

SCRATCH KOT ORODJE ZA UTRJEVANJE MATEMATIČNIH VSEBIN IN RAZVOJ ALGORITMIČNEGA RAZMIŠLJANJA V 4. RAZREDU

¹Alenka Avguštin

¹Osnovna šola Belokranjskega odreda Semič, Slovenija

Povzetek

Prispevek obravnava, kako lahko uporaba programskega jezika Scratch pri pouku matematike prispeva k razvoju algoritmičnega razmišljanja in hkrati utrjevanju matematičnih vsebin pri učencih 4. razreda. Naš cilj je opisati način dela v razredu, kjer so učenci s pomočjo Scratcha ustvarili matematični kviz za utrjevanje zaokroževanja števil ter pisnega seštevanja in odštevanja do 10 000. S preizkušanjem kvizov so utrjevali znanje, si izmenjevali povratne informacije ter na podlagi tega izboljšali svoje izdelke. Uspešnost dejavnosti smo pridobivali z metodo študije primera, kjer smo zbirali podatke z opazovanjem, spraševanjem in analizo učenčevih izdelkov. Ključne ugotovitve so pokazale, da so učenci z veseljem ustvarjali kvize, razvijali tako matematična kot računalniška znanja ter aktivno medvrstniško sodelovali pri izboljševanju svojih izdelkov. Vsi učenci so bili motivirani, kar je nakazovalo na pozitiven učinek vključevanja programiranja v pouk. V prihodnosti bi bila potrebna dodatna podpora pri vključevanju animacij ter možnost daljšega časa za izvedbo nalog.

SCRATCH AS A TOOL FOR CONSOLIDATING MATHEMATICAL TOPICS AND DEVELOPING ALGORITHMIC THINKING IN GRADE 4

Abstract

The paper explores how using the Scratch programming language in mathematics lessons contributes to the development of algorithmic thinking while simultaneously consolidating mathematical concepts in Grade 4 pupils. We aim to describe the classroom approach, where pupils used Scratch to create a mathematics quiz to practice rounding numbers and written addition and subtraction up to 10,000. By taking the quizzes, they consolidated their knowledge, exchanged feedback, and improved their projects as a result. The success of the activity was

Ključne besede:

algoritmično mišljenje,
matematika, Scratch,
medpredmetno
povezovanje,
medvrstniško učenje

Keywords:

algorithmic
thinking, mathematics,
Scratch, cross-curricular
integration, peer learning

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/>)



measured using a case study method, where data was collected via observation, questioning, and analysis of pupils' work. Key findings indicate that pupils enjoyed creating quizzes, developing mathematical and computational skills, and were actively engaged in peer collaboration to improve their projects. All pupils were motivated, which indicated a positive effect of integrating programming into the classroom. Future improvements could include additional support for incorporating animations and providing more time for task completion.

1 UVOD

Hitri digitalni razvoj vpliva na skoraj vse vidike našega življenja, zato se šolski sistem nenehno prilagaja novim izzivom in smernicam. Pri tem pa ni več vprašanje, ali digitalne tehnologije sodijo v učni prostor, temveč kako jih najučinkoviteje vključiti v poučevanje (Kreuh idr., 2023).

Da bi se pripravili na izzive prihodnosti, se pripravlja prenova učnih načrtov za šole. V vmesnem času pa v šoli s pomočjo projektov in izobraževanj postopno vključujemo v pouk aktivnosti, ki razvijajo digitalne kompetence in temeljna znanja računalništva in informatike. Slednje predstavljajo temelj za ustvarjalno in polno preživetje posameznika v 21. stoletju (RINOS, 2022). Osnovna veščina, ki naj bi jo učenci pridobili pri pouku računalništva in informatike, je zmožnost računalniškega mišljenja. Le-ta zajema različna področja znanj in spretnosti. Eno izmed njih je tudi programiranje.

