

# Prednosti i nedostaci uvođenja programiranja u nastavi informatike primarnog obrazovanja

---

Hlavsa, Dominik

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Teacher Education / Sveučilište u Zagrebu, Učiteljski fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:147:473099>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-12-01**

Repository / Repozitorij:

[University of Zagreb Faculty of Teacher Education - Digital repository](#)



**SVEUČILIŠTE U ZAGREBU  
UČITELJSKI FAKULTET  
ODSJEK ZA UČITELJSKE STUDIJE**

**Dominik Hlavsá**

**PREDNOSTI I NEDOSTACI UVOĐENJA  
PROGRAMIRANJA U NASTAVU INFORMATIKE  
PRIMARNOG OBRAZOVANJA**

**Diplomski rad**

**Zagreb, srpanj 2023.**

**SVEUČILIŠTE U ZAGREBU  
UČITELJSKI FAKULTET  
ODSJEK ZA UČITELJSKE STUDIJE**

**Dominik Hlavsa**

**PREDNOSTI I NEDOSTACI UVOĐENJA  
PROGRAMIRANJA U NASTAVU INFORMATIKE  
PRIMARNOG OBRAZOVANJA**

**Diplomski rad**

**Mentor rada:  
Mario Dumančić, prof. dr. sc.**

**Zagreb, srpanj 2023.**

## **Zahvala**

*Zahvaljujem svojem mentoru prof. dr. sc. Mariju Dumančiću na susretljivosti, stručnoj pomoći i savjetima pri izradi ovog rada.*

*Posebno se zahvaljujem svojoj obitelji, ponajviše roditeljima, na njihovom razumijevanju i strpljenju, što su mi omogućili željeno obrazovanje i bili neprestana potpora na tom putu. Zahvaljujem svojoj djevojci na uzajamnom učenju i podršci te zahvaljujem svim kolegama, kolegicama i prijateljima na njihovoj potpori tijekom studiranja.*

## SAŽETAK

Informatika i njezini sadržaji učenja prilagođavaju se neprestanim promjenama koje se događaju u tehnološkom svijetu. Sastavljanjem kurikuluma 2018. godine, Ministarstvo znanosti i obrazovanja je napokon utkalo pravi put informatici, a važnije – programiranju, u primarno obrazovanje. Kurikulum je optimalno podijeljen na četiri domene, od kojih svaka zaokružuje po jedno područje informatike u obrazovanju. Veliko značenje dano je programiranju u nastavi. Programiranje je u današnje vrijeme neophodna vještina koju treba usvojiti, ali i cijeli život njegovati, a upravo je primarno obrazovanje odličan početak upoznavanja s tim. Uvođenjem programskog jezika Scratch, učitelji i učenici zapravo kroz igru nižu linije kodova i programiraju. U Europi i svijetu zadnjih se godina javljaju projekti o programiranju koja svaka škola može prijaviti i okušati se te na kraju svoje rezultate podijeliti s ostalim školama iz drugih država. Neki od takvih projekata su *Hour of Code* i *Code Week*. U hrvatskim školama se programira u Scratchu nakon kojeg se prelazi na Python koji je viši programski jezik, kompliciraniji, ali s razumljivom sintaksom.

Uvođenje programiranja je s dosta poteškoća pronašlo put u hrvatsko primarno obrazovanje, ali je kvalitetno i metodički ispravno opisano. Provedba sadržaja iz programiranja uvelike ovisi o učitelju informatike, njegovoj sposobnosti i edukaciji. Usporedivši uvođenje programiranja u RH s uvođenjem programiranja u stranim zemljama, uočava se kako Hrvatska ne zaostaje toliko za drugim, razvijenijim zemljama. Štoviše, programiranje opisano u hrvatskom kurikulumu kvalitetnije je provedeno nego u drugim zemljama. Iako Amerikanci slove kao možda najmoćniji i najrazvijeniji narod svijeta, nastava programiranja je vrlo siromašna u njihovim osnovnim školama. Uz Estoniju, Engleska se može pohvaliti da je među prvim zemljama koja je već od prvog razreda OŠ uvela programiranje u redovnu nastavu. Druga europska zemlja – Španjolska, zadnjih godina uvodi programiranje kao izvannastavnu aktivnost, iako postoje naznake uvođenja u redovan plan i program. Veliki problem u svim državama predstavlja učiteljski kadar koji prema svim znakovima nije dovoljno obrazovan da bi znanje i sposobnosti o programiranju prenijelo na svoje učenike. Iz godine u godinu većina država ozbiljnije gleda prema programiranju kao neizostavnoj vještini koja se zbog svoje kompleksnosti, ali i važnosti, mora uvesti već od samih početaka školovanja, a to je primarno obrazovanje.

ključne riječi: informatika, kurikulum, programiranje, programski jezici

## **SUMMARY**

Computer Science as a school subject change from year to year, that is, its learning contents adapt to the constant changes happening in the technological world. By formulating the curriculum in 2018, the Ministry of Science and Education has finally paved the right path for Computer Science, and more importantly, programming, in primary education. The curriculum is optimally divided into four domains, each encompassing a specific area of computer science in education. Programming has been given significant importance in teaching. Programming is a necessary skill to acquire in today's world, one that should be nurtured throughout life, and primary education serves as an excellent starting point for introduction to it. Through the introduction of the Scratch programming language, teachers and students actually write together lines of code and program through play. In recent years, programming projects have emerged in Europe and the world, which every school can participate in, test their skills, and share their results with other schools from different countries. Some of these projects include Hour of Code and Code Week. In Croatian schools, programming is initially taught using Scratch, followed by Python, which is a higher-level programming language, more complex, but with a comprehensible syntax.

The introduction of programming has encountered several challenges in Croatian primary education, but it has been described in a quality and methodically correct manner. The implementation of programming content largely depends on the computer science teacher, their abilities and education. When comparing the introduction of programming in Croatia to other countries, it is observed that Croatia is not lagging behind as much as other more developed countries. Furthermore, programming described in the Croatian curriculum is implemented with higher quality than in other countries. Although Americans are considered perhaps the most powerful and developed nation in the world, programming education in their elementary schools is very poor. Alongside Estonia, England can boast of being among the first countries to introduce programming into regular curriculum starting from the first grade of primary school. Another European country, Spain, has been introducing programming as an extracurricular activity in recent years, although there are indications of its potential inclusion in the regular curriculum. A significant problem in all countries is the teaching staff, who, according to all signs, are not sufficiently educated to pass on their knowledge and programming skills to their students. Year after year, most countries are taking programming more seriously as an essential skill that needs

to be introduced from the very beginning of education, namely primary education, due to its complexity and importance.

