razlika među testiranim skupinama u aktivnostima lekcije ($U=145.500$, $p=0.011$). Tablica 6 prikazuje deskriptivnu statistiku lekcije Sortiranje 2.

Grupa	Prosječan broj bodova	Prosječno vrijeme	Najduže vrijeme	Najbrže vrijeme
E	93.41%	22 min 7 s	48 min 3 s	3 min 23 s
K	98.75%	11 min 52 s	33 min 50 s	3 min 47 s

Tablica 6 Deskriptivna statistika Lekcije Sortiranje 2

Iz tablice je vidljivo da je grupa K imala bolji prosječan rezultat (98.75%) u odnosu na E skupinu (93.41%).

Videolekcije u obje grupe su trajale oko 13 minuta. Unatoč tome, opet su učenici E skupine proveli više vremena za obradu lekcije. Primjetna su i najbrža vremena u grupama, oko 3.5 minute, što ukazuje da pojedini učenici nisu detaljno obrađivali ovu aktivnost.

4.3 Analiza završnog testa (ZT)

Završni test u formi Aktivnog kviza odradilo je 25 učenika grupe E, odnosno 24 učenika grupe K (slika 2). Test se sastojao od ukupno šest zadataka. Rezultati Mann-Whitney U testa su pokazali statistički značajnu razliku između grupa ($U=192.500$, $p=0.031$) u ukupnim rezultatima, te za zadatke Z2 ($U=21.000$, $p=0.000$) i Z6 ($U=144.500$, $p=0.012$). Tablica 6 prikazuje deskriptivnu statistiku za završni test.

Grupa	N	AS	Mdn	Mod	Min	Max
E	25	14.82	14.50	14.50	7.00	19.00
K	24	12.25	13.00	16.00	1.00	18.00

Tablica 6 Deskriptivna statistika po skupinama za rezultate aktivnosti A4 (završni test)

Prema rezultatima, učenici grupe E su postigli ukupno bolje rezultate (AS=14.82, Mdn=14.5, Mod=14.5, Min=7, Max=19) u usporedbi s rezultatima učenika grupe K (AS=12.25, Mdn=13, Mod=16, Min=1, Max=18). Z2 pitanje je vezano za *bubble sort* na koje nitko u grupi K nije točno odgovorio, dok je u grupi E samo dvoje učenika pogrešno odgovorilo. Na to je vjerojatno utjecala činjenica da je grupa E pisala ZT dan nakon grupe K.

4.3.1 Analiza rezultata *selection sort* zadataka u (ZT*)

Budući da je u ovom istraživanju poseban naglasak na algoritmu *selection sort*, analizirali smo rezultate završnog testa (ZT) dobivene isključivanjem učenika koji nisu odradili aktivnost A1, u kojoj je detaljnije objašnjen ovaj algoritam. Također smo isključili prva dva zadatka iz ZT koji su se odnosili na ostale algoritme sortiranja i nisu bili promijenjeni u grupama. Takvih učenika je 19 u grupi E, odnosno 22 u grupi K. U ovako filtriranim podacima rezultat Mann-Whitney U testa nije pokazao statistički značajnu razliku između grupa (U=177.500, p=0.408). Tablica 7 pokazuje deskriptivnu statistiku rezultata.

Grupa	N	AS	Mdn	Mod	Min	Max
E	19	10.18	10.0	9.50	6.00	14.00
K	22	9.27	9.25	12.0	1.00	14.00

Tablica 7 Deskriptivna statistika po skupinama za rezultate ZT*

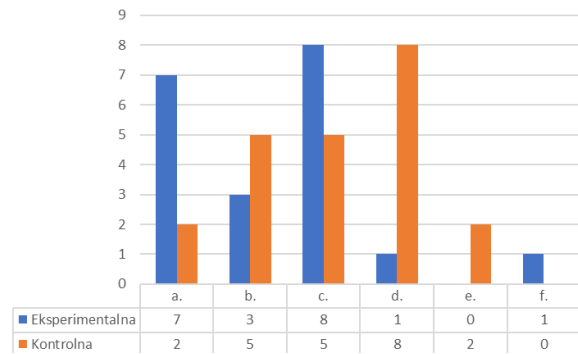
Dodatnom analizom suse uspoređivali rezultati za svako pitanje pri čemu se pokazala statistički značajna razlika za zadatak 6 (U=89.500, p=0.007).