S programiranjem se učenci učijo algoritemskega mišljenja, ki predstavlja strategijo za reševanje problemov in logično razmišljanje. Programiranje učencem omogoča, da razmišljajo o svojem razmišljanju. Pri iskanju napak v programu pa se učijo prepoznavati napake v lastnem razmišljanju (RINOS, 2018).

Pri načrtovanju pouka je priporočeno, da se digitalna tehnologija uporablja pri aktivnih oblikah in metodah dela (Dolinar idr., 2023). Skozi sodelovalno učenje se učenci učijo drug od drugega. Deljenje idej, izkušenj in znanja spodbuja medsebojno učenje ter omogoča pridobivanje različnih perspektiv in rešitev (Didakt.UM, 2020). Tako medvrstniško učenje prispeva k boljši kakovosti učenja, saj si lahko učenci medsebojno pomagajo in na ta način prihajajo do znanja, ki je trajnejše.

Ker smo želeli pri učencih razvijati algoritmično mišljenje, smo iskali načine, kako programiranje smiselno vključiti v pouk. Pojavilo se je vprašanje, kje in kdaj ga

medpredmetno povezati z učno snovjo, da bi učenci ne le utrjevali obravnavane učne vsebine, temveč tudi razvijali temeljna znanja računalništva in informatike. Vemo, da veliko večšin reševanja problemov učenci že pridobijo pri matematiki, vendar pogosto ne znajo miselno razčleniti postopkov reševanja, kar je ključno za razvoj algoritmičnega mišljenja. Zato smo se odločili, da programiranje smiselno povežemo z matematiko. Tako bodo učenci utrjevali matematična znanja ter hkrati razvijali veščine za reševanje problemov.

V prispevku bomo predstavili način povezovanja matematike in programiranja v 4. razredu z uporabo programskega jezika Scratch. Namen prispevka je predstaviti način, kako lahko uporaba vizualnega orodja Scratch pri pouku matematike prispeva k razvoju temeljnih znanj računalništva in hkrati utrjevanju matematičnih vsebin pri učencih 4. razreda. Namenjen je učiteljem 2. triletja osnovne šole kot dodatna ideja za izvedbo pouka. Naš cilj je opisati način dela v razredu, kjer so učenci s pomočjo programiranja ustvarili lastno učno gradivo za utrjevanje matematičnih znanj. S preizkušanjem sošolčevih kvizov so nato aktivno utrjevali matematično znanje in si medvrstniško podajali povratno informacijo. Tako so lahko svoje učno gradivo sproti prilagajali in izboljševali.

2 ZASNOVE ŠTUDIJE PRIMERA IN METODOLOGIJA

Eden od načinov, kako lahko programiranje smiselno medpredmetno povežemo z matematiko, je vključevanje vizualnega programskega jezika. Z razvojem tehnologije in željo po učenju programiranja se je razvilo veliko programskih jezikov. Eden izmed njih je vizualni programski jezik Scratch, ki je prosto dostopen. Preveden je v veliko različnih jezikov, med njimi tudi v slovenščino in je tudi zato tako priljubljen. Ustvarjen je bil za učenje programiranja za otroke od 8. leta starosti naprej. Učencem omogoča učenje programiranja s sestavljanjem blokov in možnost takojšnjega vpogleda v trenutni izdelek. Bloke lahko učenec premika in razstavlja, dokler ni zadovoljen s končnim izdelkom. Učenci lahko z njim programirajo interaktivne zgodbe, igre in animacije. Svoje izdelke lahko delijo z drugimi ter tudi preizkušajo izdelke drugih (Lajovic, 2011). Scratch je primeren za učence 2. triletja, ker je vizualno podprt z barvami in oblikami ter tako olajša razumevanje. S preizkušanjem dobimo hitro povratno informacijo o delovanju, kar omogoča takojšnje izboljšave. Izbran programski jezik tako ponuja priložnost za medpredmetno povezovanje z matematiko, kar smo preizkusili v konkretni učni situaciji.

