



Добринка Р. Кузмановић¹

Универзитет у Београду, Филолошки факултет,
Београд, Србија

Оригинални
научни рад

Александар Д. Бауцал

Универзитет у Београду, Филозофски факултет,
Београд, Србија

Зашто је програмирање „иешко”? Коришћење робота у одлучавању програмирању у неформалном образовном контексту – импликације за праксу

Резиме: Програмирање је дициплинарна компетенција од суштинског значаја за живот у 21. веку. Последњих деценија образовни системи широм света реформишу наставне програме и укључују програмирање/кодирање као засебан предмет или интегрисано у друге предмете, у све млађем узрасту. Осим у формалном, програмирање се учи и у неформалном образовном контексту (НОК), а у процесу одлучавања све се чешће користе образовни роботи. Циљ овог рада јесте упознавање са предностима и изазовима у коришћењу робота Sphero (BB-8 и SPRK+) у одлучавању визуелном програмирању у НОК-у и формулисање практичних препорука. Истраживање је спроведено у оквиру пројекта „Учење за 21. век”, у 17 градова у Србији. Упољњавање уједињеника укључено је 677 ученика ($M_{узр} = 12,16$; $СД = 1,10$), у фокусирујне интервјуе 42 ученика и 5 водитеља радионица. Примењена је дескриптивна квантитативна и квалитативна тематска анализа. Резултати показују да коришћење робота има додатну образовну, мотивациону и забавну вредност у процесу одлучавања програмирању и алгоритамском мишљењу, и из ула учесника и водитеља радионица. Одлучавање у НОК-у има низ предности у односу на традиционалну наставу: учење кроз игру, искуствено, креативно учење и решавање проблема, увид да програмирање може да

¹ dobrinka.kuzmanovic@fil.bg.ac.rs;

<https://orcid.org/0000-0002-6296-0539>

Copyright © 2024 by the authors, licensee Faculty of Education University of Belgrade, SERBIA.

This is an open access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0) (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original paper is accurately cited.

буде занимљиво и не мора да буде „тешко“. У раду су формулисани изазови у подучавању програмирања у НОК-у уз помоћ робота, као и руке за наставну праксу.

Кључне речи: образовни роботи, визуелно програмирање, алгоритамошко мишљење, основношколски узраст, неформални образовни контексти

Увод

Робот је глупа машина све док ми не урадимо нешто да он постане „паметан“, што значи да ми морамо да будемо паметни!

Учесница у истраживању
(ученица 5. разреда)

Од „дигиталних генерација“ очекује се да буду активни креатори дигиталног садржаја, а не његови „конзументи“. Креирање дигиталног садржаја представља једну од пет области *Европског оквира дигиталних компетенција грађана* (енгл. *DigComp*), која укључује програмирање дефинисано као „способност развијања низа разумљивих упутстава за рачунарски систем, у циљу решавања одређеног проблема или извођења специфичног задатка“ (Vuorikari et al., 2022, p. 109). Разумевање природе програмирања кључна је компонента рачунарске писмености/културе и требало би да буде један од главних исхода информатичког образовања, поред вештина решавања проблема и рачунарског размишљања (Brodnik et al., 2021). У дигиталној култури програмирање представља „продужетак“ базичне писмености, јер омогућава писање нових врста „текста“. Учење рачунарског програмирања позитивно доприноси развоју когнитивних вештина, као што су алгоритамошко и логичко-математичко мишљење и решавање проблема, креативно и критичко мишљење, метакогниција итд. (Siegle, 2017; Scherer et al., 2019; Sáez-López et al., 2021).

Током последње деценије у образовно-политичким документима на глобалном нивоу програмирање постаје једна од приоритетних дигиталних компетенција (Balanskat and Engelhardt,

2015; Vocconi et al., 2022). Увођење програмирања у све школе у Европи предуслов је за дигиталну трансформацију образовања (European Commission / EACEA / Eurydice, 2022). У Акционом плану за дигитално образовање (2021-2027) истиче се значај стицања дигиталних компетенција и квалитетног информатичког образовања, како у формалном, тако и у неформалном образовном контексту (European Commission, 2020). Почев од 2014. године, присутан је сталан тренд интеграције програмирања/кодирања, рачунарског и алгоритамошког размишљања у програме наставе и учења, на свим нивоима образовања (Balanskat & Engelhardt, 2015). За ученике у Србији формално подучавање основа програмирања и алгоритамошког мишљења почиње од првог разреда, у оквиру предмета Дигитални свет, који је од школске 2023/24. године обавезан наставни предмет за ученике млађих разреда основне школе. У оквиру предмета Информатика и рачунарство, који је од школске 2017/18. постао обавезан за ученике старијих разреда основне школе, предвиђено је упознавање са визуелним (нпр. Scratch) и текстуалним програмским језицима (нпр. Python, Pygame, Jupyter). Упркос томе, више од трећине дванаестогодишњака у Србији, након две школске године похађања Информатике и рачунарства, према сопственој процени, не уме да користи ниједан програмски језик (Kuzmanović i sar., 2019). Бројна истраживања указују на комплексност процеса подучавања програмирања/алгоритамошког мишљења и потребу за иновативним, на ученика центрираним, педагошким приступима и алатима за учење.

Програмирање и сродни појмови

Рачунарско програмирање (енгл. *computer programming*) јесте процес развоја и имплементације различитих скупова инструкција или изворних кодова написаних у програмском језику како би се омогућило рачунару да изврши одређени задатак, реши проблем и обезбеди људску интеракцију (Balanskat & Engelhardt, 2015). Мада се појмови програмирање и кодирање (енгл. *coding*) неретко користе као синоними, програмирање је сложенији и богатији процес, који укључује анализу, разумевање и решавање проблема, креирање алгоритма и проверу његове исправности, након чега следи кодирање, односно имплементација алгоритма на циљаном програмском језику. Рачунарско размишљање (енгл. *computational thinking*) шири је појам и дефинише се као „фундаментална вештина која укључује решавање проблема, дизајнирање система и разумевање људског понашања, ослањајући се на кључне концепте рачунарске науке” (Wing, 2006, p. 33).

Визуелно, блоковско програмирање (енгл. *block-based visual programming*) – у литератури се користи синтагма „превуци и спусти” програмирање (енгл. *drag and drop*) – односи се на софтвер или апликацију у којој ученици бирају одговарајуће блокове кода и спајају их како би креирали програм (нпр. Scratch, ScratchJr, Alice, Kodu, Blockly итд.) (Cheng, 2019; Myers, 1990). Уместо писања кодова у текстуалном облику, управља се графичким елементима. Емпиријски је потврђено да се визуелни програмски језици, креирани у образовне сврхе, лакше уче него текстуални (Cárdenas-Cobo et al., 2021). Последњих година све се чешће користе у настави програмирања у основношколском, али и млађим узрастима. Већина истраживања о доприносу визуелног програмирања рачунарском размишљању ученика школског узраста спроведена је управо у неформалном образовном контексту (Luo et al., 2020).