U ovom zadatku je trebalo napisati program za sortiranje niza silazno. Kod učeničkih rješenja ovog zadatka, primjećuju se sljedeće grupe ponuđenih rješenja:

- Potpuno točan odgovor;
- Nije riješeno ili pogrešno riješeno;
- Uzlazno sortiran niz;
- Prepisan algoritam za sortiranje;
- Pogrešno prepisan algoritam za sortiranje;
- Korištenje gotove metode `sort()`.

Među navedenim grupama rješenja skupine c i e nisu pogrešna rješenja već nedovršena (c) vjerojatno zbog nepažnje u čitanju zadataka ili točno ali korištenjem funkcije `sort()` (f) što nije eksplicitno navedeno u zadatku. Odgovori skupine b su netočni kao i skupina d i e jer je prepisan algoritam za sortiranje.

Slika 3 prikazuje distribuciju odgovora za zadatak 6.



Slika 3. Distribucija odgovora za zadatak 6

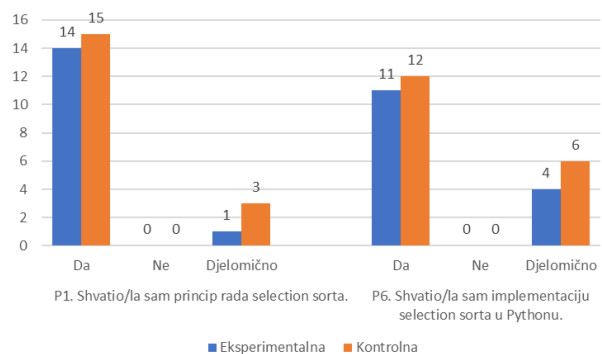
Iz analize odgovora je vidljivo da su učenici iz E skupine (n=16) češće ponudili točno (a i f) ili nepotpuno rješenje (c), dok su učenici iz K skupine (n=15) češće ponudili pogrešna rješenja (b, d i e).

4.4 Analiza stavova učenika

Stavovi učenika su ispitani anketnim pitanjima. Prvu anketu je popunilo 15 učenika E i 18 učenika K grupe. Drugu anketu je popunilo 17 učenika E i 19 učenika K grupe. Pitanja su svrstana u tri skupine koje će biti analizirane u sljedećim poglavljima.

4.4.1 Stavovi o usvojenosti *selection sort* algoritma

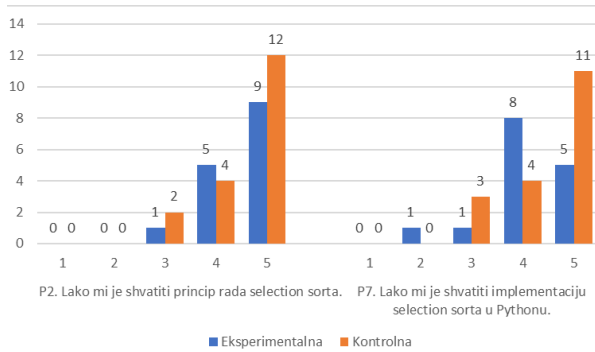
U pitanjima iz skupine koja se odnose na razumijevanje samog koncepta i implementacije *selection sort*a nisu utvrđene statistički značajne razlike među grupama. Slika 4 prikazuje raspodjelu odgovora na dva pitanja vezana uz princip rada *selection sort*-a.



Slika 4. Distribucija odgovora za pitanja vezana uz shvaćanje rada algoritma sortiranja

Može se uočiti da učenici iz E skupine u nešto većoj mjeri smatraju da su shvatili princip rada *selection sort*-a i njegove primjene u Pythonu.