Učenci 4. razreda so pri matematiki v programskem jeziku Scratch izdelali kviz za utrjevanje matematičnih znanj iz področja aritmetike in algebre pri sklopu naravnih števil

ter računskih operacij in njihovih lastnosti. Utrjevali so zaokroževanje števil na desetine in stotice ter pisno seštevanje in odštevanje naravnih števil do 10 000. Te vsebine so bile izbrane, saj gre za proceduralni tip matematičnega znanja, ki se lepo poveže z algoritemskim načinom razmišljanja. Tako smo cilje pri matematiki povezali s temeljnimi znanji računalništva in informatike iz sklopa algoritmov in programiranja. Dejavnost je potekala v treh šolskih urah.

V uvodnem delu so učenci najprej preizkusili že pripravljen kviz s poštevanko v Scratchu. V njem je potapljač postavljaj vprašanja o poštevanki, ob pravilnem odgovoru pa je veselo zaplaval proti ribi. Učenci so sprotno spremljali, kako se potapljač odziva na njihove odgovore. Ko so vsi zaključili kviz, smo se pogovorili o njihovi uspešnosti in analizirali, kako je bil kviz sestavljen. Čeprav poštevanka ni bilo nujno predznanje za nadaljnje delo, smo jo vključili v uvodni del, ker je njeno redno utrjevanje ključno za dolgoročno pomnjenje temeljnih matematičnih dejstev. Poleg so učenci na preprostem primeru spoznali način delovanja kviza, pri čemer so zaznali pomembnost takojšnje povratne informacije.

Po uvodni motivaciji so učenci dobili izziv, da ustvarijo svoj kviz v Scratchu. Njihova naloga je bila ustvariti program, ki bo uporabnikom zastavljal vprašanja o zaokroževanju števil na desetine in stotice ter o pisnem seštevanju in odštevanju naravnih števil do 10 000. Ob vsakem odgovoru je moral program podati takojšnjo povratno informacijo o pravilnosti rešitve. Da bi kvizi postali še bolj zanimivi, so morali učenci vključiti tudi animacijo ali gibanje figur, ki so jih lahko zasnovali po lastnih zamislih.

Pred začetkom ustvarjanja smo skupaj določili kriterije uspešnosti: kviz vsebuje 5 matematičnih vprašanj iz opisanih tem; vsebuje sprotno povratno informacijo; dodana je animacija ali gibanje figure ob pravilnem odgovoru; kviz je shranjen in deljen s sošolci ter učiteljem; vsak učenec preizkusi vsaj en kviz svojega sošolca in mu poda povratno informacijo.

Učenci so najprej premislili, katera vprašanja bi bila najboljša za utrjevanje dogovorjenih matematičnih znanj, nato pa se lotili sestavljanja blokovnih ukazov in sestavljanja svojega kviza. Med ustvarjanjem so nenehno preizkušali delovanje programa, popravljali morebitne napake in izboljševali svoje rešitve. Po želji so kviz obogatili z dodatnimi animacijami, novimi liki, zankami ali dodatnimi vprašanji. Učitelj jih je sprotno spremljal in jim po potrebi pomagal.

V zaključnem delu so učenci svoje znanje utrdili tako, da so preizkusili kvize svojih sošolcev. Pri tem so pozorno spremljali morebitne pomanjkljivosti ali napake v delovanju ter drug drugemu podali povratno informacijo. Sledili so popravki lastnega kviza in samoevalvacija s pomočjo vprašalnika.

Učinkovitost dejavnosti smo pridobivali z metodo študije primera, kjer smo zbirali podatke z opazovanjem, spraševanjem in analizo učenčevih izdelkov. V izvedeno aktivnost je bilo vključenih 15 učencev 4. razreda osnovne šole. Vsi koraki dejavnosti so bili izvedeni v računalniški učilnici, vsak učenec za svojim računalnikom. Ker učenci prvotno niso poznali izbranega programskega okolja, smo predhodno izvedli 5 šolskih ur spoznavanja in preizkušanja le-tega. Naučili so se vstaviti ozadje in figure, animirati figure, se seznanili z operatorji in pogojnimi stavki ter ustvarili enostavne algoritme. Tudi učitelji smo se predhodno izobrazili in preizkusili zmožnosti izbranega programskega jezika, da smo lahko zagotovili ustrezno podporo učencem. Za izdelavo učenčevih računov v Scratchu smo aktivirali učiteljski račun, ki omogoča enostavno ustvarjanje računov za učence in hiter pregled njihovih izdelkov.