Образовни работи у подучавању програмирању

Током последњих деценија образовна роботика (енгл. *educational robotics*) налази све ширу примену као интердисциплинарни образовни алат у различитим дисциплинама: STEAM (енгл. *Science, Technology, Engineering, Arts and Mathematics*) – наука, технологија, инжењерство, уметности и математика, језици, рачунарство итд. (Anwar et al., 2019). Образовна роботика представља један од класичних начина подучавања програмирању који омогућава интерактивност у окружењу физичког програмирања (Sun & Zhou, 2022). Ефикасна је за развијање рачунарског размишљања, јер захтева од ученика да систематски извршавају задатке и развијају секвенционирани, корак-по-корак команде, како би програмирали робота (Chalmers, 2018). Образовни работи налазе примену у различитим (формалним и неформалним) контекстима за учење, на свим нивоима образовања, од предшколског до академског (Huang et al., 2023).

Програмирање робота (енгл. *robot-based programming*) савршено отелотворује теорију конструкционизма С. Паперта (Harel & Papert, 1991), креатора првог програмског језика за децу – LOGO. Својим пионирским радом осамдесетих година прошлог века Паперт је показао да мала деца могу да науче програмски језик и кодирају робота корњачу како би решила проблем (Anwar et al., 2019). Паперт сматра да је учење најефикасније и највише контекстуализовано када ученици сами конструишу ствари, кроз интеракцију са опипљивим објектима, као што су рачунар или робот. Друга кључна теорија која објашњава улогу образовних робота у подучавању програмирању јесте конструктивизам (Piaget, 1970). Заједничко за оба приступа јесте схватање да се знање активно конструише у интеракцији са околином. Коришћење робота има потпору и у теорији искуственог учења (Kolb, 1984), према којој разумевање настаје кроз

трансформацију искуства, а не кроз примену теоријског знања у пракси.

За стицање вештина програмирања и рачунарског мишљења код деце основношколског (и млађих) узраста најчешће се користе алати (окружења) за визуелно програмирање и роботски програмабилни комплети (нпр. Beebot, Mbot, Lego Mindstorms, Micro:bits, Sphero итд.) (Rich et al., 2019; Wei et al., 2020).

Циљеви истраживања и истраживачка питања

Циљ овог истраживања јесте стицање увида у предности и изазове подучавања програмирању у неформалном образовном контексту (НОК), уз коришћење робота, из угла ученика старијег основношколског узраста и водитеља радионица програмирања. На основу добијених емпиријских налаза предложене су практичне препоруке за организовање процеса подучавања програмирању.

У складу с циљевима рада формулисана су истраживачка питања:

1. Који су мотиви учесника за похађање радионица програмирања у НОК-у?
2. Које су предности подучавања програмирању уз помоћ робота у НОК-у – из угла учесника и водитеља радионица?
3. Шта су изазови у стицању вештина програмирања и алгоритамског мишљења у НОК-у уз помоћ робота – из угла учесника и водитеља радионица?

Методологија истраживања

Радионице програмирања. Истраживање је спроведено у оквиру пројекта „Учење за 21. век“, који је реализовао Стартит, у сарадњи са канцеларијом УНИЦЕФ-а у Србији. Циљ пројекта била је подршка развоју базичних вешти-

на програмирања, кодирања и алгоритамског мишљења код ученика старијих разреда основне школе, с фокусом на девојчицама и деци из осетљивих група. Креиран је програм четвородневне обуке, реализован кроз интерактивне радионице са учесницима (најчешће у основним школама и стартап центрима). С обзиром на чињеницу да је за скоро половину младих ово било прво искуство са програмирањем и роботима, ова обука имала је за циљ упознавање са основама програмирања и алгоритамског мишљења (упознавање за апликацијом и роботом, рашчлањивање проблема на кораке, осмишљавање правила и креирање алгоритма, креирање програмског низа блокова, препознавање и исправљање грешака у програму итд.).

Радионице су водили обучени водитељи с вишегодишњим искуством у конципирању и реализовању едукативних радионица с децом основношколског узраста.

Током радионица коришћена су два модела Sphero робота: BB-8 и SPRK+² и апликација за визуелно (блоковско) програмирање SpheroEdu™ App³ (Слика 1).

Слика 1. Sphero SPRK+, BB-8 роботи и коришћени током радионица програмирања (а и б) и SpheroEdu апликација инсталирана на таблети (в).

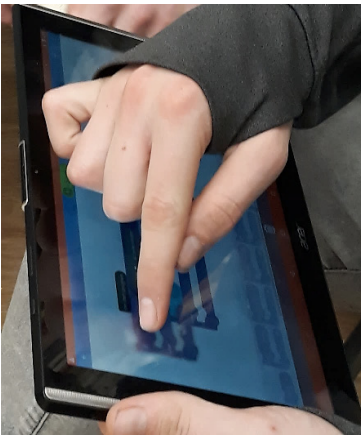


2 <https://www.sphero.com/>

3 <https://sphero.com/pages/apps>



(б)



(в)

Реч је о роботској лопти која поседује програмабилне сензоре и ЛЕД светла, а покреће се помоћу апликације која је доступна на различитим платформама (нпр. iOS, Android, Windows) и може се (бесплатно) инсталирати на таблет, мобилни телефон или рачунар.

Врста истраживања и инструменти. Реализовано је комбиновано (миксметодско) истраживање. Полазници су након последње радионице попуњавали краћи упитник који се односио на мотивацију за учешће у радионицама (питање отвореног типа), степен задовољства радионицама и самопроцену стечених знања и вештина (петостепена скала Ликертовог типа).

У оквиру квалитативног истраживања на основу унапред припремљених водича органи-

зовано је шест фокусгрупних интервјуа са полазницима и један са водитељима радионица (након последње радионице). Уз сагласност ученика у истраживању (у случају деце тражена је и сагласност родитеља/старатеља) интервјуи су снимани, а на основу аудио-записа сачињени су транскрипти, који су коришћени за тематску анализу (Braun & Clarke, 2006; Vilig, 2013).

Узорак. У квантитативно истраживање укључено је 677 ученика, из 17 градова у Србији, узраста између 11 и 14 година ($M_{\text{узраст}}=12,16$; $SD=1,10$), 445 девојчица (65%) и 232 дечака (35%). У узорку је: 30% ученика петог разреда, 32% ученика шестог, 23% седмог и 15% ученика осмог разреда.

