

SVET ALGORITMOV PRI PREDMETU ŠPORT V I. VZGOJNO-IZOBRAŽEVALNEM OBDOBJU OSNOVNE ŠOLE

¹Igor Časar

¹Dvojezična osnovna šola Genterovci, Slovenija

Povzetek

Ključne besede:

algoritmi, spodbudna učna
okolja, I. vzgojno-
izobraževalno obdobje

Keywords: algorithms,
stimulating learning
environments, the first
educational cycle

Spodbudno učno okolje, v katerem imajo abstrakcija, dekompozicija, generalizacija in prepoznavanje vzorcev osrednjo vlogo, učencem omogoča razvoj temeljnih znanj računalništva in informatike. V prispevku je predstavljena študija primera inovativne učne prakse, ki pri učencih prvega vzgojno-izobraževalnega obdobja razvija koncept algoritmov. Raziskava preučuje, kako lahko s piktogrami pri predmetu šport razvijamo temeljna računalniška znanja. Cilj raziskave je analiza učnih pristopov, metod ter učinkov integracije računalniških orodij v razredni pouk, pri čemer upoštevamo Okvir računalništva in informatike od vrta do srednje šole ter učni načrt za predmet šport. Podatke smo pridobili s kvalitativno metodo. Študija primera je bila izvedena na namenskem, majhnem vzorcu homogene skupine učencev DOŠ Genterovci, starih od 7 do 9 let. Podatki so bili zbrani z opazovanjem, analizo izdelkov in polstrukturiranim intervjujem. Analiza podatkov je bila opravljena s tematsko analizo. Analiza odgovorov je pokazala, da je znotraj spodbudnega učnega okolja mogoče razvijati temeljna znanja računalništva in informatike brez uporabe digitalnih naprav. Učenci lažje opišejo problem, prepoznajo ključne podrobnosti in poiščejo rešitev. Učenci niso imeli težav s prepoznavanjem in ustvarjanjem algoritmov. Z oblikovanjem ekip in razdeljenimi vlogami se v večji meri razvija sodelovalno učenje, ki spodbuja deljenje izkušenj in skupno reševanje izzivov. Ugotovitve kažejo, da je mogoče podobne metode uporabiti tudi pri drugih predmetih, kar odpira nove možnosti za integracijo računalniškega mišljenja v osnovnošolsko izobraževanje.

THE WORLD OF ALGORITHMS IN PHYSICAL EDUCATION IN THE FIRST EDUCATIONAL CYCLE OF PRIMARY SCHOOL

Abstract

A stimulating learning environment, where abstraction, decomposition, generalization, and pattern recognition play a central role, enables pupils to develop fundamental computer science and informatics

Copyright: © 2025

Avtorji/The author(s). To
delo je objavljeno pod
licenco Creative Commons
CC BY Priznanje avtorstva
4.0 Mednarodna.

Uporabnikom je dovoljeno
tako nekomercialno kot
tudi komercialno
reproduciranje,
distribuiranje, dajanje v
najem, javna priobčitev in
predelava avtorskega dela,
pod pogojem, da navedejo
avtorja izvirnega dela.

<https://creativecommons.org/licenses/by/4.0/>



knowledge. This paper presents a case study of an innovative teaching practice that develops the concept of algorithms in pupils in the first educational cycle. The research examines how integrating pictograms into physical education can contribute to the development of fundamental computer science knowledge in pupils in the first educational cycle. The research aims to analyze teaching approaches, methods, and the effects of integrating computational tools into classroom instruction, considering the Framework for Computer Science from Preschool to Secondary School and the Physical Education curriculum. Data was collected using qualitative methods. The case study was conducted on a targeted small sample of a homogeneous group of pupils from the bilingual primary school Genterovci, aged 7 to 9. The data was collected through observation, product analysis, and semi-structured interviews. Data analysis was conducted using thematic analysis. The analysis of the responses showed that, within a stimulating learning environment, it is possible to develop fundamental computer science knowledge without using digital devices. Pupils were able to describe the problem more easily, identify key details, and find solutions. Pupils had no difficulty recognizing and creating algorithms. Collaborative learning was further developed by forming teams and assigning roles, encouraging experience-sharing and joint problem-solving. The findings suggest that similar methods could be applied to other subjects, opening new possibilities for integrating computational thinking into primary school education.