3 REZULTATI

Učna dejavnost je potekala uspešno po načrtovanih korakih. Med izvajanjem dejavnosti smo opazili veliko zavzetost in motiviranost učencev. Učenci so pokazali dobro razumevanje ukazov in zaporedij. Večinoma so nalogo opravili samostojno, trije učenci pa so potrebovali nekaj vodenja oziroma pomoči učitelja. Nekateri učenci so potrebovali več časa za oblikovanje vprašanj in sestavljanje blokovnih ukazov, kar je povzročilo različen tempo napredovanja. Učenci so med preizkušanjem kvizov svojih sošolcev aktivno sodelovali pri iskanju napak glede na kriterije ter bili pozorni tudi na matematično pravilnost izračuna. Sprotno so si med seboj predlagali izboljšave in jih nato tudi popravili. Tako so tudi samoevalvirali svoje delo in znanje.

Tabela 1: Doseganje kriterijev uspešnosti

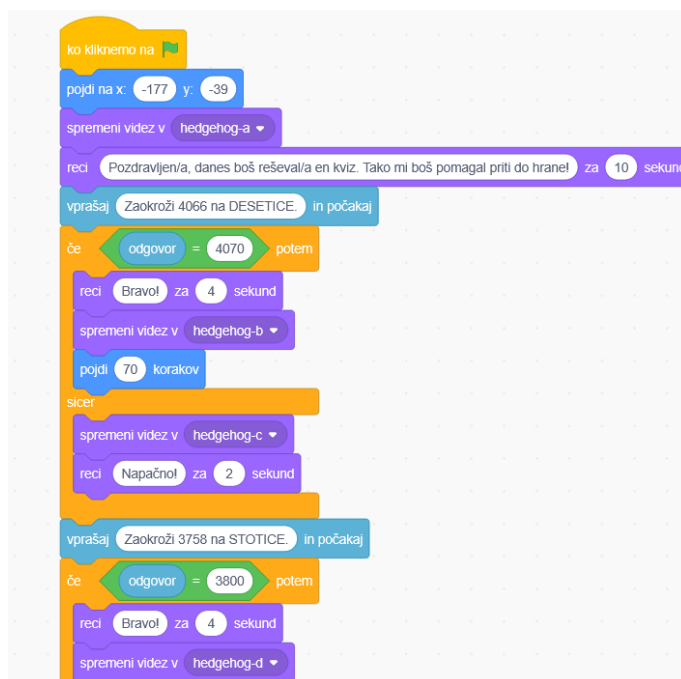
Kriterij uspešnosti	Število učencev, ki so kriterij dosegli
Kviz vsebuje 5 vprašanj o zaokroževanju ter seštevanju in odštevanju št.	14
Vključena je povratna informacija	15
Dodana je animacija ob pravilnem odgovoru	12
Kviz je shranjen in deljen	15
Učenec je preizkusil kviz sošolca	15