Dva pitanja su bila vezana uz procjenu težine shvaćanja *selection sort*-a. Slika 5 prikazuje distribuciju odgovora na takva pitanja.

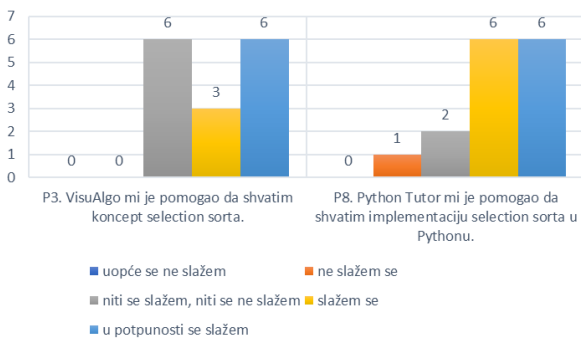


Slika 5. Distribucija odgovora vezana uz težinu shvaćanja algoritma sortiranja

Može se uočiti da, slično prethodnim odgovorima, učenici iz E skupine smatraju lakšim shvaćanje principa rada samog algoritma i njegove primjene u Pythonu.

Grupa E je imala tri dodatna pitanja vezana uz korištene alate za vizualizaciju. Ukupno 8 učenika je tvrdilo da je samostalno koristilo *VisuAlgo*, a 9 *PythonTutor* pri učenju *selection sorta*. Slika 6 prikazuje distribuciju odgovora na pitanja vezana uz korištene alate.

Učenici su nešto pozitivnije ocijenili Python Tutor, vjerojatno zato što im pomaže u razumijevanju implementacije samog algoritma u programskom jeziku koji im je poznat, za razliku od *VisuAlgo* koji bi trebao pomoći u razumijevanju principa rada samog algoritma za što je potrebna veća razina apstrakcije.



Slika 6. Distribucija odgovora vezana uz alate za vizualizaciju

Grupa E je imala još jedno dodatno pitanje otvorenog tipa „...što mislite može li korištenje nekih programa kao što su *VisuAlgo* i *PythonTutor* biti učinkovito u nastavi?“, od čega je 15 komentara bilo pozitivno, te po jedan negativan i jedan suzdržan. Neki od komentara su sljedeći:

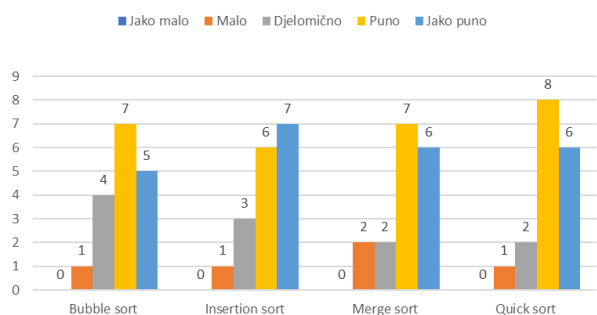
- „Mogu biti korisni. Bolje predočavaju problematiku zadataka.“
- „Mislim da može, jer pojednostavni dosta toga i pokazuje nam korak po korak što radi svaki dio koda.“
- „Jako, ali sve ovisi o volji učenika.“

- „Mislim da može, pogotovo *PythonTutor*, jer se gradivo lakše shvati jer objašnjava red po red.“
- „Mislim da može. Omogućit će lakše i zanimljivije učenje.“
- „Mislim da to može biti vrlo učinkovito jer ga i sama znam koristiti tijekom nastave uz dopuštenje nastavnice. Lakše je percipirati neke imaginarne dijelove.“

4.4.2 Stavovi o usvojenosti ostalih algoritama sortiranja

Jedna skupina pitanja se odnosila na ostale algoritme sortiranja. Kod ovih algoritama je opisan samo princip rada, bez implementacije u programskom jeziku. Ni u ovoj skupini pitanja nisu utvrđene statistički značajne razlike u odgovorima grupa. I ovdje većina učenika smatra da su shvatili način na koji rade algoritmi. Manji dio učenika je djelomično shvatio ostale algoritme, dok po jedan učenik iz svake grupe nije shvatio princip rada složenijih algoritama *merge sort* i *quick sort*.