1 UVOD

Temeljna znanja računalništva in informatike (v nadaljevanju RIN) so za posameznika – ne glede na to, ali gre za učenca, dijaka ali odraslo osebo – nujna za kakovostno vključevanje v družbo 21. stoletja (RINOS, 2022). Pri tem je smiselno izpostaviti ključno področje, računalniško razmišljanje, ki ga Cuny (2010) opredeljuje kot miselni proces, v katerem posameznik oblikuje probleme in njihove rešitve. Te rešitve so predstavljene na način, ki omogoča njihovo razumevanje sistemu za obdelavo informacij. Eden ključnih izzivov izobraževalnega sistema je učencem omogočiti razvoj ustreznih znanj, kompetenc in veščin, hkrati pa se izogniti pastem sodobne tehnologije, kot je prekomerna uporaba digitalnih naprav. Ker lahko pretirana raba digitalnih naprav vpliva na učne in socialne navade učencev, je pomembno razmisliti o pristopih k poučevanju RIN, ki ne temeljijo zgolj na uporabi tehnologije.

Temeljna znanja RIN je mogoče razvijati na različne načine, pri čemer uporaba digitalnih naprav ni nujno potrebna. Chen Peng (2023) izpostavlja dva osrednja pristopa: aktivnosti, ki temeljijo na uporabi digitalnih naprav, ter aktivnosti, ki se izvajajo brez njih. Looi s sodelavci (2018) predlagajo strategijo postopnega uvajanja temeljnih znanj RIN, pri čemer v začetni fazi izvajamo aktivnosti brez uporabe naprav, nato pa postopoma vključujemo digitalna orodja. Več raziskav (Busuttil in Formosa, 2020; Chen in Chi, 2020; Csizmadia idr., 2019; Minamide idr., 2020; Saxena et al., 2020; Threekunprapa in Yasri, 2020) poudarja, da so igre s kartami, namizne igre, nalepke in gibanje med najpogostejšimi pristopi za razvoj temeljnih znanj RIN brez uporabe digitalnih naprav.

Naša raziskava preučuje vpliv vpeljevanja koncepta algoritmov pri predmetu šport na razvoj temeljnih računalniških znanj pri učencih prvega vzgojno-izobraževalnega obdobja. Naše raziskovalno vprašanje je: S katerimi dejavnostmi, brez uporabe zaslona, lahko vpeljemo koncept algoritmov v pouk športa in prispevamo k razvoju temeljnih znanj računalniškega razmišljanja pri učencih prvega vzgojno-izobraževalnega obdobja?

2 ALGORITMI IN NARAVNE OBLIKE GIBANJA

Računalniški algoritmi imajo razmeroma malo sestavin, saj digitalne naprave lahko sledijo le nekaj vrstam navodil. Med ključne funkcije algoritmov sodijo: sprejemanje vhodnih podatkov, zagotavljanje izhodnih podatkov, shranjevanje vrednosti, sledenje navodilom v zaporedju, izbira med možnostmi in ponavljanje navodil v zanki. Kljub temu, da so omejeni na te osnovne elemente, omogočajo izvajanje tako preprostih kot tudi zahtevnejših nalog (CS Unplugged, 2010).

Algoritmi so osrednji koncept računalniškega razmišljanja. Opredeljeni so kot postopki, ki omogočajo reševanje problemov ali izvajanje nalog. Njihova učinkovitost je odvisna od natančnega sledenja zaporedju korakov, kar omogoča hitro in pravilno reševanje tudi ob spreminjanju vhodnih podatkov (CS Unplugged, 2010). Eden najbolj prepoznavnih primerov uporabe algoritmov je Rubikova kocka. Špur (2015) izpostavlja praktično uporabo algoritmov, saj poudarja, da Fridrichova metoda temelji na vnaprej določenih algoritmih, ki omogočajo hitro reševanje Rubikove kocke.

Izvajanje naravnih oblik gibanja spodbuja celosten razvoj otrok. Učitelji športne vzgoje lahko z uporabo teh aktivnosti ne le izboljšajo telesno pripravljenost učencev, temveč tudi prispevajo k njihovemu kognitivnemu in socialnemu razvoju (Kovač, Bergoč, idr., 2015).