У фокус групе са полазницима било је укључено 42 деце (64% девојчица и 36% дечака), из 6 градова у Србији. У фокус групи са водитељима учествовало је 5 особа (1 мушког и 4 женског пола).

Резултати истраживања и дискусија

Мотивација за похађање радионица програмирања. У Табели 1 приказани су одговори ученика на питање *Зашто си се пријавио/ла на радионице*, које се односи на њихову мотивацију за учешће у радионицама. На ово питање, задато у оквиру упитника, одговорило је 88% (593) испитаника. Одговори су разврстани у осам тема и више подтема.

Табела 1. Мотивација ученика за похађање радионица програмирања роботика.

Врсте мотива	Одговори ученика	N	%
1. Интересовање за роботе и програмирање		265	44,5
Занимају их роботи и роботика	– <i>Зашто што волим роботике, роботика су забавни.</i> – <i>Желео сам да видим како роботика изгледа уживо.</i> – <i>Занимало ме је како роботика функционишу.</i> – <i>Желела сам да сазнам што више о роботикама.</i> – <i>Зашто што волим све што се тиче роботике.</i>	150	25,3
Желе да науче да програмирају роботе	– <i>Да научим како да програмирам роботика.</i> – <i>Желео сам да научим како се креће роботика.</i>	27	4,5
Занима их програмирање и/или информатика	– <i>Волим програмирање.</i> – <i>Желим да научим да програмирам.</i> – <i>Волим програмирање и Информатику.</i>	72	12,0
Занимају их роботи и програмирање	– <i>Зашто што волим роботика и програмирање.</i>	16	2,7
2. „Звучало” им је занимљиво, забавно, привлачно...		216	36,4
Радозналост, знатижеља	– <i>Из радозналости.</i> – <i>Занимало ме је шта се ради на овим радионицама.</i>	35	5,9
Забавно, занимљиво	– <i>Мислила сам да је забавно и уверила сам се.</i> – <i>Звучало ми је забавно.</i> – <i>Деловало ми је интересантно.</i>	114	19,2
Жеља за стицањем новог знања и искуства	– <i>Хтела сам да сазнам, научим нешто ново, проширим своје знање.</i> – <i>Желела сам да стекнем ново искуство.</i>	67	11,3
3. Дружење са вршњацима и учење		52	8,8
Дружење	– <i>Због дружења са другарима.</i>	11	1,9
Дружење и учење	– <i>Зашто што сам први пут учила да програмирам, упознала нове пријатеље и постојала нови човек, у добром смислу.</i> – <i>Због дружења са другарима и роботикама.</i>	41	6,9
4. Пријавили су их други		25	4,1
Наставници	– <i>Наставница ме је одабрала.</i> – <i>На позив разредне.</i>	18	3,0
Родитељи	– <i>Родитељи су ме пријавили.</i> – <i>Мама ме је пријавила, видела је на интернету.</i>	5	0,8
Вршњаци	– <i>Због другарице.</i>	2	0,3
5. Практична примена наученог		9	1,7
Користиће им у школи	– <i>Користиће ми на Информатици и рачунарству.</i> – <i>Треба ми за школу.</i>	3	0,5

Користиће им у будућој професији	– Хоћу да уишем Информатику. – У будућности желим да се бавим роботиком. – Желим да постанем програмер.	5	0,8
Шири, друштвени значај програмирања	– Желим да знам да програмирам, можда у будућности моју временити човечанство.	1	0,2
6. Модели у породици		1	0,2
Родитељи се баве програмирањем	– Мој тата програмира, ја сам хтела и ја.	1	0,2
7. Начин подучавања		6	1,0
Радионичарски рад	– Зашто што волим радионице.	6	1,0
8. Други разлози	– Не знам. – Зашто што сам добра у математици. – Зашто што волим ирице. – Збој хране.	19	3,6
		593	100

Најчешћи мотив за пријављивање на радионице јесте заинтересованост за работе. На другом месту је очекивање полазника да ће им на радионицама бити занимљиво, забавно, изазовно и да ће стећи нова знања и искуства. Упркос чињеници да је данас занимање програмера једно од најтраженијих и најбоље плаћених занимања, мање од 2% младих наводи инструменталне мотиве, корист од програмирања у академској сфери или будућој професији. У Америци ученици не разумеју вредност или сврху програмирања, не повезују га са стварним животом и доживљавају га као игру (Elsawah & Hill, 2023).

Девојчице из нашег узорка чешће од дечака као мотив за похађање радионица наводе подстицај од стране одраслих (наставника и родитеља), о чему сведоче и друга истраживања (Cheng, 2019), што се доводи у везу са индукваном нижом дигиталном самоефикасношћу девојчица, слабијим интересовањем и родним стереотипима о традиционално „мушким занимањима“ (Aivaloglou & Hermans, 2019; Montuori et al., 2022). Резултати више међународних истраживања указују на статистички значајне ро-

дне разлике у корист дечака, како у погледу дигиталне самоефикасности, односно процене властитих дигиталних вештина, тако и у погледу спремности да своју професионалну каријеру граде у ИТ сектору (OECD, 2019; Elsawah & Hill, 2023; Smahel et al., 2020). С друге стране, резултати директног процењивања дигиталних компетенција ученика негирају постојање родних разлика (npr. Kuzmanović, 2017; Sun et al., 2022).

Предности учења програмирања у НОК-у уз помоћ робота

Предности из уџла полазника радионица. У Табели 2 приказани су одговори на тврдње задате у оквиру упитника (на петостепеној Ликертовој скали) које се односе на степен задовољства полазника различитим аспектима радионица.

Табела 2. Задовољство радионицама – родне разлике.

	М (SD)		Разлика	
	Девојчице	Дечаки	t	p
Радионице су занимљиве.	4.87 (.43)	4.74 (.69)	2.51	.012*
Радионице су забавне.	4.88 (.47)	4.77 (.62)	2.35	.019*
Радионице су корисне.	4.77 (.59)	4.71 (.68)	1.24	.216
Радионице су захтевне.	2.60 (1.36)	2.46 (1.39)	1.29	.197
Свиђа ми се начин на који смо учили да програмирамо.	4.76 (.55)	4.68 (.63)	1.56	.118
Свиђају ми се апликације које смо користили.	4.60 (.72)	4.46 (.79)	2.27	.027*
Дешавало ми се да не разумем шта треба да радим.	3.39 (1.46)	3.09 (1.59)	2.42	.016*
Лакше ми је када радимо у групи него самостално.	4.20 (1.13)	4.19 (1.12)	.11	.910
На радионицама сам научио/ла да сарађујем.	4.22 (1.00)	4.30 (.96)	-.969	.333
На радионицама сам научио/ла много нових ствари.	4.46 (.74)	4.42 (.85)	.612	.541
Заинтересован/а сам да поново похађам радионице.	4.57 (.71)	4.58 (.73)	-.117	.907

* $p < .05$

Као што се види из горње табеле, степен задовољства радионицама веома је висок. Налаз је потврђен у другим истраживањима, такође на испитаницима основношколског узраста (нпр. Park et al., 2015). Родне разлике нису изражене, радионице су нешто занимљивије и забавније девојчицама и више им се свиђају коришћене апликације, међутим, оне чешће од дечака (57% према 50%) изјављују да им се дешавало током радионица да не разумеју шта треба да раде. Важно је нагласити да се скоро половина (42%) полазника, подједнако девојчица и дечака, у оквиру ових радионица први пут сусрела са неким програмским језиком (иако је већина њих похађала Информатику и рачунарство у школи), с тим што су девојчице ређе од дечака (11% према 19%) имале искуства у коришћењу програмских језика у изваншколском контексту.