Keywords: computer science, curriculum, programming, programming languages

## SADRŽAJ

|                                                                                                                 |    |
|-----------------------------------------------------------------------------------------------------------------|----|
| SAŽETAK .....                                                                                                   | 5  |
| SUMMARY .....                                                                                                   | 6  |
| 1. UVOD.....                                                                                                    | 1  |
| 2. INFORMATIKA – NASTAVNI PREDMET .....                                                                         | 1  |
| 2.1. Informatika u osnovnoj školi .....                                                                         | 2  |
| 2.2. Kurikulum predmeta Informatika.....                                                                        | 2  |
| 2.3. Izborna nastava.....                                                                                       | 5  |
| 3. INFORMATIČKA PISMENOST .....                                                                                 | 6  |
| 4. PISMENOST PROGRAMIRANJA .....                                                                                | 7  |
| 5. PROJEKTI ZA POTICANJE PROGRAMIRANJA .....                                                                    | 8  |
| 5.1. Hour of Code.....                                                                                          | 8  |
| 5.2. Code Week .....                                                                                            | 10 |
| 5.3. Run Marco.....                                                                                             | 11 |
| 6. PROGRAMSKI JEZICI .....                                                                                      | 12 |
| 6.1. Programski jezici namijenjeni učenju .....                                                                 | 13 |
| 6.2. Scratch.....                                                                                               | 13 |
| 6.3. Python.....                                                                                                | 15 |
| 7. UVOĐENJE PROGRAMIRANJA U NASTAVU INFORMATIKE PRIMARNOG<br>OBRAZOVANJA U REPUBLICI HRVATSKOJ .....            | 16 |
| 8. UVOĐENJE PROGRAMIRANJA U NASTAVU INFORMATIKE PRIMARNOG<br>OBRAZOVANJA U SJEDINJENIM AMERIČKIM DRŽAVAMA ..... | 18 |
| 9. UVOĐENJE PROGRAMIRANJA U NASTAVU INFORMATIKE PRIMARNOG<br>OBRAZOVANJA U ENGLESKOJ .....                      | 21 |



|                                                                                               |    |
|-----------------------------------------------------------------------------------------------|----|
| 10. UVOĐENJE PROGRAMIRANJA U NASTAVU INFORMATIKE PRIMARNOG<br>OBRAZOVANJA U ŠPANJOLSKOJ ..... | 23 |
| ZAKLJUČAK.....                                                                                | 25 |
| LITERATURA .....                                                                              | 26 |
| POPIS SLIKA.....                                                                              | 28 |
| Izjava o samostalnoj izradi rada.....                                                         | 29 |

## **1. UVOD**

U današnjem vremenu neprestanog i ubrzanog razvitka jedan od ključnih faktora je tehnologija. Kako u industriji i gospodarstvu, tako i u školstvu. Predmet Informatika dobiva sve više na važnosti u školi pa se shodno s time proširuju nastavni sadržaji iz nastave Informatike i nadovezuju s ostalim predmetima pomoću informacijske i komunikacijske tehnologije (IKT). Informatika je napokon pronašla put do samog početka školovanja pa učenici već od prvog razreda osnovne škole mogu pohađati nastavu Informatike, no, treba naglasiti da je ta nastavna i dalje izborna te ovisi o roditeljskoj ili učenikovojoj odluci o pohađanju tog predmeta. Kurikulum predmeta Informatika se kao i sama informatika i tehnologija nadograđuje i proširuje, tj. ishodi unutar domena se nadovezuju ili mijenjaju jer je ipak jedna od odlika informatike otvorenost prema novim saznanjima i dostignućima koji se događaju svakodnevno.

Dolaskom djece u osnovnu školu, treba pronaći pravi način kako ih zainteresirati za programiranje. Jedan od načina je korištenje zabavnih i pristupačnih programskih jezika kao što je Scratch. Nadalje, djelatnici škole (ravnatelji, učitelji informatike) trebaju biti dovoljno obrazovani, željni i motivirani kako bi svoje stečeno znanje prenijeli učenicima te zajedno s njima i oni sami učili nove stvari s obzirom da se informatika mijenja i ažurira gotovo danomice.

U sljedećim poglavljima ovog rada riječ će biti o programiranju, implementaciji programiranja od 1. razreda osnovne škole te od prednostima i manama programiranja u nastavi Informatike u kurikulumu i praksi u Republici Hrvatskoj te ostalim državama. Rad se usredotočuje na domenu B. u kurikulumu - Računalno razmišljanje i programiranje. Govori se o informatici kao nastavnom predmetu, programskim jezicima koji služe u edukacijske svrhe (Scratch i Python) te o informatičkoj pismenosti učenika i njihovom predznanju s kojim dolaze u osnovnu školu.

## **2. INFORMATIKA – NASTAVNI PREDMET**

Gugić i sur. (1997) navode kako za informatiku nema jedinstvene definicije koja će je opisati, već je ona jednostavno mlada znanost čije su glavne odlike informacije – obavijesti. S obzirom da se čovječanstvo nalazi u modernim vremenima neprestanog razvitka i širenja tehnologije, neobično je da je informatika kao nastavni predmet kasno ušla u škole. Treba naglasiti da je tek u školskoj godini 2020./2021. Ministarstvo znanosti i obrazovanja uvrstilo je informatiku kao izborni predmet od prvog razreda osnovne škole. Kurikularnom reformom informatika je uvedena kao obvezan

predmet u višim razredima (5. – 8.). Godišnja satnica u svim razredima osnovne škole je 70 sati, što je u nekim razredima premalo zbog širine i kompleksnosti nastavnih sadržaja. Isto se tako može i reći da je nastavni sadržaj u nižim razredima prekratak i jednostavan tako da učiteljima informatike ostane i dovoljno vremena za neke dodatne sadržaje koji su zanimljiviji od teorije i upisivanja odgovora u udžbenike i radne bilježnice. Napuštanjem Nastavnog plana i programa tehničke kulture i informatike, koji je izašao 1999. godine, i sastavljanjem Kurikuluma za nastavni predmet Informatika 2018. godine, Informatika postaje neizostavan predmet i dio školstva, te važnije, doživljava ogromnu kurikularnu reformu sadržaja koji su u toku s vremenom i promjenama.