2.1 VZORČNI UČNI SCENARIJ

Upoštevajoč teoretična izhodišča in raziskovalno vprašanje je pri učencih prvega vzgojno-izobraževalnega obdobja (v nadaljevanju I. VIO) smiselno koncept algoritmov uvajati s pomočjo piktogramov, ki ne vključujejo uporabe digitalnih zaslonov. Tako smo oblikovali vzorčni učni scenarij, katerega cilj je razvijanje algoritmičnega razmišljanja pri učencih brez uporabe digitalnih naprav. Aktivnosti so učence usmerjale k:

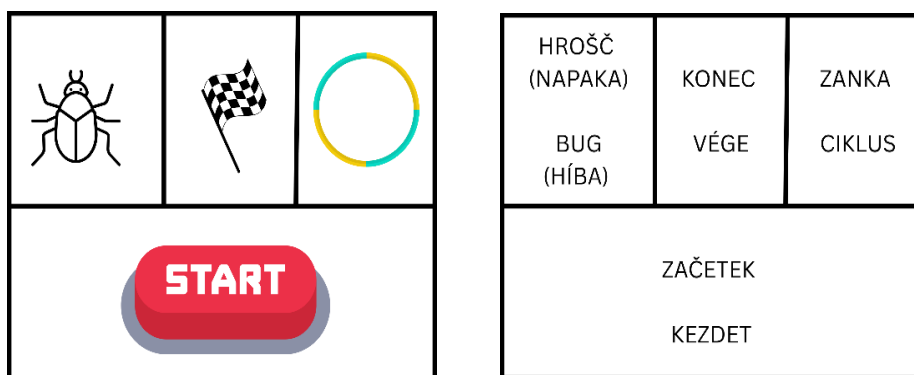
- opisu problema,
- identifikaciji ključnih vidikov problema,
- razgradnji kompleksnega problema na manjše, logične korake,
- uporabi algoritmov (zaporedja korakov) pri reševanju problema,
- evalvaciji samega procesa.

1. faza: Uvod v temo

Učna ura se je začela s postopnim uvajanjem v koncept algoritmov in aktivacijo predznanja učencev z metodo iskanje parov, kjer so učenci obračali kartice s skritimi simboli. Po odkritju para so ga razvrstili v eno izmed dveh kategorij: znani in neznani pojmi. Učitelj je vodil razpravo o neznanih pojmi ter poskrbel, da so učenci razumeli vse ključne koncepte za nadaljnje delo.

2. faza: Oblikovanje algoritma

Učenci so nato razvrščali pojme v dve skupini in s tem oblikovali vstopne podatke: (1) pojmi, ki se uporabljajo pri ustvarjanju algoritma, in (2) pojmi, povezani s športnimi aktivnostmi – naravne oblike gibanja.



Slika 1: Piktogrami za ustvarjanje algoritmov (brošč, konec, zanka in začetek)



Slika 2: Piktogrami za prikaz naravnih oblik gibanja

Skupaj z učiteljem so nato oblikovali kriterije uspešnosti, ki jim bodo pomagali oceniti lastno delo in delo sošolcev.

Tabela 1: Oblikovanje kriterijev uspešnosti

Odgovori učencev	Oblikovan kriterij uspešnosti
Da vem, kaj počnem.	Razumljiva aplikacija
Da bodo navodila dobra.	
Da bomo čim manjkrat ponavljali.	Pravilna izvedba
Da bomo čim prej končali.	
Da bomo imeli čim manj vprašanj.	
Da bomo začeli z go in končali s finish.	
Ko se ne bomo kregali.	Sodelovanje
Ko si bomo pomagali.	

Po določitvi kriterijev uspešnosti so učenci delovali v skupinah t. i. ekip razvijalcev. Učitelj, v vlogi naročnika, je podal zahteve za oblikovanje aplikacije:

1. Uporaba algoritma (vsak uporabnik izvede isto zaporedje).
2. Aplikacija ima jasno določen začetek in konec.
3. Aplikacija omogoča izvedbo vsaj treh dejavnosti.
4. Potrebno je preverjanje delovanja aplikacije.



Slika 3: Prikaz osnovnih zahtev aplikacije

Kot prikazuje slika 1, vidimo grafični prikaz zahtev naročnika. Začetna kartica predstavlja zagon programa in zadnja kartica zaključek procesa. Kartice si sledijo od leve proti desni, vmes imamo tri bele kartice, ki označujejo poljubno aktivnost naravnih oblik gibanja.

Vsaka ekipa je razvila svoj algoritem s fizičnimi piktogrami, ki so predstavljali zaporedje aktivnosti. Pri preizkušanju aplikacije je preizkuševalec dal ukaz "go", nato pa je spremljal pravilnost izvedbe. Če so odkrili napako, so uporabili posebno "kartico hroščka", s katero so označili problematično mesto v algoritmu.