Дубљи увид у предности коришћења робота у учењу програмирања у НОК-у, из перспективе полазника, стечен је на основу тематске анализе података добијених током фокусгрупних интервјуа. Издвојене су три теме: робот као алат за учење и забаву, педагошки приступ у подучавању и атмосфера и односи у групи. У оквиру

сваке теме дефинисано је више подтема. Учесници су своја искуства стечена током радионица често поредили са учењем програмирања на часовима Информатике и рачунарства, и када то од њих није тражено.

1. тема: Робот као алат за учење и забаву (учесници)

- ☑ Могућност за аутоматско тестирање кода и непосредан увид у његову исправност једна је од главних предности подучавања програмирању помоћу робота.
 - *Можемо да тестирамо наш код помоћу робота, ако не изврши команду коју смо му задали, знамо да смо нешто поједили.*
- ☑ Атрактивност и мултифункционалност робота:
 - *Роботи су занимљиви: крећу се, скачу, преврћу, мењају боју, производе светло и звук... Још када бисмо имали неке „праве“ роботе, нпр. да нам робот донесе њескавицу. 😊*
 - *Програмирали смо робота кроз игру „Врући кромчирићи”: он је прво љаве боје, када крене пребације се у црвену и на крају експлодира...*
 - *Многи је леише када робот најави квадрант, него када га на екрану нацртамо...*

- ☑ Учење програмирања у игровном контексту:
 - Увек смо радили кроз неке игре, што је занимљивије, није чиста теорија... нпр. две особе окренуће леђима једна ка другој и прва мора да објасни речима другој шта робот треба да ради, како би ова програмирала – игра се зове „Програмери милионери“! 😊

2. тема: Педагошки приступ у подучавању (учесници)

- ☑ Радионичарски начин рада:
 - На радионицама је динамично, узбудљиво, смешно, нејредвидљиво, постоје када се међусобно такмичимо... • Никада није досадно, радимо оно што нам се свиђа.
- ☑ Рад у малим групама и колаборативно решавање проблема:
 - Ја сам радила са мојим тимом, заједно смо смислиле решење (додавале степен, секунде итд.), што је много лакше него када радим сама..
- ☑ Развој креативности:
 - Најзанимљивије је било када смо сами смишљали своју игру. • Наши наставник Информатике воли да се држи плана, једино када имамо тесну дају нам да ишито сами своје програме, а овако нам увек он задаје нешто.
- ☑ Учење стратегија учења:
 - Учимо да решавамо проблеме „корак по корак“. • Наставница нам је рекла да ми овде учимо ствари које ће нам помоћи да решимо неке друге проблеме у животи...
- ☑ Коришћење мобилних уређаја, телефона и таблета уместо рачунара:
 - Занимљивије ми је да програмирам у апликацији на мобилним уређајима него на рачунару. • Није ми се свидело преко компјутера, иако имам иеи из Информатике.
- ☑ Увид да програмирање може да буде занимљиво:
 - Схватила сам да програмирање може да буде занимљиво, постоје ако се учи на радионицама.

3. тема: Атмосфера и односи у групи (учесници)

- ☑ Мање је стресно него на часовима, нема временског притиска, не оцењује се:
 - На часу се многе жури, некада не стигнемо да напишемо све команде, изоставимо неко слово и појешимо... Са овом наставницом смо све радили напунене... • Нема негативних последица ако се појешо, лоших оцена.
- ☑ Слобода у тражењу додатних појашњења од водитеља, одсуство страха од грешке:
 - Није ми тешко да ишам ако нешто не знам. • Сремни су сваком да помоћу, не мрзи их да понове... • И ако нешто појешимо, наставница нам објасни, нпр. ако не знамо значење неке речи на енглеском.
- ☑ Позитиван однос према водитељима радионица:
 - Имају смисла за хумор. • Мисле на нас. • Воле децу! Да нас не воле, не би ово ни радили!

Предности из угла водитеља радионица.

На основу тематске анализе одговора водитеља издвојене су такође три теме, као код учесника радионица, али се подтеме делимично разликују.

1. тема: Робот као алат за учење и забаву (водитељи радионица)

- ☑ Као и полазници радионица, и водитељи сматрају да се вештине програмирања лакше и брже стичу уз коришћење робота:
 - Деца много лакше науче да израчунају улове квадрата и нацртају квадрат правилно ако имају робота, иј. моћив да то ураде... • Понашање робота је најбоља повратна информација.
- ☑ Примена робота омогућава искуствено учење, које изостаје у класичном инструктивничком подучавању програмирању, доминантном у формално-образовном контексту:
 - Теоријско знање прожеито личним искуством, директном применом, учи се на примеру, деца кроз праксу могу

директно да испробају да ли функционише алтернатива која су најпробавили.

- ☑ Роботи су компатибилни са алатима за визуелно програмирање, који су, када је реч о учењу првог програмског језика, добра алтернатива традиционалним алатима за текстуално програмирање: • Једна од предности роботиа јесте то што су компатибилни са апликацијама за визуелно програмирање, које је много једноставније него текстуално, пошто је за млађу децу и децу која се са програмирањем први пут срећу.
- ☑ Роботи су привлачни и занимљиви деца, што је добра основа за „увлачење” у програмирање: • Много деца не би дошла на ове радионице да није било роботиа... Још када би имали неке најпреднији роботие. • Чак и они који дођу незаинтересовани, у једном моменту се „укаче”.