### *2.1. Informatika u osnovnoj školi*

Tehnološki razvitak i informatizacija se u Hrvatskoj događala brzo, s iznimkom Informatike kao predmet u školama. U prošlom stoljeću te početkom ovoga, Informatika se spominjala i ukratko poučavala u sklopu predmeta Tehnička kultura: „Edukacija u osnovnim školama iz informatike je orijentirana na upoznavanje svojstava elektronskih računala, a u srednjim školama na primjenu računala u rješavanju jednostavnih problema vezanih za struku koja se uči.“ (Radošević, 1992, str. 171). Naglasak je bio na računalima i opremi (hardware), dok se o programima (software), programiranju i pogotovo robotici nije davala gotovo nikakva važnost. Početkom 21. stoljeća rijetko koje kućanstvo je posjedovalo računalo, a i oni koji su ga posjedovali, većinom nisu bili umreženi, tj. nisu imali pristup internetu. Neprestanom modernizacijom postalo je izvjesno da se programiranje mora uvesti u nastavni plan i program škole. U početku se programiralo u programskim jezicima PASCAL, BASIC i LOGO. Danas se u nižim razredima programira u Scratchu, koji je odličan prvi izbor jer je vrlo dinamičan i, važnije, vizualan programski jezik koji olakšava učenje i njegovo razumijevanje. U višim razredima programirati se počinje u 5. razredu u programskim jezicima Pythonu i nešto rjeđe u C# (čitaj C sharp). Gugić i sur. (1997) navode sljedeće ishode koje učenici prvi puta trebaju ostvariti u 5. razredu, a to su: programiranje, korake programiranja, algoritam. U srednjim školama informatika je bila zastupljenija, no ovaj će se rad fokusirati na informatiku u primarnom obrazovanju.

### *2.2. Kurikulum predmeta Informatika*

Izglasano 2018. godine, kurikulumom je Informatika doživjela pravi preporod. Bitno je naglasiti da je kurikulum za razrede 5. – 8. te za srednje škole stupio na snagu od školske godine 2018./2019.,

a za razrede 1. – 4. (primarno obrazovanje) od 2020./2021. Kurikulum je vrlo opširan i u njemu su navedeni i opisani odgojno-obrazovni ishodi te njihove razrade, prikazani su godišnji fondovi sati i elaborirani su oblici izvođenja nastave te je na kraju prikazan popis kvalifikacija za stručno osoblje – učitelje i nastavnike informatike. U kurikulumu su opisane četiri domene Informatike koje su jednake za sve razrede osnovne i srednje škole, dok su ishodi i razrade ishoda drugačiji na taj način da se svakim idućim razredom ishodi nadovezuju i proširuju.

Predmet je u kurikulumu podijeljen na četiri domene (Ministarstvo znanosti i obrazovanja, 2018):

- A. Informacije i digitalna tehnologija
- B. Računalno razmišljanje i programiranje
- C. Digitalna pismenost i komunikacija
- D. E-društvo

| 2. RAZRED                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Domena                                 | Ishod                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Informacije i digitalna tehnologija    | Nakon druge godine učenja predmeta Informatika u domeni Informacije i digitalna tehnologija učenik:<br>A.2.1 objašnjava ulogu programa u uporabi računala<br>A.2.2 uz pomoć učitelja prepoznaje internet kao izvor nekih usluga i podataka te pretražuje preporučene sadržaje.                                                                                                                                                                                                                          |
| Računalno razmišljanje i programiranje | Nakon druge godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik:<br>B.2.1 analizira niz uputa koje izvode jednostavan zadatak, ako je potrebno ispravlja pogrešan redoslijed<br>B.2.2 stvara niz uputa u kojemu upotrebljava ponavljanje.                                                                                                                                                                                                                         |
| Digitalna pismenost i komunikacija     | Nakon druge godine učenja predmeta Informatika u domeni Digitalna pismenost i komunikacija učenik:<br>C.2.1 prema savjetima učitelja odabire uređaj i program za jednostavne školske zadatke<br>C.2.2 izrađuje digitalne radove kombiniranjem različitih oblika sadržaja uz podršku učitelja<br>C.2.3 uz pomoć učitelja surađuje i komunicira s poznatim osobama u sigurnome digitalnom okruženju.                                                                                                      |
| e-Društvo                              | Nakon druge godine učenja predmeta Informatika u domeni e-Društvo učenik:<br>D.2.1 prepoznaje i opisuje neke poslove koji se koriste informacijskom i komunikacijskom tehnologijom<br>D.2.2 koristi se e-uslugama u području odgoja i obrazovanja<br>D.2.3 analizira neke opasnosti koje mogu nastupiti pri uporabi računala i interneta te pravilno na njih reagira<br>D.2.4 odgovorno se ponaša pri korištenju sadržajima i uslugama na internetu radi zaštite osobnih podataka i digitalnoga ugleda. |

*Slika 1: Odgojno-obrazovni ishodi prema ishodima i domenama za 2. razred OŠ*

U domeni A. naglasak je na multimediji – tekst, slika, zvuk, video te na temeljnom razumijevanju rada računala i uređaja (računalna periferija). Također, veliki značaj od prvog razreda daje se sigurnosti na internetu i prepoznavanju vjerodostojnosti i točnosti informacija.

„Postoje jasni nedostaci u dječjoj spoznaji o rizicima na mreži, i usprkos brzom povećanju korištenja interneta među djecom i adolescentima, mnogima nedostaju digitalne vještine i kritična sposobnost procjene sigurnosti i vjerodostojnosti sadržaja i odnosa koje doživljavaju na mreži. To

odražava potrebu za još rasprostranjenijom digitalnom pismenošću koja može zaštititi i osnažiti djecu.“ (UNICEF, 2017)

Učenici uz pomoć učitelja prepoznaju, pretražuju i analiziraju informacije na internetu, komuniciraju s njima poznatim osobama, razumiju ulogu programa i podataka. Sve se to odvija u sigurnome digitalnom okruženju.

Domena B. se odnosi na programiranje i računalno razmišljanje s ciljem rješavanjem nekog problema i pojednostavljenjem procesa ili zadatka. Vježba se apstrakcija i kreativnost jer je programiranje širok pojam, no u primarnom obrazovanju svodi se na kreiranje kraćih i jednostavnijih programa. Temeljni pojam s kojim se kreće u nastavu programiranja naziva se algoritam, preko kojega se dolazi do dijagrama toka, a tek kasnije do čistog programiranja, pisanja naredbi i shvaćanja pravilnog slijeda programiranja.

Domena C. se usredotočuje primjenu IKT-a u nastavi informatike, ali i u nastavi ostalih predmeta. Ishodi se temelje na shvaćanju da se tehnologija neprestano mijenja i unaprjeđuje te da treba ostati u koraku s tim razvitkom. Učitelji i nastavnici informatike, prije svih, trebaju poznavati suvremene tehnologije, programe i stranice koje će njima i učenicima olakšati učenje i korištenje istih. „Jasno definiranje suvremenih vještina i kompetencija te obrazovanje nastavnika u skladu s njima pretpostavke su za nove nastavnike da budu sposobni pozitivno utjecati na ostale nastavnike u školskom okruženju.“ (Vrkić Dimić, 2013, str. 55). Temeljna odlika ove domene je jedna i od samih temelja Informatike kao znanosti, a to je da se čovjek neprekidno treba usavršavati i biti otvoren prema novim dostignućima i saznanjima.