E grupa je imala dodatno pitanje koje se odnosi na to koliko im je *VisuAlgo* pomogao u razumijevanju rada pojedinog algoritma. Distribucije odgovora za sve algoritme su bile podjednake, neovisno o činjenici je li riječ o jednostavnijem ili složenijem algoritmu, većina učenika smatra da im je *VisuAlgo* puno ili jako puno pomogao u razumijevanju svakog pojedinog algoritma. Slika 7 prikazuje distribuciju odgovora učenika za navedena pitanja.

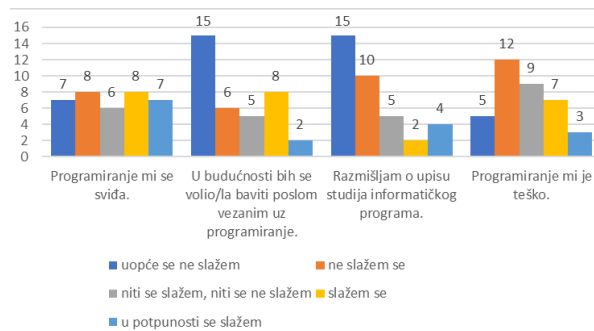


Slika 7. Distribucija odgovora vezana uz alate za vizualizaciju

4.4.3 Stavovi o programiranju

Jedna grupa pitanja se odnosila stavove o programiranju (Likertova skala). Jednak broj učenika ima pozitivan stav prema programiranju kao i negativan. Zanimljivo je bilo vidjeti da se puno više učenika u budućnosti ne vidi u poslu vezanom za programiranje, niti razmišljaju o upisu studija s informatičkim programom što je iznenađujuće jer se radi o učenicima prirodoslovno-matematičke gimnazije. Slika 8 prikazuje distribuciju učeničkih

odgovora na pitanja o stavovima prema programiranju.

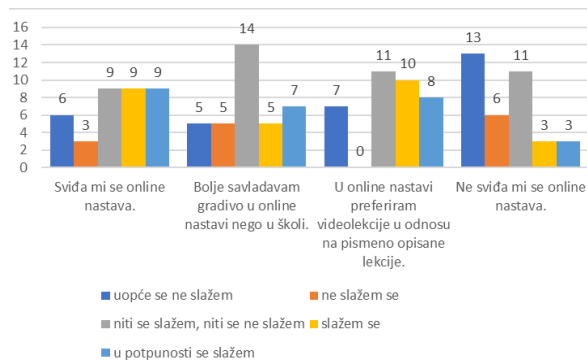


Slika 8. Distribucija odgovora vezana uz stavove o programiranju

4.4.4 Stavovi o online nastavi

Posljednja skupina pitanja ispituje učeničke stavove o aktualnom provođenju online nastave (Likertova skala 1-5). Iz učeničkih odgovora na tvrdnje se ne primjećuje izrazito pozitivan ili negativan stav prema online nastavi. Polovica učenika ima pozitivan stav prema online nastavi, oko četvrtina njih ima negativan stav, dok su ostali suzdržani.

Slika 9 prikazuje distribuciju odgovora na pitanja vezana uz stavove o online nastavi.



Slika 9. Distribucija odgovora vezana uz stavove o online nastavi

U pitanju otvorenog tipa „Napišite kratko što vam se sviđa/ne sviđa u provođenju online nastave.” većina učenika kao pozitivnu stranu online nastave navodi fleksibilnost, tj. činjenicu da sami mogu organizirati svoje vrijeme. Troje učenika je navelo videopozive i videolekcije kao pozitivno u ovakvom tipu nastave. Negativni stavovi prema online nastavi su nešto raznolikiji i najčešće se odnose na: preveliku količinu gradiva, previše zadataka, negativne strane Loomena (usporenost, problem s brzinom), termine postavljanja zadataka na Loomen (tijekom cijelog dana), nedostatak komunikacije s profesorom.