2. тема: Педагошки приступ у подучавању (водитељи радионица)

- ☑ Радионичарски начин рада: • Деца воле радионице јер нису формалне, нема оцењивања, кажњавања... • На радионицама су, не зато што морају, већ зато што сами желе, а у сваком тренутку могу да одустану, без негативних консекуенци.
- ☑ Приступ подучавању није традиционално школски: • Не намећемо им дефиницију, већ они сами долазе до ње, и онда ми то можемо да обликујемо, да буде практична. • Смер учења је обрнут, од праксе ка теорији...
- ☑ Стицање трансферних знања и стратегија решавања проблема примењивих у школском и изваншколском контексту: • Деца су учила практичноје решавања проблема. • Ја сам инсистирао да прво прочитају код који пишу и изврше логичку проверу, зато што они најчешће конструишу блокове и желе одмах да „испеку”. • Углавном циљ ове радионице је ПРОБАЈ! Пробај и види

што ће да буде, није практично ако не ради, пробај да пронађеш соодветну праксу и да је испробиш. То је најважнија добит од ових радионица...

- ☑ Радионице су прилика да се препознају деца која имају предиспозиције за програмирање, а нису имала ову врсту искуства (нпр. деца из осетљивих социјалних група, нижег СЕС-а): • Имали смо необразованог ромског дечака од 12 година који једва сриче, али је природно технички талент, за 30 минута схватио како работи ради, који су му делови, како се креће... и, до краја праксе дана он је „мајстор”! Ако због ове радионице тај клинац одлучи да се бави нечим сличним, ја сам срећан!
- ☑ Стицање увида у властита интересовања и способности: • Не морају сви да се баве програмирањем, у добити су и они који схвати да то не желе и зато не желе!

3. тема: Атмосфера и односи у групи (водитељи радионица)

- ☑ Когнитивни циљеви нису у првом плану: • Нама као вршњачким едукаторима битнији су ти млади људи него програмирање! Програмирање је „успутна станица”.
- ☑ Близак однос с децом: • Однос је неједнаки, блискији, слободнији, блискији.

Дакле, и учесници и водитељи радионица сматрају да је најважнија предност работа, као алата за учење програмирања, могућност за пружање непосредног увида у исправност кода. Због чињенице да су занимљиви и забавни образовни работи имају висок потенцијал за мотивисање ученика и стимулисање на активно трагање за решењем проблема, што је потврђено и у другим истраживањима (нпр. Tengler et al., 2020). У нашем истраживању обе циљне групе истичу предност НОК-а. У бројним студијама истражује се ефикасност примене работа у различитим формалним и неформалним контекстима за учење, од основног до универзитет-

ског образовања, међутим, недовољно је доказа о утицају на исходе учења и друге вештине ученика (нпр. колаборација, комуникација) (Anwar et al., 2019). Из перспективе учесника радионица висок проценат остварио је предвиђене исходе учења, међутим, не треба занемарити чињеницу да се ради о самопроценама.

Изазови у учењу програмирања у НОК-у уз коришћење робота – из угла учесника и водитеља радионица

Изазови из угла учесника радионица. На основу тематске анализе одговора учесника радионица издвојене су три теме, односно три општије категорије изазова у подучавању програмирању у НОК-у уз помоћ робота и у оквиру њих више подтема.

1. тема: Програмирање је тешко и захтевно (учесници)

- ☒ Ангажује сложеније когнитивне процесе:
 - *Мора дообро да се размишља да би се уклопили „блокови“.*
 - *Тешко је, теже од других предмета, слично математизици.*
- ☒ Захтева прецизност, фокусираност и пуну пажњу:
 - *Да би нас робот разумео и урадио шта желимо, морамо да му дамо прецизне и јасне инструкције, иначе, од тога посла нема ништа!*
 - *Испробујмо неку команду, попробујмо само једно слово, кликнуемо на „ентер“ и не знамо где је грешка, онда морамо све испочетка...*
- ☒ Изискује предзнање из одређених предмета:
 - *Рачунари боље разумеју енглески, него српски... Нпр. ако напишеш: Hello, what's your name?, он ће да каже име, али ако напишеш: Како се зовеш?, неће!*
 - *Не можеш да знаш за колико ствари треба да се окрене робот, ако не знаш математику.*
 - *Ко је добро научио Скреч, њему је ово било лако.*
- ☒ Тражи много учења и континуирану посвећеност:
 - *Мора стално да се учи*

и „ајдејтује“ знање, јављају се нови програмски језици...

- ☒ Физички је напорно, много времена испред екрана:
 - *Програмери много дуго седе испред екрана.*
 - *Заболује глава и очи од гледања у екран, уочи ми се враћ.*

2. тема: Родне разлике (учесници)

- ☒ Девојчице су мање заинтересоване за програмирање од дечака и недостаје им самопоуздање:
 - *Дечацима боље иде програмирање, они знају више времена проводе на компјутеру, телефону, више су упућени у то.*
 - *Дечаци током живота више заволе компјутере, због игрица, а програмирањем може да се формира своја игрица.*
 - *Дечацима раније кује рачунар него девојчицама.*
 - *Ми ако нешто попробујемо, ми бисмо, претворили смо се, док дечаки желе да виде на интернету у чему су попробујали и ваљда иако схватили... не знам, не умет да објасним...*
 - *Девојчице занима нешто реалније, нпр. цртање.*

3. тема: Изазови техничке природе (учесници)

- ☒ Роботи нису лако доступни, поготову они комплекснији („хуманоидни“):
 - *Волео бих да смо имали додатне ствари које би укључили у игру с роботима, или да имамо неке другачије или савршеније роботе.*
 - *Има један робот који се зове Виктор (може да се наручи преко сајта), он може да помера ствари, љуби се, срећан је, љубан, сјава...*
- ☒ Радионице кратко трају:
 - *Волео бих да радионице трају дуже.*
 - *Да можемо да обновимо ово што смо научили, овако ћемо заборавити.*

Изазови из угла водитеља радионица.

1. тема: Недовољно предзнања и тешкоће у примени (водитељи радионица)

- ☒ Недовољно предзнање из Информатике и рачунарства:
 - *Било је деце која нису знала ни шта је Скреч, или су знали да постоји,*

али никада у њему нису радили, иако је њо њројраму обавезан... • Посиоје ојромне разлике у начину и квалифицију њодучавања наставника, шиио је очииледно шииоком ових радионица. • Деца која не њоседују базична знања брзо њосијану незаинтересована за сииицање најреднијих вешиина.

- ☒ Недостатак знања из сродних предмета (нпр. математика, енглески језик) важних за учење програмирања: • Да би знали да најраве формулу квадрати, морају да знају шиио да ураде у математици. • Они који немају шиио знање морају да се сналазе како знају и умеју, да цршају, раде сисииемом њокушаја и њојрешака, и онда шиио њресииаје да буде „cool“!
- ☒ Школско знање је инертно, формализовано: • Чак и када њоседују знање, шииешко им је да њовежу са овим шиио раде на радионицама. • Некој деци су шииребала два дана да схвати сличности између Скреча и апликације СфероЕгу! • Као да њосииоји неки „баи“ у начину њодучавања, који не њредвиђа њрактиичну ујошреду. • Тешко им је када у финалној фази (након шиио су савладали њеиљу, услове ако-онда, алјорииме ишг.) шииреба сами да осмисле, њовежу и инииејришу све шиио су научили шииоком њрејходних радионица.