E-društvo se u domeni D. odnosi na upotrebu i pristup raznim uslugama na netu koje učenicima, ali i svim ljudima, olakšavaju svakodnevni život. Uči se kako sigurno i točno koristiti pojedine usluge, kako biti uzoran i korektan pripadnik internetske zajednice, kako zaštititi svoju te poštivati tuđu privatnost. Temelj ove domene je izobraziti učenike kako postati obrazovani član e-društva.

Kurikulum za nastavu informatike je organiziran i opisan po razredima. Svaki razred se sastoji od zasebne tablice u kojemu su navedene domene i ishodi koji se trebaju ostvariti tim domenama. Svaki ishod ima više podishoda koji opisuju ciljeve koje učenici trebaju ispuniti na kraju pojedine školske godine. Nakon toga, kao i svakom kurikulumu, navedene su i opisane cjeline koje se odnose na vrednovanje učenika, a to su vrednovanje za učenje, vrednovanje kao učenje i

vrednovanje naučenog. Dalje, opisane su četiri razine usvojenosti (zadovoljavajuća, dobra, vrlo dobra i iznimna) koje učiteljima trebaju olakšati zaključivanje ocjena jer na temelju tih priloga mogu doći do lakog i smislenog zaključka o završnoj ocjeni, kao i o povratnoj informaciji koju će prenijeti učeniku i njegovim roditeljima s obrazloženjem zašto je zaključena ta ocjena.

Na samom kraju je naveden popis preporučenih kvalifikacija za učitelje i nastavnike informatike s obzirom na njihovu vrstu završenog obrazovanja. Određene vrste i razine studija sa svojim stečenim akademskim nazivima imaju prednost pred drugima. Magistar informatike ili primarnog obrazovanja imati će prednost pred stručnim specijalistom ili prvostupnikom (baccalaureus) informatike. Učitelj informatike može se zvati informatičarem: „Informatičari su sustavni analitičari, organizatori sustava, sustavni inženjeri, programeri, operateri i slični profili informatičkih kadrova.“ (Šimović, Maletić, Afrić, 2010, str. 17).

Dokument postupno i spiralno opisuje i nadovezuje ishode učenja koje učenici nakon njihova usvajanja trebaju znati samostalno koristiti i dalje unaprjeđivati. Nastava informatike učenike priprema da postanu samostalni, svjesni, odgovorni i kreativni građani digitalnog svijeta. Također, učenike potiče da budu u konstantnom koraku s razvitkom tehnologije, da poštuju ostale korisnike interneta, brinu o svojoj privatnosti i osobnim podacima. Tutek (2006) opisuje kako kroz nastavu učenici razvijaju kritičko mišljenje i razmišljanje, postaju kreativni i ono najvažnije, inovativni. Najvažnija karakteristika, tj. svrha samog kurikuluma je stavljanje učenika u prvi plan učenja, dok je učitelj (nastavnik) taj koji mora prilagoditi sebe, ali i nastavni sadržaj. „I nastavnici trebaju odgovorili zahtjevu vremena i tehnologiji te steći digitalnu kompetentnost.“ (Anđelić, V., 2015, str. 9). Izrazito je važno usredotočiti se na ishode učenja i rješavanje problema, a ne samo na nastavni sadržaj. Nastavni sadržaj je postavljen kao točka od koje se kreće poučavati, a putem učitelji odabiru kako će taj sadržaj proširivati i izmjenjivati s obzirom na znanje samih učenika i s obzirom na promjene u samoj Informatici koje se događaju neprestano. „Pri planiranju odgojnoobrazovnog rada vodimo računa o razvojnoj razini učenika, na temelju koje se odlučujemo za odgovarajuća računalna rješenja.“ (Bahč, T., 2023, str. 3)

### *2.3. Izborna nastava*

S obzirom da novi kurikulum za predmet Informatika treba omogućiti učenicima pripremanje za učenje i rad u društvu koje se razvojem tehnologije ubrzano mijenja, može se postaviti pitanje: *Zašto je Informatika i dalje izbran predmet u nižim razredima osnovne škole?* Od 1. – 4. razreda

učenicima (roditeljima) se daje na izbor hoće li pohađati predmet, dok je od 5. razreda Informatika obvezan predmet. Važno je učenike naučiti pravilnom i sigurnom korištenju računala jer većina njih koji u svojim kućanstvima upotrebljavaju računalo, koriste ga za video-igre te za provođenje vremena na internetu, a čije provođenje često nije pod nadzorom roditelja. Godišnji fond od 70 sati je dovoljan, barem u nižim razredima osnovne škole jer učenici već dolaze s predznanjem o računalima, neki s više, a neki s manje znanja. Ali u suštini, svi oni znaju koristiti računalnu periferiju za unos podataka. Nastavni sadržaj popraćen udžbenikom za predmet Informatika obuhvaća vrlo opće i široko područje učenja. Primjerice, prvo poglavlje u udžbenicima za prvi razred osnovne škole sadrži dvadesetak stranica o tome kako se služiti mišem, tipkovnicom te kako pokrenuti i zatvoriti određene programe i aplikacije. To su već sve automatske radnje koje su učenici znali jer je većina njih koristila računalo i prije početka primarnog obrazovanja. Može se reći da su neki sadržaji u udžbenicima i nastavnim planovima suvišni jer se puno sati odvajaju na učenje istih, a ti sati se mogu upotrijebiti za učenje stvari i sadržaja s kojima učenici nisu baš upoznati ili s kojima se ne znaju služiti na pravilan način. Informatika bi trebala postati obvezan predmet od prvog razreda osnovne škole kako bi još više dobila na važnosti te kako bi se nastavni sadržaji i ishodi u kurikulumu izmijenili s ciljem boljeg i ispravnijeg učenja.

### **3. INFORMATIČKA PISMENOST**

Već na početku treba objasniti razliku između informatičke i informacijske pismenosti. Informatička pismenost podrazumijeva poznavanje hardvera i softvera (fizičkih dijelova i opreme računala te programa i aplikacija) te njihovu uporabu u svakodnevnom životu. Informacijska pismenost ima za cilj pronaći, razumjeti, iščitati i iskoristiti danu ili pronađenu informaciju. Informaciju na internetu treba uzeti s dozom opreza, treba znati kako provjeriti dostojanstvenost iščitane informacije kako se ne bi nepotrebno dijelile lažne i neprovjerene informacije.