Tabela 1 prikazuje analizo učencevih izdelkov po kriterijih uspešnosti. Iz tabele je razvidno, da so učenci v veliki meri uspešno dosegli zastavljene kriterije. Le eden izmed učencev ni imel vseh 5 vprašanj, saj je med urejanjem po nesreči izbrisal zaporedje ukazov in moral program napisati na novo. Zaradi časovne omejitve mu je zmanjkalo časa še za 2 vprašanja. Večina učencev je znala vključiti animacijo oziroma gibanje figure, trije učenci pa so imeli težavo pri tem, kar kaže na potrebo po dodatni podpori pri tej veščini. Kljub temu so vsi učenci v svoje kvize vključili sprotno povratno informacijo, kot prikazuje slika 1, kar je ključno za utrjevanje matematičnih znanj. Po izdelavi so učenci znali samostojno shraniti in deliti svoj kviz. Nekaj učencev je svoj kviz dodatno obogatila z zgodbo, dodatnimi animacijami in gibanji likov, kot prikazujeta slika 1 in slika 2. Pri analizi kvizov smo ugotovili, kateri učenci imajo še primanjkljaje pri matematičnih vsebinah kot tudi v algoritmičnem mišljenju.



Slika 1: Posnetek zaslona učenčevega kviza

Na posnetku zaslona je prikazan primer kviza, ki ga je ustvaril učenec v skladu s predstavljenimi kriteriji in vključuje vprašanja iz dogovorjenega matematičnega področja. Po vpisu odgovora sledi povratna informacija. Ob pravilnem odgovoru, se figura premakne naprej oziroma kako drugače spremeni.



Slika 2: Posnetek zaslona kode kviza

Prikazan je tudi posnetek zaslona kode kviza, ki je bila uporabljena za ustvarjanje kviza v Scratchu. Učenec je pri vsakem vprašanju vključil vhodno-izhodni ukaz in pogojni stavek, ki je podal povratno informacijo o pravilnosti rešitve. Dodal je tudi animacijo in gibanje figure.

Večina učencev je pri oblikovanju svojih kvizov vključila štirimestna števila za zaokroževanje števil ter raznolike račune seštevanja in odštevanja s štirimestnimi števili. Pri vsaj enem računu je večina vključila en ali dva prehoda, eden učenec pa tudi več. Štirje učenci so oblikovali račune, v katerih so kombinirali števila z različnim številom mest, na primer štirimestno in trimestno število. Po drugi strani pa so trije učenci izbrali enostavnejše račune s trimestnimi števili, pri čemer sta dva učenca zapisala račune brez prehoda. Ugotovili smo, da so učenci izbirali različno težavnost nalog, pri čemer so nekateri vključili bolj zapletene primere, drugi pa so se osredotočili na bolj enostavne naloge.

Po končani aktivnosti so učenci odgovorili na anketni vprašalnik. Rezultati so pokazali, da je velika večina učencev Scratch prepoznala kot koristen in zabaven pripomoček za učenje matematike. Vsem učencem je bilo všeč ustvarjanje lastnih kvizov in utrjevanje matematike

na interaktiven način, kar je pripomoglo k boljšemu razumevanju vsebine. Bili so navdušeni nad preizkušanjem sošolčevih kvizov in nad vrstniško povratno informacijo. Vsi učenci so dejavnost ocenili kot zanimivo in zabavno. Večina učencev je izrazila željo, da bi programiranje v Scratchu še uporabili pri matematiki. Nekateri pa so predlagali vključevanje tudi pri drugih predmetih, na primer pri slovenščini ali naravoslovju.

4 ZAKLJUČEK

V prispevku smo predstavili način povezovanja matematike in programiranja v 4. razredu z uporabo programskega jezika Scratch. Naše ugotovitve so usklajene z ugotovitvami Jeglič (2017), ki je v svoji študiji primera ugotovila, da učenci radi uporabljajo Scratch in da jim je bilo ustvarjanje v obliki kviza všeč. Tudi naši učenci so aktivnosti v Scratchu zaznali kot zabavne in koristne za utrjevanje matematike. Učenci so z veseljem pristopili k izdelavi matematičnih kvizov, pri čemer so razvijali tako matematična kot računalniška znanja. Z aktivnostjo so poglobili znanje zaokroževanja števil na desetice in stotice, pisnega seštevanja in odštevanja do 10 000 ter hkrati urili algoritmični način razmišljanja. Naši rezultati in rezultati Jeglič (2017) kažejo pozitivne učinke medpredmetnega povezovanja pri učenju, kar podpira smiselnost uporabe programskega jezika Scratch v učnem procesu. Dejavnost se je izkazala za uspešno, saj so učenci samostojno izdelali kvize, jih medsebojno preizkušali in si med seboj podali povratne informacije, kar je pomembno za poglobljeno učenje. Učenci so bili motivirani za tovrstno učenje, saj so ga dojemali kot poučno igro, kar je pripomoglo k boljšemu razumevanju matematičnih vsebin.