5 Ograničenja istraživanja

Postoji nekoliko ograničenja istraživanja koja treba uzeti u obzir. Jedno od ograničenja je činjenica da grupe nisu pisale završni test (ZT) u isto vrijeme već s jednim danom odmaka čime postoji mogućnost prijenosa informacija o zadacima testa između grupa. Dodatno ograničenje je svakako online nastava koja se u vrijeme provođenja testa provodila u potpunosti. Ostaje pitanje bi li rezultati bili isti da se nastava provodila frontalno.

Osim navedenog, ograničenja su broj sudionika i duljina trajanja eksperimenta. Poželjno je obuhvatiti veću populaciju učenika i nastavnika kao i duže vrijeme trajanja eksperimenta kako bi rezultati bili mjerodavniji.

Jedno od ograničenja se odnosi na ujednačenost skupina. Zbog oblika nastave nije bilo moguće provesti predtest kojim bi se utvrdila ujednačenost skupina po predznanju i sposobnostima. Kako bi na neki način mogli utvrditi ujednačenost skupina nastavnica koja predaje razrediima treću godinu je kategorizirala svakog učenika prema dojamu programerskih sposobnosti u jednu od težinskih skupina: *lošiji*, *srednji*, *bolji*. Za utvrđivanje ujednačenosti skupina proveden je Kruskal-Wallis test koji je pokazao da nema statistički značajnih razlika između eksperimentalne i kontrolne skupine za aktivnost A1 ($H(2)=1.814$, $p=0.404$), ni za ZT* ($H(2)=2.771$, $p=0.250$).

Sva navedena ograničenja su rezultat ograničenja istraživanja u edukaciji gdje se trebaju omogućiti što prirodniji uvjeti sudionicima kako se samom intervencijom ne bi djelovalo na rezultate istraživanja.

6 Rasprava

Rezultati ovog istraživanja nisu utvrdili prednosti grupe za čije poučavanje su korišteni vizualizacijski alati. Iako su neka istraživanja pokazala ima značajnu prednost u korištenju vizualizacije, prvenstveno kad su konstruktivistički utemeljene (Grissom, McNally, & Naps, 2003), rezultati ovog istraživanja su u skladu s istraživanjima koja nisu pokazala da korištenje vizualizacije pospješuje učenje (Jarc et al., 2000; Mladenović et al., 2020). Završni test (ZT) je u konačnici bila jedina aktivnost u kojoj je Eksperimentalna grupa postigla statistički značajno bolje rezultate u odnosu na Kontrolnu grupu. Zbog toga smo vjerodostojniju analizu (ZT*) dobili isključivanjem u analizi učenika koji nisu odradili aktivnost A1 u kojoj je bio objašnjen princip rada i implementacija *selection sorta*, te isključivanjem prva dva zadatka ZT-a koji su se odnosili na ostale algoritme sortiranja. Time u konačnici nije utvrđena

statistički značajna razlika među grupama za završni test (ZT*). Spomenuta aktivnost A1 je bila u obliku lekcije na Loomenu, te također nije pokazala statistički značajnu razliku među grupama. Ova aktivnost se pokazala možda kao i vjerodostojnija i reprezentativnija nego sam završni test, jer je poticala aktivnost učenika bez nekakvog posebnog „tereta“ testa. Pri analizi učeničkih odgovora na anketna pitanja o usvajanju algoritama sortiranja, nisu utvrđene razlike među grupama. Učenici smatraju da su usvojili proučavane koncepte. Implementacija algoritma zahtjeva veće mentalne napore (Bernát, 2014). Upravo zbog toga je teško usvajanje algoritama sortiranja ispitati u ovakvom tipu nastave, gdje ne možemo utvrditi vjerodostojnost učeničkih rješenja pri predaji za dataka i samostalnost u rješavanju. Osim toga, ipak nedostaje frontalna nastava gdje nastavnik može napraviti dodatne intervencije prema reakcijama učenika, te odgovarati na njihova pitanja koja ipak u online nastavi uglavnom izostaju.