Informatička pismenost u Hrvata doživljava porast iz godine u godinu. Funkcije koje pruža računalo s internetom uvelike olakšava svakodnevni život. Nadrljanski (2006) navodi pet sposobnosti informatičke pismenosti: prepoznavanje potrebe za informacijom, nalaženje informacije, njezina analiza i vrednovanje, korištenje analizirane informacije te ako je informacija pouzdana – objavljivanje iste. Ubrzani način života znači učenje i upijanje informacija u hodu. Međunarodni institut za istraživanje GfK (Growth from Knowledge) proveo je kratko istraživanje u Hrvatskoj pod nazivom Kakva nam je informatička pismenost? U statistiku su ušli svi stanovnici

stariji od 15 godina, a rezultati su pokazali da u Hrvatskoj korisnici najčešće upotrebljavaju računalo za opće informiranje o dnevnim događajima, zatim za usluge e-maila te na kraju društvene mreže. U istraživanju korisnici su trebali sami sebe procijeniti kako se služe internetom, a rezultati su sljedeći:

- početnik: 24%
- dobri poznavalac: 45%
- vrlo dobar poznavalac: 24%
- stručnjak: 7%

Istraživanje se provodilo krajem 2009. godine, a može se reći da je u današnje vrijeme puno manje početnika u radu s računalom jer se mlade generacije od vrlo rane dobi susreću s računalom, digitalnim sadržajima i tehnologijom.

Informatička pismenost je temelj za daljnje programiranje na računalu pa je izrazito važno učenike od samih početaka školovanja na pravilan način učiti o pisanju, prečacima s kojima će si olakšati svakodnevno korištenje računala i njegovih sustava. „Upravo obrazovanje u kojem je učenik smješten u središnju poziciju, obrazovanje koje promovira i ostvaruje aktivne oblike učenja (i poučavanja) omogućuje razvijanje informatičke i informacijske pismenosti.“ (Vrkić Dimić, 2014, str. 385).

#### **4. PISMENOST PROGRAMIRANJA**

Programiranje je trenutno jedno od najtraženijih poslova na tržištu pa se postavlja pitanje zašto se ne programira dovoljno u hrvatskim školama. S jedne strane, učenicima možda ne treba takva vrsta nastavnog sadržaja koja može biti teško shvatljiva i zahtjevna, a i učenici koji su talentirani za programiranje, naučiti će to doma sami. S druge strane, programiranje izrazito razvija kritičko razmišljanje, sposobnost razrješavanja problema, apstraktno mišljenje, ali i vrlo važne vrline – upornost, strpljivost i maštovitost.

Programiranje nije samo pisanje računalnih programa. Programiranje je rješavanje problema, otklanjanje grešaka, razvijanje logičkog razmišljanja i računalnog razmišljanja, a to podrazumijeva razvoj strategija za rješavanje problema koji se mogu odnositi i na neprogramerska područja. Zbog toga se može reći da programiranje mijenja način razmišljanja. (Bubica, Mladenović, Boljat, 2014, str. 1)



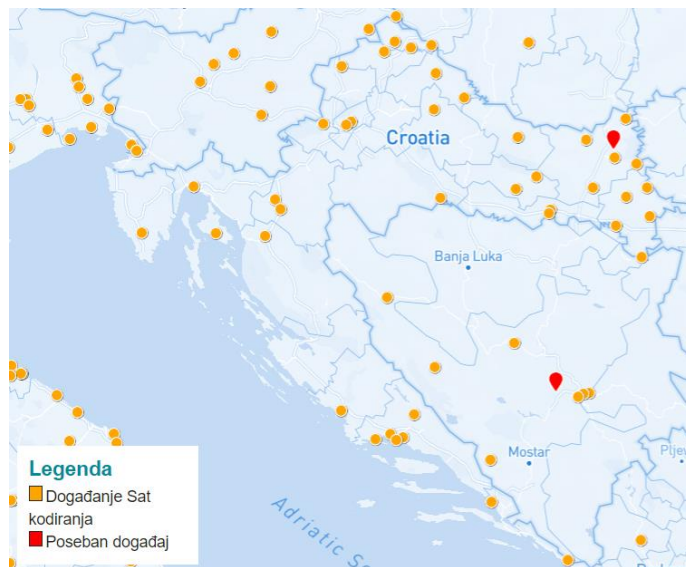
Većina učenika unutar primarnog obrazovanja ne zna mnogo o programiranju, a oni koji znaju, znaju zbog vlastitog interesa ili jer su im roditelji neka vrsta programera. S obzirom na brzi tehnološki razvitak, o programiranju se može govoriti kao o jednoj novoj vrsti pismenosti. Jedan od ključnih faktora tog razvitka je upravo programiranje jer gotovo svaka nova ideja, naprava i alat u sebi sadrže nebrojene linije koda.

## 5. PROJEKTI ZA POTICANJE PROGRAMIRANJA

Kako bi se što više učenika privuklo na programiranje, škole, ali i ostale hrvatske i europske obrazovne institucije, osmislile su razne projekte u kojima učenici aktivno i samostalno programiraju. Odaziv učenika je svake godine sve veći, a mnogi od njih i nakon završetka projekta, nastavljaju s programiranjem. Svi ti projekti su potpuno besplatni, od softvera, alata do materijala za učenje. U Republici Hrvatskoj zadnjih se godina provode projekti Europski tjedan programiranja (Code Week) i Sat kodiranja (Hour of Code).