Pri izvedbi dejavnosti so se pojavili tudi izzivi. Nekateri učenci so imeli težave pri vključevanju animacij in gibanja figur, kar kaže na potrebo po dodatni podpori pri teh elementih v Scratchu. Prav tako je bilo opaziti razlike v tempu napredovanja učencev, kar bi lahko izboljšali z dodatnim časom. Dejavnost bi lahko pri naslednji uri matematike tudi nadgradili z vključevanjem spremenljivk. V prihodnje bi lahko razmislili, kako bi lahko programiranje medpredmetno povezali še z drugimi predmeti, npr. s slovenščino ali naravoslovjem.

Pri medpredmetnem povezovanju matematike in računalništva je ključno, da so cilji obeh predmetov jasno opredeljeni in da je dejavnost skrbno načrtovana. Pri tem je tudi pomembno premišljeno vključevanje digitalne tehnologije – v pravi meri in le takrat, ko dejansko prispeva k obogatitvi pouka in nadgraditvi znanja. Izbrati je potrebno tudi ustrezne didaktične metode, ki omogočajo sodelovanje učencev, spodbujajo raziskovanje in ustvarjalnost ter tako prispevajo k boljšemu razumevanju in povezovanju znanj.

LITERATURA

- Dolinar, M., Poberžnik, A. in Jerše, L. (2023). *Vodenje in podpora učencem pri pridobivanju digitalnih kompetenc: 1. del: Strokovna izhodišča in priporočila*. Zavod RS za šolstvo. https://www.zrss.si/pdf/vodenje_in_podpora_ucencem_pri_pridobivanju_digitalnih_kompetenc.pdf
- Jeglič, A. (2017). *Računanje v Scratchu* [Diplomsko delo, Univerza v Ljubljani, Pedagoška fakulteta]. Repozitorij UL. http://pefprints.pef.uni-lj.si/4772/1/diploma_anja_jeglic.pdf
- Kreuh, N., Pesek, I., Košutnik, R., Čuk, A., & Možina Podbršček, I. (2023). *Projekt Dvig digitalne kompetentnosti se predstavi*. Zavod RS za šolstvo. https://www.zrss.si/pdf/Projekt_Dvig_digitalne_kompetentnosti_se_predstavi.pdf
- Lajovic, S. (2011) Scratch: Nauči se programirati in postani računalniški maček. Pasadena
- Ministrstvo za izobraževanje, znanost in šport. (2022). *Okvir računalništva in informatike od vrtca do srednje šole: Poročilo strokovne delovne skupine za analizo prisotnosti vsebin računalništva in informatike v programih osnovnih in srednjih šol ter za pripravo študije o možnih spremembah (RINOS)*. https://redmine.lusy.fri.uni-lj.si/attachments/download/3060/Porocilo_RINOS_10_1_22.pdf
- Ministrstvo za izobraževanje, znanost in šport. (2018). *Snovalci digitalne prihodnosti ali le uporabniki? Poročilo strokovne delovne skupine za analizo prisotnosti vsebin računalništva in informatike v programih osnovnih in srednjih šol ter za pripravo študije o možnih spremembah (RINOS)*. https://redmine.lusy.fri.uni-lj.si/attachments/download/2786/Porocilo_RINOS_30_5_18.pdf
- Projekt Didakt.UM. (2020). *Strokovna podlaga za sodelovalno učenje*. Univerza v Mariboru. <https://didakt.um.si/oprojektu/projektneaktivnosti/Documents/Strokovna-podlaga-sodelovalno-ucenje.pdf>