Prema navedenim rezultatima odbacujemo postavljenu hipotezu H1 i ne možemo potvrditi da učenici koje se poučava primjenom vizualizacije ostvaruju bolje rezultate u aktivnostima online nastave u kojima se ispituje usvojenost *selection sort-a* u odnosu na učenike koji se poučavaju na klasičan način, bez vizualizacije.

Kad je u pitanju istraživačko pitanje o stavu učenika prema online nastavi možemo reći da učenici nemaju jasno izražen stav. Online nastava je novitet učenicima i vidjeli smo da učenici nemaju izrazito negativan niti pozitivan stav prema online nastavi. Kao što je rečeno, ovo su učenici s dobrim sposobnostima rješavanja problema i već razvijenom samostalnosti u učenju, pa u ovakvom tipu nastave cijene činjenicu da sami mogu organizirati vrijeme.

7 Zaključak

Budući da je online nastava novitet u obrazovanju Republike Hrvatske, učenici i nastavnici su bili suočeni s naglim promjenama u načinu odrađivanja svojih obaveza bez vremena za pripremu. U poučavanju apstraktnih tema kao što su algoritmi sortiranja, nastavnici imaju na raspolaganju više vizualizacijskih alata koje mogu integrirati u svoju nastavu zajedno s drugim nastavnim materijalima. Premda zahtjeva dodatni trud nastavnika, ova prednost online nastave se može iskoristiti kako bi se potaknula učenička aktivnost i samostalnost, što je i ključ kvalitetnog i trajnog učenja. Rezultati provedenog kvaziekperimenta tijekom online nastave pokazali su različite rezultate. Iako je u jednoj aktivnosti (A3) kontrolna skupina postigla bolje rezultate, a u završnom testu eksperimentalna,

ukupni rezultati nakon ujednačavanja za dataka nisu pokazali statistički značajne razlike između skupina. Ovakvi rezultati su u skladu s brojnim provedenim istraživanjima u kojima se nije pokazala značajna prednost primjene vizualizacija. Učenici Eksperimentalne grupe su u anketnim pitanjima potvrdili ulogu vizualizacije kao pomoći u poučavanju algoritama sortiranja. Većina učenika iz obje grupe je kao bitnu stavku online nastave naglasila fleksibilnost, što potvrđuju svojim sudjelovanjem u većini provedenih aktivnosti online nastave za vrijeme ovog istraživanja. Premda nisu utvrđene razlike među grupama, ne treba odustati od korištenja vizualizacije, nego treba poticati njeno aktivno korištenje kako bi postala izvor za konstrukciju učeničkog znanja.

Literatura

- Bernát, P. (2014). The Methods and Goals of Teaching Sorting Algorithms in Public Education. *Acta Didactica Napocensia*, 7(2), 1-9.
- Boljat, I., & Pletikosa-Grgić, I. (2013). Pedagoški potencijal algoritamske vizualizacije u nastavi informatike. *Methodological Horizons*, 8(18).
- Cohen, L., Manion, L., & Morrison, K. (2013). *Research methods in education*. Routledge.
- Cunningham, K., Blanchard, S., Ericson, B., & Guzdial, M. (2017). Using Tracing and Sketching to Solve Programming Problems. <https://doi.org/10.1145/3105726.3106190>
- Dann, W., & Cooper, S. (2009). Alice 3: Concrete to Abstract. *Communications of the ACM*, 52(8), 27. <https://doi.org/10.1145/1536616.1536628>
- Gómez-Henríquez, L. M. (2001). Software visualization: An overview.
- Grissom, S., McNally, M. F., & Naps, T. (2003). Algorithm Visualization in CS Education: Comparing Levels of Student Engagement. In *Proceedings of ACM Symposium on Software Visualization*.
- Guo, P. J. (2013). Online python tutor: Embeddable web-based program visualization for cs education. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education* (pp. 579–584). <https://doi.org/10.1145/2445196.2445368>
- Hundhausen, C. D., Douglas, S. A., & Stasko, J. T. (2002). A meta-study of algorithm visualization effectiveness. *Journal of Visual Languages and Computing*, 13(3), 259–290. <https://doi.org/10.1006/jvlc.2002.0237>
- Jarc, D. J., Feldman, M. B., & Heller, R. S. (2000). Assessing the benefits of interactive prediction using Web-based algorithm animation