Slika 4: Prikaz uporabe kartice hroščka ob zaznani napaki

Vsak ustvarjen algoritem so preizkusili vsaj 3 učenci. Tako smo zagotovili, da je algoritem ponovljiv s strani različnih deležnikov. Po uspešno opravljenem preizkusu in evalvaciji je sledila naslednja faza.

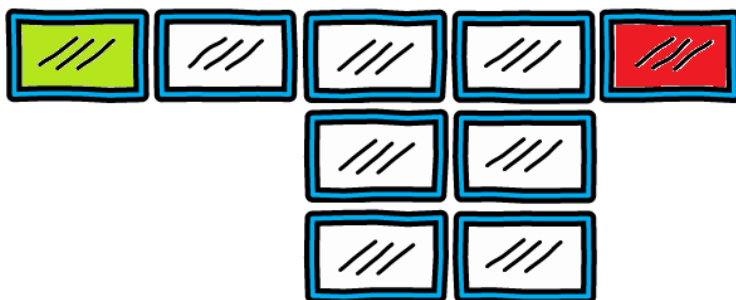
3. faza: Povečanje kompleksnosti

Po prvem preizkusu aplikacije so učenci dobili dodatno zahtevo: vaja 2 in 3 se ponovi trikrat. To je zahtevalo razširitev algoritma. Učenci so preizkusili dve različni strategiji:

1. Dodajanje korakov po horizontalni osi (več zaporednih ukazov).
2. Dodajanje korakov po vertikalni osi (uporaba podalgoritmov).



Slika 5: Povečanje kompleksnosti po horizontalni osi



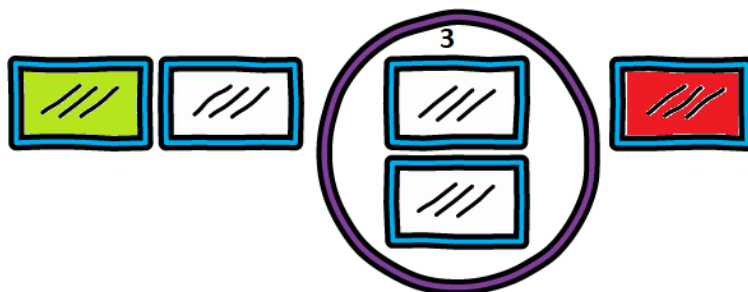
Slika 6: Povečanje kompleksnosti aplikacije z dodajanjem po vertikalni osi

Če so učenci pri evalvaciji ugotovili, da je algoritem postal nepregleden, jih je učitelj usmeril k iskanju optimizirane rešitve.

4. faza: Optimizacija algoritma

Zadnja faza predstavlja optimizacijo algoritma. Učitelj vpelje koncept zanke, ki omogoča poenostavitev ponavljajočih se ukazov. Učitelj doda dve novi zahtevi, s katerima usmeri delo in razmišljanje učencev. Ti dve zahtevi sta:

- večja preglednost,
- skrajšati algoritem.



Slika 7: Primer optimiziranega algoritma

Ob koncu učnega scenarija so učenci izvedli evalvacijo, pri kateri so analizirali svoje algoritme, prepoznali napake in predlagali izboljšave. S pomočjo intervjujev smo preverili njihovo razumevanje temeljnih konceptov, kot so zanke, algoritmično razmišljanje, prepoznavanje vzorcev, abstrakcija in dekompozicija.



Slika 8: Prikaz izvedbe aktivnosti

V raziskavi smo uporabili kvalitativno metodo zbiranja podatkov. Podatke smo pridobili z opazovanjem učencev med izvajanjem aktivnosti, analizo njihovih izdelkov ter polstrukturiranim intervjujem.

Opazovanje je potekalo v naravnem učnem okolju, pri čemer smo beležili ključne interakcije med učenci in učiteljem. Posebno pozornost smo namenili morebitnim težavam pri razumevanju konceptov ter odzivom učencev pri oblikovanju algoritmov. Analiza izdelkov nam je omogočila oceniti, v kolikšni meri so učenci razumeli in pravilno uporabljali koncept algoritmov ter kako so skozi ponovna preizkušanja in izboljšave razvijali svoje rešitve. Vprašanja v polstrukturiranem intervjuju so bila prilagojena starosti učencev in oblikovana tako, da so omogočala odprte odgovore.