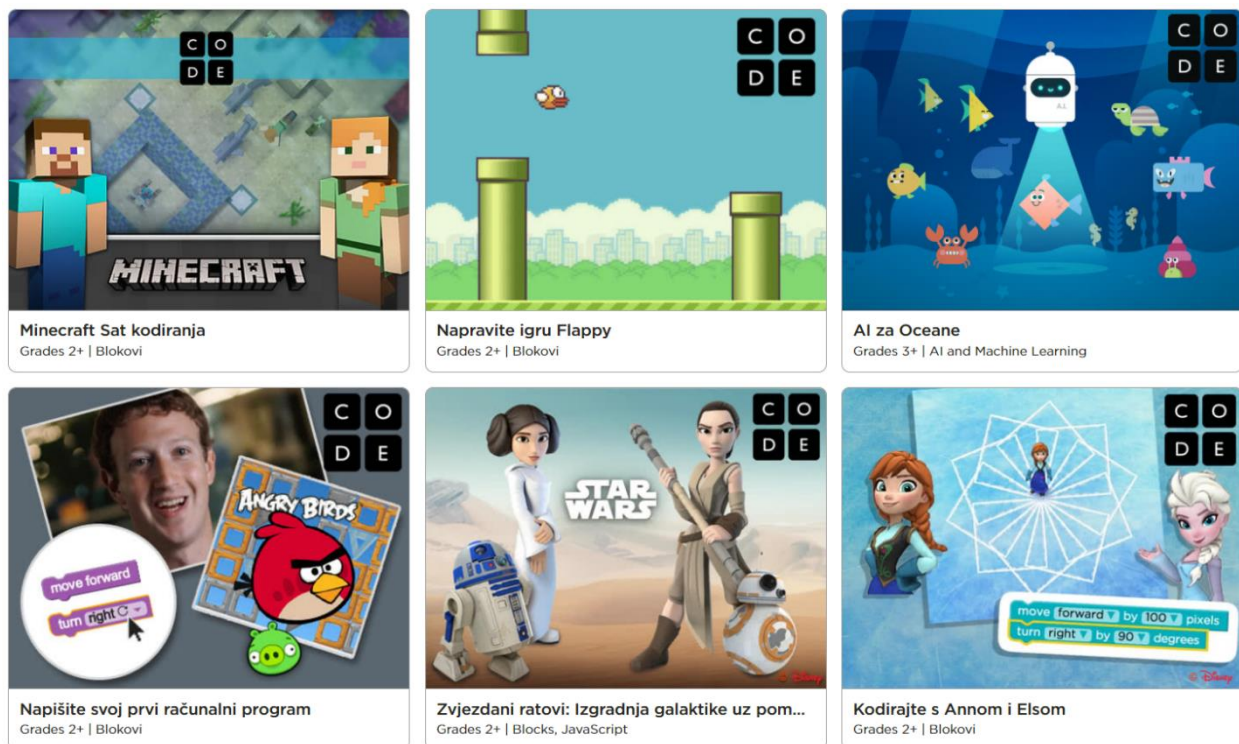
### 5.1. Hour of Code

Sat kodiranja je međunarodni projekt čija je predodžba bila da svako dijete ima pravo učiti o informatici i programiranju. Glavni cilj nije bio u sat vremena poučiti učenike kako da postanu stručnjaci i programeri, već da otkrije zabavnu i kreativnu stranu informatike. Trenutno se održava preko 60 000 projekata u cijelom svijetu, s 10 000 organizatora i s preko 1 500 000 sudionika, dok je u Hrvatskoj trenutno aktivno 78 projekata.



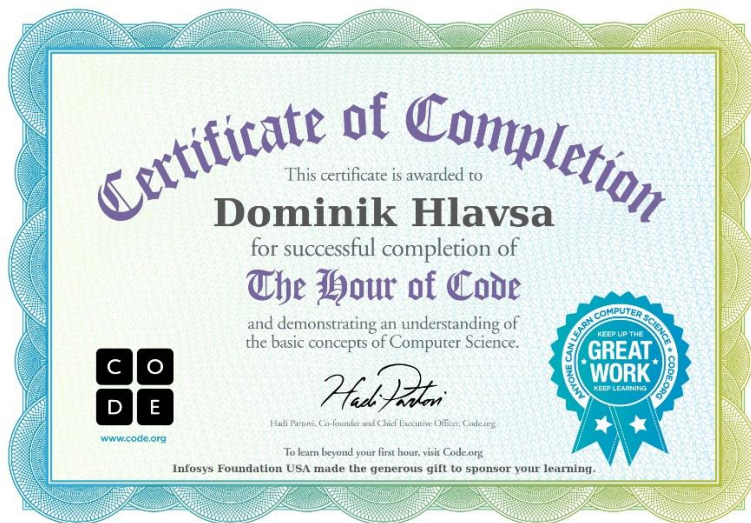
Slika 2: Hour of Code - događanja u RH

S rastućom popularnosti video igara, projekt svoje učenje gradi na popularnim igrama Minecraft, Angry Birds, ali i preko popularnih animiranih filmova Frozen, Star Wars i sl. Cilja se na učenike mlađih uzrasta, s obzirom na tematsku i vizualnu podlogu koju projekt pruža. Sat kodiranja se održava godišnje, kroz Tjedan obrazovanja za računalne znanosti (eng. CSEdWeek – Computer Science Education Week). Tradicijski se održava krajem studenog ili početkom prosinca, ali se Sat kodiranja može održati u bilo koje doba godine s obzirom da je to besplatna platforma. Koncept je vrlo jednostavan – učitelj treba odabrati i proučiti jednosatni vodič koji će kasnije proći sa svojim učenicima. Postoje mnogobrojni vodiči koji su prilagođeni prema znanju i uzrastu. Bitna karakteristika je i tzv. offline programiranje gdje se nalaze lekcije koje za svoju svrhu imaju razvijanja rada u timu i međusobne komunikacije između učenika kao i kognitivnih procesa i apstraktnog razmišljanja. Hour of Code postao je i ostao projekt zajednice koji i dalje postoji zahvaljujući rastućem broju novih članova – učitelja i učenika. Uspjeh se očitava u širokom sudjelovanju učenika svih uzrasta, rasnih i socioekonomskih skupina, a mnogi koji prođu taj prvi sat kodiranja, ostaju i nadograđuju svoje znanje preko drugih raznih projekata i tečaja koji se nude.



Slika 3: Hour of Code - primjeri projekata

Na kraju sata, učitelj za svoje učenike može ispisati certifikate na kojima piše njihovo ime te kako su uspješno pohodili i položili Hour of Code tečaj. Učenicima je to odlična povratna informacija i uspomena s kojom se mogu hvaliti i biti ponosni kasnije u životu.



Slika 4: Hour of Code - certifikat

## 5.2. Code Week

Code Week je volonterski projekt koji se održava u školama diljem Europe već 10 godina. Učitelji, nastavnici i ravnatelji s ovim projektom nude učenicima mogućnost istraživanja i analiziranja programiranja i vlastite digitalne mašte. Učenje i poučavanje programiranja pridonosi shvaćanju svijeta oko sebe koji se konstantno mijenja. Također, potiče sudionike na slojevito razumijevanje o radu te funkcioniranju tehnologije, ali i razvija pojedinačne sposobnosti i vještine kako bi istraživao, izgrađivao i nadograđivao svoje, ali i ideje drugih. „Čini se da je najvažnija sposobnost „naučiti kako učiti“. Jer, ono što znamo danas, biti će zastarjelo mnogo prije nego što vjerujemo. Moramo pronalaziti nove putove da radimo stare stvari i nove putove da radimo nove stvari.“ (Hutinski, Ž., Aurer, B., 2009, str. 265). Code Week ima istu shemu kao i Hour of Code, s jednom bitnom razlikom, a to je duljina trajanja samog projekta. Kako i naziv govori, traje tjedan dana kojih treba pomno isplanirati. Volonteri i ambasadori s više iskustva o programiranju stvaraju projekte i koordiniraju tjedan programiranja za ostale učesnike. No, u svakom trenutku bilo tko može kreirati vlastitu aktivnost i postaviti je na stranicu Code Week-a. Na službenoj stranici <https://codeweek.eu/scoreboard> mogu se vidjeti trenutni projekti te poredak svjetskih zemalja s najviše događanja. Treba napomenuti kako je lista napravljena prema broju popisanih događanja s

obzirom na ukupno stanovništvo te zemlje pa tako zemlje s manjim brojem događanja mogu stajati visoko na listi jer ukupno imaju i manji broj stanovnika. Daleko ispred svih prednjači Austrija s preko 1500 trenutnih projekata, slijede je Luksemburg i Malta, dok su na začelju zemlje Bliskog istoka. Hrvatska je trenutno 10. zemlja na listi što je vrlo dobar rezultat, pogotovo kada se uzme u obzir da su iza Hrvatske ostali SAD, Francuska, Njemačka, UK i druge bogatije zemlje.