2. тема: Недостатак креативности и самопоуздања (водитељи радионица)

- ☒ Недостатак креативности код већине деце: • Данашња деца као да не схватијају да ће биши усииешни њројрамери само ако буду креативни, већина шииражи да им се да задатјак и не њоказују жељу да смисле нешиио своје... • Деци се „не да“ да њраве личне њројекти! Радије шиираже задатјак, неіо шиио іа сами смишљају...
- ☒ Недостатак самопоуздања, страх да се испроба ново (поготову код девојчица): • Деца која се не сналазе најдоље на радионицама знају да буду несијурна и

ујлашена, али се обично ослободе. • Била је једна девојчица која није никако хииела да узме шииаблеи у руке, іоворила је: „Ја шиио не знам, нећу“... На крају су је најоворили да ишак њроба...

3. тема: Недостатак интересовања за програмирање (водитељи радионица)

- ☒ Доминантно интересовање за роботе, а не за програмирање: • Реиика су деца која се њасионирано заинтересују за њројрамирање! Најлеишеим је да возеродіа, цршају уиушању шиио којој се робоіи креће, њоседно девојчице... Ту се осећају сијурно, не морају мноіо да се дају, а емошивно добију мноіо. • Већина њих, њоіошову у овом узрасіи, није сушиински заинтересована за њројрамирање, ниши за сииицање најреднијих диіишлних вешиина.
- ☒ Отпор према усвајању стручних појмова из области: • Моје искуствіо је да леіо њрмењују алјоришам, али им се не свиђа шииа реч, не желе да је имају у свом вокадулару. • Мислим да им је њревише шииручних информација, њоіошову за краіко време.
- ☒ Недостатак мотивације за учење Информатике и рачунарства: • Деца на Информатику не іледају као на нешиио корисно, осим оних која намеравају шииме да се даве заіио шиио су чули да њројрамери добро зарађују или желе да њраве видео-ііре. Однос њрема знању чак није ни њраімаишчан, они само желе да добију оцену и да се реше родишља.
- ☒ Родне разлике постају све израженије са узрастом: • До 12,13 іодина девојчице су веома заинтересоване, нема разлика, са 14 іодина њочиње грушиивено индукована незаинтересованосіи за шииехноіију, одједном шиио њресииаје да буде „cool“, и с њима мора мноіо више да се ради да дисиие их њокренули.

4. тема: Изазови техничке природе (водители радионица)

- ☒ Роботи су потрошна роба: • *После одређеног времена роботи почињу да се троше, старе, краће им траје батерија, физички се оштећују услед уопште.*
- ☒ Узрасна хетерогеност и величина група: • *Ради на три нивоа, што је веома захтевно уколико један водитељ реализује радионицу. • Некада су групе веће од оптималне броја.*
- ☒ Организациони проблеми: • *Нема увек довољно простора, неке ствари не могу да се реализују.*

Као најважније изазове у учењу/подучавању програмирању у НОК-у помоћу робота учесници и водитељи наводе: когнитивну комплексност, недостатак предзнања и родне разлике у погледу интересовања. Водитељи указују на недовољно развијене трансверзалне компетенције ученика, које су последица традиционалног приступа подучавању у школском контексту, као и на примарно интересовање за роботе, а не за програмирање. У литератури постоји већи број радова који се баве изазовима/тешкоћама у учењу програмских језика на различитим нивоима образовања, као и стратегијама за њихово превазилажење (Bosse & Gerosa, 2016; Massoudi, 2019). Као најчешћи изазови наводе се: природна когнитивних захтева који су у основи програмирања (висок ниво апстракције, сложена синтакса), недовољно развијене когнитивне вештине, недостатак интересовања за учење програмирања и неодговарајући приступи и методе у подучавању (коришћење инертних материјала за подучавање динамичких концепата) (Gomes & Mendes, 2007; Elsayah & Hill, 2023). На глобалном нивоу постоји проблем „осипања” студената и неуспеха на предметима и студијским програмирањима из области програмирања (Cheah, 2020).

Како показују истраживања, ставови према програмирању значајно предвиђају вешти-

не рачунарског размишљања, поготову када је реч о девојчицама: чак и када имају напредније вештине од дечака, њихови негативнији ставови према програмирању и нижа рачунарска само-ефикасност могу утицати на даљи развој вештина (Sun et al., 2022). Бројна истраживања указују на неопходност подучавања девојчица програмирању што раније током основне школе, јер са узрастом постају све подложније утицају стереотипних класификација „предмета за дечаке” и „предмета за девојчице” (Scherer et al., 2019).

Ограничења истраживања

Узорак ученика није репрезентативан, што онемогућава генерализацију добијених налаза. У узорку ученика заступљеније су девојчице него дечаки. Налази о образовној добити коришћења робота у подучавању програмирања у НОК-у нису поуздани, с обзиром на то да су добијени на основу самоизвештаја. Као и у већини сличних студија реализованих у НОК-у (Gardner et al., 2022), не може се говорити о дугорочним ефектима радионица, поготову у когнитивном домену.

Закључак и препоруке

Од пресудног значаја јесте давање тим младим особама и њиховим општедама, зашто им је програмирање (не)важно и на који начин ће га усвојити!

Учесница у истраживању
(водитељка радионица)

Резултати нашег истраживања показују да је учење програмирања „тешко” или захтевно – не само когнитивно већ и емоционално и физички, како из перспективе оних који га уче, тако и оних који га подучавају, али да га иновативни педагошки приступи, прилагођени потребама ученика, могу учинити „мање тешким”.

Коришћење Sphero робота као алата за подучавање визуелном програмирању и алгоритамском мишљењу у старијем основношколском узрасту има додатну педагошку, мотивациону и забавну вредност. Принцип разумевања кода анализом понашања физичког робота, односно непосредан увид у исправност кода, олакшава учење програмирања и има упориште у више теоријских приступа. Педагошки ефикасним показали су се облици учења примењиви у радионичарском приступу: искуствено, проблемско и пројектно учење, учење путем открића, колаборативно, кооперативно и учење у игровном контексту. „Гејмификација” представља високо вреднован приступ у развијању интересовања и интринзичке мотивације ученика, поготову у фази „увођења” у област програмирања.

С обзиром на налаз да је велики проценат деце заинтересован за програмирање робота, не због програмирања већ због робота, како би се подстакла и одржала мотивација за учење и професионално бављење програмирањем, неопходно је обезбедити више предуслова.