Pri analizi odgovorov je potrebno upoštevati vpliv strokovnega delavca. Uspešnost izvedene aktivnosti je odvisna od spretnosti strokovnega delavca, da z opazovanjem in sprotnim podajanjem povratnih informacij usmerja učence, jih motivira in spodbuja k izboljšavam.

Pridobljene podatke smo obdelali z vsebinsko analizo (kodiranjem), pri čemer smo identificirali tematske vzorce, kategorizirali odzive učencev in primerjali odgovore med različnimi skupinami učencev.

3 REZULTATI

Učenčevu predznanje smo preverili s pomočjo vprašanj o osnovnih pojmih računalniškega razmišljanja. Rezultati so pokazali, da so učenci v največji meri seznanjeni s pojmom vrstica (90%) in ukaz (100%), medtem ko so imeli največ težav s poznavanjem pojmov zanka (5%) in algoritem (15%).

- Podatke o učenčevem razumevanju konceptov računalniškega razmišljanja smo pridobili s kombinacijo opazovanja, analize izdelkov učencev in polstrukturiranih intervjujev.
- Opazovanje je pokazalo, da je 50% učencev potrebovalo pomoč pri poenostavljanju algoritma.
- Analiza izdelkov (fizičnih algoritmov in kartic hroščkov) je pokazala, da so učenci najbolj uspešno uporabljali abstrakcijo, saj je 90% učencev brez težav delalo s piktogrami.
- Intervjuji so potrdili, da so učenci razumeli sosledje ukazov (80% uspešnost pri razumevanju smeri branja algoritma), medtem ko je bila pravilna uporaba zanke prisotna pri 50% učencev.

Tabela 2: Pripomoček za opazovanje interakcij in rezultati

Koncept	Indikator	Rezultat (odstotek uspešnosti)
Abstrakcija	Ali je učenec imel težave z delom s piktogrami?	90%
Algoritmčno razmišljanje	Ali je učenec razumel, da si ukazi sledijo od leve proti desni?	80%
	Ali so pravilno uporabljali zanko?	50%
Dekompozicija	Ali so učenci poenostavili algoritem?	50%
Generalizacija in prepoznavanje vzorcev	Ali so učenci uspešno izvedli algoritme?	70%

S pomočjo metode kodiranja smo analizirali odgovore učencev in izpostavili pet najpogostejše uporabljenih besed v njihovih izjavah: hrošček, zanimivo, večkrat, korak, drugačno.

Na podlagi vsebinske analize smo odgovore učencev razvrstili v štiri izstopajoča področja:

Tabela 3: Izstopajoča področja odgovorov

Področje	Ključne besede v odgovorih učencev
Sodelovanje	pomagati, skupaj, pokazati
Motivacija	dobro, zanimivo, še večkrat
Algoritem	hrošček, to je enostavno, korak
Inovativnost	novo, drugačno, gibanje in učenje

Intervjuji so razkrili naslednje ključne vidike učenčevega doživljanja učne ure:

- Napake niso bile negativno doživete – Učenci so poročali, da so delali napake, vendar so bile te predstavljene na zabaven način s konceptom "hroščka", kar je zmanjšalo občutek neuspeha.
- Več praktičnega dela – Pouk se jim je zdel bolj zanimiv, saj so bolj aktivno sodelovali v primerjavi s prejšnjimi urami.
- Sproščeno okolje – Učenci so izrazili zadovoljstvo, da je bila aktivnost izvedena v telovadnici, saj jim je to omogočalo več gibanja in sproščenosti.
- Spodbuda za nadaljnje učenje – Nekateri učenci so izrazili željo, da bi podobne aktivnosti izvajali večkrat.

4 ZAKLJUČEK

Rezultati naše raziskave potrjujejo, da je mogoče temeljne koncepte računalniškega razmišljanja (RIN) uspešno razvijati pri učencih brez uporabe digitalnih naprav. Učni scenarij, ki je temeljil na algoritmih, je učencem omogočil aktivno učenje, sodelovanje in eksperimentiranje s koncepti, kot so abstrakcija, dekompozicija in algoritično razmišljanje. Največji dosežek so učenci pokazali pri razumevanju abstrakcije (90% uspešnosti pri delu s piktogrami), medtem ko so imeli največ težav z razumevanjem in uporabo zanke (le 5% učencev je predhodno poznalo ta koncept, pri izvajanju nalog pa ga je uspešno uporabilo le 50% učencev).