Slika 5: Code Week tabela zemalja

Najveći obol ovaj projekt daje zabavi i učenju kroz kreativnost, a jedan od ambasadora europskog tjedna programiranja, Alessandro Bogliolo, najbolje je opisao čemu služi ovakav projekt:

*„Od pamtivijeka smo brojne stvari stvarali s pomoću kamena, željeza, papira i olovke. Sve to preoblikovalo je naše živote. Sada živimo u drukčijem vremenu u kojemu se naš svijet oblikuje programiranjem. Različita vremena traže drukčije poslove i vještine. Tijekom Europskog tjedna programiranja svakom Europljaninu želimo pružiti priliku da otkrije programiranje i zabavi se s njime. Hajde da naučimo programirati kako bismo oblikovali svoju budućnost.“*

### 5.3. Run Marco

Run Marco postoji u više verzija – kao web stranica i aplikacija. Osmišljena je s ciljem zabavnog i vizualnog učenja programiranja, a namijenjena je mlađim uzrastima (6 – 12 godina). U hrvatskom

jeziku je popularniji naziv Bjež' Marko ili Trči, Marko. Platforma pripada grčkoj tvrtki Allancode koja se od 2014. bavi razvijanjem raznih softvera i programa. Idealna je za one koji se prvi put susreću s programiranjem, a način na koji se programira je vrlo jasan i pregledan. Pomoću tekstualnih blokova koji se slažu jedan ispod drugog, stvaraju se naredbe kojima se lik Marko pomiče po zaslonu. Postoje dva lika, lik dječaka Marca i lik djevojčice Sophie. Postoji mogućnost odabira mape, tj. prostora po kojem će se kretati odabrani lik (džungla, snježno kraljevstvo i dr.) Osnovne naredbe su: *korak naprijed*, *okret ulijevo*, *okret udesno*. Ova aplikacija se često koristi za vrijeme Tjedna programiranja, ali neke škole je koriste i u drugim prilikama, za vrijeme redovite nastave informatike.



Slika 6: Run Marco - primjer zadatka

## 6. PROGRAMSKI JEZICI

Najosnovnija definicija programskog jezika bi bila da je to jezik za pisanje programa koje računalo razumije i izvršava. Definiran je sintaksama, instrukcijama i pravilima. Osnovna podjela programskih jezika je na niže i više. Niži jezici mogu se dalje podijeliti na strojni i asemblerski. Strojni programski jezik radi na principu binarnog koda i to je zapravo jedini jezik koje računalo prepoznaje. Asemblerski jezik radi na sličnom principu kao i strojni, samo što je njegov binarni kod zamijenjen s grafičkim simbolima (OR, NOT, AND). S druge strane, viši programski jezici su kompliciraniji i složeniji (sintaktično). Temelje se na nizovima, varijablama te složenim

algebarskim i matematičkim izrazima. Takvi jezici su vizualniji i bliži su ljudskim jezicima. Čitljiviji su pa je korisnicima olakšan rad i razumijevanje. Primjeri takvih jezika su Pascal, Python, Java, C#, Ruby. No, samo računalo ne razumije takvu vrstu jezika. Stoga, potrebni su zasebni programi koji pretvaraju tekst u niži programski jezik (binarni kod) koji računalo razumije. Jedan takav primjer je kompajler (eng. compiler) ili jezični prevoditelj koji, prije izvršenja programa, pretvara, tj. prevodi cijeli kod u strojni jezik. Drugi, sličan primjer, bio bi interpreter koji prilikom izvršenja programskog koda prevodi naredbu po naredbu.

### *6.1. Programski jezici namijenjeni učenju*

Prvi programski jezik s kojim će učenici krenuti u svijet programiranja treba biti jasan, pregledan i interaktivan. Već na prvi pogled treba privući učenike i ostalu djecu na proučavanje samog rada programa. Pomoću takvog jezika usvojiti će osnove koje se kasnije mogu upotrijebiti i u ostalim programskim jezicima. Mrđen, Livaja i Acalin (2019) definirali su cilj programskog jezika namijenjen edukaciji: „Glavni cilj svakog programskog jezika koji je namijenjen učenju programiranja mora biti usvajanje osnovnih znanja i koncepta programiranja koji se mogu primijeniti u ostalim programskim jezicima, a ne samo u jeziku namijenjenom učenju programiranja.“ U većini osnovnih, ali i srednjih strukovnih škola programiranje se poučava na tradicionalan način (učiteljevo prezentiranje teme nakon čega slijedi zadavanje zadataka). Učenik ne smije biti samo objekt koji se poučava, već aktivni sudionik procesa programiranja. „Tradicionalni načini učenja i podučavanja zamjenjuju se istraživačkim i problemskim metodama što osobu koja uči stavlja u poziciju samostalnog istraživača i korisnika informacija koji je aktivno uključen u proces traženja informacija.“ (Lazić-Lasić, Špiranec, Banek Zoricaf, 2012, str. 129). Ovdje veliku ulogu igra učitelj koji treba percipirati problem učenicima te ga zajedno s njima na glas postupno rješavati, pritom ne odajući odgovore, već samo natuknice i naznake odgovora kako bi učenici samostalno, ali i uz pomoć suradničkog učenja došli do rješenja problema. „Bitno je da učenici ne dobivaju servirane informacije i gotova rješenja, jer se tada ne potiče njihovo razmišljanje, inicijativa, usmjerenost na zadatak, samokontrola i samostalnost.“ (Krelja Kurelović, Vasiljević, Bodiš, 2013, str. 8).

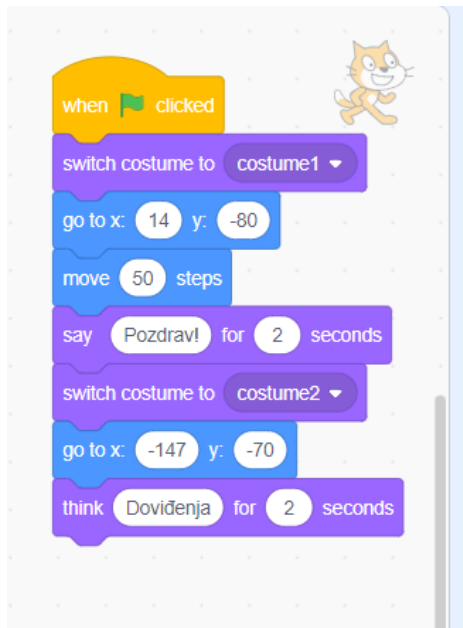
### *6.2. Scratch*

Scratch je programski jezik stvoren 2003. godine na prestižnom Tehnološkom institutu Massachusetts (MIT). Najveća svjetska zajednica za programiranje u kojem sudjeluju djeca upravo