Од раног основношколског узраста требало би развијати алгоритамско, креативно мишљење и решавање проблема и подржати стицање знања из дисциплина за које је потврђено да су неопходне за овладавање програмским језицима (тзв. STEM дисциплине). Када се говори о предусловима за успех у програмирању, налази показују да је, осим когнитивних, и те како важно подржати некогнитивне способности, попут истрајности, толеранције на фрустрацију, саморегулације емоција и узети у обзир имплицитна уверења о властитим способностима, претходна искуства, ниво дигиталне и домен специфичне самоефикасности, ставове према програмирању (поготову код девојчица). Бројна истраживања, укључујући и наше, указују на друштвено индуковано, слабије интересовање девојчица за програмирање, које се повећава са узрастом и постаје најизраженије на крају оба-

везног образовања: *било ми је занимљиво, али што ипак није нешто што бих ја волела да радим у живоју.* Стога је важно од раног основношколског узраста, па чак и у предшколском, системски подстицати и подржавати интересовање девојчица за програмирање и утицати, на ширем друштвеном плану, на смањење родних стереотипа о „мушким” и „женским” занимањима.

Традиционални, инструктивистички приступ подучавању отежава учење програмирања, те је стога неке од принципа учења у НОК-у пожељно применити и у формалном: што више могућности за самостално испробавање, слободно изражавање идеја и учење из сопствених грешака. Групе у којима је више овакве деце знатно су продуктивније, а појединци успевају да „повуку” целу групу. Наши испитаници указују на неопходност промена у приступу подучавању програмирања у формално-образовном контексту. Исказано речима ученика, *на радионицама је занимљивије, на Информатистици нам само показују на рачунару шта треба да радимо, нема ипре... Овде имамо робота, у школи немамо ништа. Да се укине Информатика каква је сада, са роботима би све било лакше и више бисмо се шрудили!*

За успех у подучавању програмирању, осим наведеног, веома је важна позитивна атмосфера у групи и подржавајући односи, који се обично везују за НОК. Када нема строге дисциплине и када се не прибојавају да траже додатна појашњења, деца креативније мисле и продуктивнија су у решавању проблема.

И најзад, према речима једне од водитељки радионица, *у програмирању је важно знаћи зашто, а не само поседовати алатку. Најважније је да постоји то нешто што младе покреће и за шта ће они везати свој идентитет!*

Литература

- Anwar, S., Bascou, N. A., Menekse, M., & Kardgar, A. (2019). A Systematic Review of Studies on Educational Robotics. *Journal of Pre-College Engineering Education Research (J-PEER)*, 9(2), Article 2. <https://doi.org/10.7771/2157-9288.1223>
- Balanskat, A., & Engelhardt, K. (2015). *Computing our future: Computer programming and coding priorities, school curricula and initiatives across Europe*. European Schoolnet.
- Bocconi, S., Chiocciariello, A., Kampylis, P., Dagienė, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M. A., Jasutė, E., Malagoli, C., Masiulionytė-Dagienė, V., & Stupurienė, G. (2022). *Reviewing Computational Thinking in Compulsory Education*. Publications Office of the European Union.
- Bosse, Y., & Gerosa, M. A. (2016). Why is programming so difficult to learn? Patterns of Difficulties Related to Programming Learning. *ACM SIGSOFT Software Engineering Notes*, 41(6), 1-6. <http://doi.acm.org/10.1145/3011286.3011301>
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77–101. <https://doi.org/10.1191/1478088706qp063oa>
- Brodnik, A., Csizmadia, A., Futschek, G., Kralj, L., Lonati, V., Micheuz, P., & Monga, M. (2021). *Programming for All: Understanding the Nature of Programs*. ArXiv:2111.04887 [Cs]. <https://arxiv.org/pdf/2111.04887.pdf>
- Cárdenas-Cobo, J., Puris, A., Novoa-Hernández, P., Parra-Jiménez, A., Moreno-León, J., & Benavides, D. (2021). Using Scratch to improve learning programming in college students: A positive experience from a non-weird country. *Electronics*, 10(10), 1180. <https://doi.org/10.3390/electronics10101180>
- Chalmers, C. (2018). Robotics and computational thinking in primary school. *International Journal of Child-Computer Interaction*, 17, 93-100. <https://doi.org/10.1016/j.ijcci.2018.06.005>
- Cheah, C. S. (2020). Factors contributing to the difficulties in teaching and learning of computer programming: A literature review. *Contemporary Educational Technology*, 12(2), ep272. <https://doi.org/10.30935/ced-tech/8247>
- Cheng, G. (2019). Exploring factors influencing the acceptance of visual programming environment among boys and girls in primary schools. *Computers in Human Behavior*, 92, 361–372. <https://doi.org/10.1016/j.chb.2018.11.043>
- Elsawah, W., & Hill, C. (2023). Barriers to programming education in UAE primary schools: a qualitative review from ICT teachers' perspectives. *Discover Education*, 2(20). <https://doi.org/10.1007/s44217-023-00043-0>
- Europska komisija (2020). *Akcijski plan za digitalno obrazovanje 2021–2027. Prilagodba obrazovanja i osposobljavanja digitalnom dobu*. <https://eur-lex.europa.eu/legal-content/HR/TXT/PDF/?uri=CELEX:52020DC0624&-from=EN>
- Europska komisija / EACEA / Eurydice (2022). *Informatičko obrazovanje u školama u Evropi*. Izvješće Eurydicea. Ured za publikacije Europske unije.
- Gardner, T., Leonard, H. C., Waite, J., & Sentance, S. (2022). What do We Know about Computing Education for K-12 in Non-formal Settings? A Systematic Literature Review of Recent Research. *Proceedings of the 2022 ACM Conference on International Computing Education Research, Volume 1* (str. 264–281). <https://doi.org/10.1145/3501385.3543960>