- courseware. *SIGCSE Bulletin (Association for Computing Machinery, Special Interest Group on Computer Science Education)*. <https://doi.org/10.1145/331795.331889>
- Koppelman, H., & Van Dijk, B. (2010). Teaching abstraction in introductory courses. In *ITiCSE'10 - Proceedings of the 2010 ACM SIGCSE Annual Conference on Innovation and Technology in Computer Science Education*. <https://doi.org/10.1145/1822090.1822140>
- Korhonen, A., & others. (2003). *Visual algorithm simulation*. Helsinki University of Technology.
- Ministarstvo znanosti i obrazovanja Republike Hrvatske. (2020). *Akcijski plan za provedbu nastave na daljinu*.
- Ministarstvo znanosti i obrazovanja. (2018). *Kurikulum nastavnoga predmeta informatika za osnovne i srednje škole*. Retrieved from https://mzo.hr/sites/default/files/dokumenti/2018/OBRAZOVANJE/Nacionalni-kurikulumi/informatika/kurikulum_nastavnoga_predmeta_informatika.pdf
- Mladenović, M., Žanko, Ž., & Aglič Čević, M. (2020). The impact of using program visualization techniques on learning basic programming concepts at the K–12 level. *Computer Applications in Engineering Education*. <https://doi.org/10.1002/cae.22315>
- Myller, N., Bednarik, R., Sutinen, E., & Ben-Ari, M. (2009). Extending the Engagement Taxonomy: Software Visualization and Collaborative Learning. *Trans. Comput. Educ.*, 9(1), 7:1–7:27. <https://doi.org/10.1145/1513593.1513600>
- Naps, T., Cooper, S., Koldehofe, B., Leska, C., Rößling, G., Dann, W., ... McNally, M. (2003). Evaluating the educational impact of visualization. *ACM SIGCSE Bulletin*. <https://doi.org/10.1145/960492.960540>
- Naps, T. L., Rößling, G., Almstrum, V., Dann, W., Fleischer, R., Hundhausen, C., ... Velázquez-Iturbide, J. Á. (2002). Exploring the Role of Visualization and Engagement in Computer Science Education. *SIGCSE Bull.*, 35(2), 131–152. <https://doi.org/10.1145/782941.782998>
- Price, B. A., Baecker, R. M., & Small, I. S. (1993). A principled taxonomy of software visualization. *Journal of Visual Languages & Computing*, 4(3), 211–266.
- Sorva, J. (2013). Notional machines and introductory programming education. *ACM Transactions on Computing Education*, 13(2), 1–31. <https://doi.org/10.1145/2483710.2483713>
- Statter, D., & Armoni, M. (2016). Teaching abstract thinking in introduction to computer science for 7 th graders. In *ACM International Conference Proceeding Series*. <https://doi.org/10.1145/2978249.2978261>
- Statter, D., & Armoni, M. (2017). Learning abstraction in computer science: A gender perspective. In *ACM International Conference Proceeding Series*. <https://doi.org/10.1145/3137065.3137081>

The role of the use of visualization tools for teaching sorting algorithms in online teaching

Abstract

Learning and teaching of sorting algorithms are very challenging for students and teachers as well. The issues are more highlighted in the online form of teaching which was the only form of schooling from March to June of the school year 2019/2020 in Croatia due to the COVID-19 pandemic. The use of visualization tools could be used for lowering the abstraction of complex programming concepts such as sorting algorithms. We conducted research among two high-school classes (n=52) in one science and mathematics high school while teaching sorting algorithms during online schooling. In the experimental group, we used visualization tools VisuAlgo and Python Tutor for teaching sorting algorithms and their implementation in Python as well. We examined students' attitudes towards programming and online teaching as well. We present the results of the research in this paper.

Keywords: visualization; online teaching; sorting algorithms; selection sort; informatics.