Naša raziskava potrjuje ugotovitve Looija s sodelavci (2018), ki poudarjajo pomen postopnega uvajanja računalniškega razmišljanja – najprej skozi aktivnosti brez naprav, nato pa s postopno integracijo digitalnih tehnologij. Podobno kot pri raziskavi Busuttila in Formose (2020) se je tudi pri nas izkazalo, da fizične in interaktivne metode izboljšujejo motivacijo učencev ter jim omogočajo boljše razumevanje pojmov skozi gibanje in igro. Analiza intervjujev je pokazala, da učenci niso doživljali napak kot nečesa negativnega, kar se ujema z raziskavami Threekunprape in Yasrija (2020), ki poudarjajo, da je igrifikacija pri

učenju RIN ključna za zmanjševanje strahu pred neuspehom in spodbujanje raziskovalnega pristopa. Vpeljava "hroščka" kot simbola za napake je učencem omogočila, da so jih zaznavali kot naraven del učnega procesa, namesto da bi jih dojemali kot neuspeh.

Naša študija primera prispeva k razumevanju, kako lahko že v zgodnjem obdobju izobraževanja spodbujamo ključne koncepte računalniškega razmišljanja na način, ki je prilagojen otroški igri in naravnim oblikam učenja. V prihodnosti bi bilo smiselno nadalje raziskovati dolgoročne učinke takšnih pristopov in jih prilagoditi različnim starostnim skupinam ter učnim okoljem.

LITERATURA

- Busuttil, L., in Formosa, M. (2020). Enhancing computational thinking through unplugged activities in primary education: A systematic literature review. *Journal of Educational Computing Research*, 58(6), 1097-1130. <https://doi.org/10.1177/0735633120915697>
- Chen, W., in Chi, M. (2020). Computational thinking cultivation in unplugged activities. *Education and Information Technologies*, 25(6), 5001-5026. <https://doi.org/10.1007/s10639-020-10203-9>
- Chen Peng, H. (2023). Computational thinking without computers: Exploring the effectiveness of unplugged activities in primary education. *Computers & Education*, 194, 104735. <https://doi.org/10.1016/j.compedu.2022.104735>
- Cuny, J. (2010). Computational thinking: Ten years later. *CSTA Voice*, 5(7), 10-11.
- Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., in Woollard, J. (2019). Computational thinking: A guide for teachers. Computing At School. Dostopno na <https://community.computingatschool.org.uk/resources/5316/single>
- Kovač, M., Bergoč, Š. (ur.), idr. (2015). Posodobitve pouka v osnovnošolski praksi. Športna vzgoja, šport [Elektronski vir]. Zavod RS za šolstvo.
- Looi, C. K., Sun, D., in Teh, L. W. (2018). Towards computational thinking in K-12: Lessons from integrated science, technology, engineering and mathematics education. *Journal of Computers in Education*, 5(4), 475-505. <https://doi.org/10.1007/s40692-018-0119-2>
- Minamide, T., Shikano, H., in Miwa, K. (2020). The effects of computational thinking experiences in early childhood education: A study using programming picture books and robots. *International Journal of Child-Computer Interaction*, 26, 100212. <https://doi.org/10.1016/j.ijcci.2020.100212>
- RINOS. (2022). Temeljna znanja računalništva in informatike od vrtca do srednje šole: Okvir in priporočila. Ljubljana: Ministrstvo za izobraževanje, znanost in šport. Dostopno na <https://www.racunalnistvo-in-informatika-za-vse.si/>
- Saxena, A., Lo, C. K., Shih, J. L., in Chu, S. K. W. (2020). Promoting computational thinking in children with educational robots: A systematic review. *International Journal of STEM Education*, 7(1), 1-18. <https://doi.org/10.1186/s40594-020-00247-8>
- Špur, B. (2015). Reševanje Rubikove kocke kot metoda za spodbujanje algoritmičnega razmišljanja. Diplomsko delo, Univerza v Ljubljani, Pedagoška fakulteta.
- Threekunprapa, W., in Yasri, P. (2020). Unplugged coding using physical computing kits to enhance computational thinking in primary school students. *Education and Information Technologies*, 25(4), 2911-2937. <https://doi.org/10.1007/s10639-019-10083-8>
- CS Unplugged. (2010). CS Unplugged: Computer Science without a Computer. University of Canterbury. Dostopno na <https://www.csunplugged.org/>