- Gomes, A., & Mendes, A. J. (2007). Learning to program - difficulties and solutions. *International Conference on Engineering Education – ICEE 2007*, 7. CEE. https://www.academia.edu/14406171/Learning_to_program_difficulties_and_solutions
- Harel, I., & Papert, S. (1991). *Situating constructionism*. Ablex Publishing Corporation.
- Huang, R., Liu, D., Chen, Y., Adarkwah, M. A., Zhang, X.L., Xiao, G.D., Li, X., Zhang, J. J., & Da, T. (2023). *Learning for All with AI? 100 Influential Academic Articles of Educational Robots*. Smart Learning Institute of Beijing Normal University.
- Jacob S. R., & Warschauer M. (2018). Computational thinking and literacy. *Journal of Computer Science Integration*, 1(1), 1–21. <https://doi.org/10.26716/jcsi.2018.01.1.1>
- Kolb, D. A. (1984). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice-Hall, Inc.
- Kuzmanović, D. (2017). *Empirijska provera konstrukta digitalne pismenosti i analiza prediktora postignuća* (doktorska disertacija). Filozofski fakultet Univerziteta u Beogradu.
- Kuzmanović, D., Pavlović, Z., Popadić, D. i Milošević, T. (2019). *Korišćenje interneta i digitalne tehnologije kod dece i mladih u Srbiji: rezultati istraživanja Deca Evrope na internetu*. Institut za psihologiju Filozofskog fakulteta u Beogradu.
- <https://www.lse.ac.uk/media-and-communications/research/research-projects/eu-kids-online/participating-countries/national-languages/serbia>
- Kuzmanović, D. (2023). *Informatika i računarstvo u osnovnim školama u Srbiji: empirijski nalazi i praktične implikacije*. Zavod za vrednovanje kvaliteta obrazovanja i vaspitanja (u procesu objavljivanja).
- Luo F., Antonenko P. D., & Davis E. C. (2020). Exploring the evolution of two girls' conceptions and practices in computational thinking in science. *Computers & Education*, 146, 103759. <https://doi.org/10.1016/j.compedu.2019.103759>
- Massoudi, M. (2019). A review on challenges and solutions in learning programming courses at undergraduate level. *International Journal of Applied Research*, 5(8), 146–149. <https://www.allresearchjournal.com/archives/2019/vol5issue8/PartC/5-8-51-648.pdf>
- Montuori, C., Ronconi, L., Vardanega, T., & Arfé, B. (2022). Exploring Gender Differences in Coding at the Beginning of Primary School. *Front. Psychol*, 13, 887280. <https://doi.org/10.3389/fpsyg.2022.887280>
- Myers, B. (1990). Taxonomies of visual programming and program visualization. *J. Vis. Lang. Comput.*, 1, 97–123.
- OECD (2019). *PISA 2018 Results (Volume II): Where All Students Can Succeed*. PISA, OECD Publishing. <https://doi.org/10.1787/b5fd1b8f-en>
- Park, I., Kim, D., Oh, J., Jang, Y., & Lim, K. (2015). Learning Effects of Pedagogical Robots with Programming in Elementary School Environments in Korea. *Indian Journal of Science and Technology*, 8. <https://doi.org/10.17485/ijst/2015/v8i26/80723>
- Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas*. Basic Books, Inc.
- Piaget, J. (1970). *Science of education and the psychology of the child*. Orion Press.

- Piedade, J., Dorotea, D., Sampaio, F. F., & Pedro, A. A. (2019). A Cross-analysis of Block-based and Visual Programming Apps with Computer Science Student-Teachers. *Education Sciences*, 9(181). <https://doi.org/10.3390/educsci9030181>
- Rich, P. J., Browning, S. F., Perkins, M., Shoop, T., Yoshikawa, E., & Belikov, O. M. (2019). Coding in K-8: International trends in teaching elementary/primary computing. *TechTrends*, 63(3), 311–329. <https://doi.org/10.1007/s11528-018-0295-4>
- Sáez-López, J. M., Buceta Otero, R., & De Lara García-Cervigón, S. (2021). Introducing robotics and block programming in elementary education. *RIED. Revista Iberoamericana de Educación a Distancia*, 24(1), 95–113. <http://dx.doi.org/10.5944/ried.24.1.27649>
- Scherer, R., Siddiq, F., & Sánchez Viveros, B. (2019). The cognitive benefits of learning computer programming: A meta-analysis of transfer effects. *Journal of Educational Psychology*, 111(5), 764–792.
- Siegle, D. (2017). Technology: Encouraging Creativity and Problem Solving Through Coding. *Gifted Child Today*, 40(2), 117–123. <https://doi.org/10.1177/1076217517690861>
- Smahel, D., Machackova, H., Mascheroni, G., Dedkova, L., Staksrud, E., Ólafsson, K., Livingstone, S., & Hasebrink, U. (2020). *EU Kids Online 2020: Survey results from 19 countries*. EU Kids Online, The London School of Economics and Political Science. <https://doi.org/10.21953/lse.47fdeqj01ofo>
- Sun, L., & Zhou, D. (2022). Effective instruction conditions for educational robotics to develop programming ability of K-12 students: A meta-analysis. *Journal Computer Assisted Learning*, 39, 380–398.
- Sun, L., Hu, L., & Zhou, D. (2022). Programming attitudes predict computational thinking: Analysis of differences in gender and programming experience. *Computers & Education*, 181, 104457. <https://doi.org/10.1016/j.compedu.2022.104457>
- Tengler, K., Sabitzer, B., & Kastner-Hauler, O. (2020). Programming in primary schools – challenges and opportunities. *ICERI2020 Proceedings*, 7556–7563. <https://doi.org/10.21125/iceri.2020.1640>
- Vilig, K. (2013). *Kvalitativna istraživanja u psihologiji*. Clio.
- Vuorikari, R., Kluzer, S., & Punie, Y. (2022). *DigComp 2.2: The Digital Competence Framework for Citizens – With new examples of knowledge, skills and attitudes*. Publications Office of the European Union. <https://doi.org/10.2760/115376>
- Wei X., Lin L., Meng N., Tan W., & Kong, S. C. (2020). The effectiveness of partial pair programming on elementary school students’ computational Thinking skills and self-efficacy. *Computers & Education*, 160, 104023. <https://doi.org/10.1016/j.compedu.2020.104023>
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>

Summary

Programming is a digital competence necessary for life in the 21st century. In recent decades, education systems around the world have redefined curricula to include programming as a stand-alone subject or integrated into other subjects, from an early age. In addition to formal education, programming is also taught in non-formal educational contexts (NEC) and the use of educational robots is becoming more common. The aim of this paper is to familiarize with the advantages and challenges of using Sphero robots (BB-8 and SPRK+) in teaching visual programming in the NEC and to formulate practical implications. The research was conducted as a part of the Learning for the 21st Century project in 17 cities in Serbia. The quantitative research included 677 students ($M = 12.16$; $SD = 1.10$), and the qualitative 42 students and 5 workshop leaders. Descriptive quantitative and qualitative, thematic analysis was applied. The results show that using robots has additional educational and motivational value in the process of teaching visual programming and algorithmic thinking. From the point of view of the students and workshop leaders, the NEC has a number of advantages compared to traditional teaching of programming: learning through play, experiential learning, creativity and initiative of participants, insight that programming can be interesting, even though it is “hard”. Challenges are formulated in the paper, as well as practical recommendations for teaching practice.

Keywords: *educational robots, visual programming, algorithmic thinking, primary school age, non-formal educational context*