se nalazi na platformi Scratch. Dostupan je u preko 200 država i na preko 70 jezika te je potpuno besplatan za korištenje. Postao je popularan zahvaljujući pametno osmišljenom sučelju u kojemu djeca mogu stvoriti kreativne i interaktivne priče, prezentacije, igre, simulacije i animacije. Scratch je za svrhe korištenja ciljao dobnu skupinu 8 – 16 godina, ali ga zapravo djeca svih uzrasta mogu koristiti, kao i odrasle osobe. Ova platforma se koristi u raznim ustanovama, pa se tako Scratch programe može pronaći u školama, knjižnicama, muzejima, ali i u običnim kućanstvima. U školama Scratch se počinje poučavati u svim stupnjevima obrazovanja – od osnovne, srednje škole pa sve do fakulteta. Također, njegova primjena je široka i stoga se može koristiti u raznim predmetima – informatika, jezicima, matematika, fizika i dr. Jedan od osnivača Scratcha Mitchel Resnick (2012) govori kako korisnici Scratcha uče važne strategije za rješavanje problema, kvalificiraju se za dizajniranje projekata i međusobno komuniciranje ideja. Scratch je postao najpopularnija platforma za početničko programiranje zbog sljedećih karakteristika:

- Vizualno programiranje (programiranje blokovima – nema ručnog upisivanja koda; svrha takve vrste programiranja je da korisnika potakne na logičko razmišljanje i rješavanje problema)
- Kreativno stvaranje/izražavanje (širok raspon audio i video izvora)
- Zajednica i dijeljenje projekata (široka zajednica koja međusobno dijeli, analizira i stvara projekte)
- Snažna i pouzdana obrazovna vrijednost (odličan temelj pri učenju o konceptima programiranja)
- Svestranost (mogućnost stvaranja jednostavnih projekata do složenih igrica i interaktivnih simulacija)

Sam način programiranja je vrlo lagan, blokovi se spajaju u smislenu cjelinu kako bi program funkcionirao. „Program omogućuje programiranje mišem povlačenjem i uklapanjem blokova koji se mogu spojiti samo ako to odgovara u određenom sintaksnom smislu i tako omogućuje učenicima da se fokusiraju na probleme koje oni žele riješiti.“ (Mrđen i sur., 2019, str. 117).



*Slika 7: Scratch - primjer jednostavnog koda*

U Hrvatskoj se u Scratchu programira od 1. do 4. razreda osnovne škole, a zatim se prelazi na složeniji programski jezik Python.

### *6.3. Python*

Python je besplatni programski jezik te je jedan od najpopularnijih i najkorištenijih programskih jezika današnjice. Stvorio ga je nizozemski programer Guido van Rossum krajem 20. stoljeća, a prvi put je pušten u javnost 1991. godine. Van Rossumova vizija je bila stvaranje programskog jezika koji će biti jednostavan, ugodan za čitanje i kodiranje. Python je objektno-orientirani, ali i skriptni jezik (sadrži biblioteke u kojima se nalaze gotove funkcije i objekti). Pogodan je za rano učenja programiranja jer je vrlo lako naučiti sintaksu, a kodovi se pišu kratko, kraće nego npr. u C-u.



*Slika 8: Python – logo*



U školama se u Pythonu počinje programirati u 5. razredu, nakon što učenici dođu s predznanjem o programiranju koje su stekli programirajući u Scratchu. Osnovna razlika između ta dva programska jezika je da je Python jezik opće namjene, dok Scratch služi samo u obrazovne svrhe. Oni učenici koji nisu pohađali izbornu nastavu informatike od 1. razreda, teže će razumjeti i raditi u Pythonu s obzirom da je to viši programski jezik sa širom sintaksom i manje vizualnim sučeljem.

## **7. UVOĐENJE PROGRAMIRANJA U NASTAVU INFORMATIKE PRIMARNOG OBRAZOVANJA U REPUBLICI HRVATSKOJ**

Upišemo li u internetsku tražilicu *programiranje u osnovnoj školi* i otvorimo službene stranice škola koje spominju programiranje u svojoj nastavi, na mnogim stranicama vidjet ćemo kako se programirati počinje tek u 5. razredu. U domeni B kurikuluma piše: koristi se programskim alatom za stvaranje programa u kojemu se koristi ulaznim i izlaznim vrijednostima te ponavljanjem (Ministarstvo znanosti i obrazovanja, 2018). Tek se ovdje spominje nekakva vrsta programskog alata, dok se u 3. razredu spominje stvaranje programa koristeći vizualno okruženje, što se odnosi na pisanje u bilježnicu ili u bloku za pisanje. Također, pojam algoritma, koji je temelj i od kojega se kreće sa samim objašnjavanjem i poučavanjem programiranja, spominje se u kurikulumu prvi puta u 5. razredu. Algoritam je ništa više nego pisanje recepta, a za to su učenici i više nego kadri mnogo ranije od 5. razreda. Nadalje, pojam petlje, koja je također vrlo važna u programiranju, prvi puta se spominje u 6. razredu, a tek u 8. razredu programiranje se korelira s drugim nastavnim predmetima (matematika, fizika i dr.). No, učitelji informatike u nižim razredima sve više preuzimaju inicijativu i poučavaju osnove programiranja služeći se dodatnim obrazovnim sadržajima i vlastitim znanjem. „Kvalitetnim informatičkim obrazovanjem koje se temelji na računalnom razmišljanju i kreativnosti omogućuje se razumijevanje i mijenjanje svijeta koji nas okružuje.“ (Ministarstvo znanosti i obrazovanja, 2018).

Želimo li pratiti implementaciju programiranja u primarnom obrazovanju, iz domene B. Računalno razmišljanje i programiranje trebamo iščitati ishode učenja. U svakom razredu pod tom domenom nalaze se po dva ishoda koji učenici trebaju ostvariti na kraju školske godine. U prvom razredu se kreće od osnova – rješavanje jednostavnih logičkih zadataka (mozgalice, slagalice, memory i slično). Nakon rješavanja danih zadataka, sljedeći korak je da učenici osmišljavaju vlastite zagonetke i logičke zadatke, stvaraju nove pouzdane korisne informacije, što je sljedeći ishod naveden u kurikulumu. „Stvaranje informacija odnosi se na sposobnost korištenja računala kako

bi dizajnirali i proizveli originalni informacijski proizvod u određenu svrhu ili za određenu publiku. Takvi originalni proizvodi mogu biti potpuno novi ili se mogu temeljiti na određenoj skupini informacija u svrhu razvoja novih ideja i razumijevanja.“ (Roth, Dekanić, Ružić, 2014, str. 25).

- B.1.1 rješava jednostavan logički zadatak
- B.1.2 prati i prikazuje slijed koraka potrebnih za rješavanje nekoga jednostavnog zadatka. (Ministarstvo znanosti i obrazovanja, 2018)

Dalje, u drugom razredu dva ishoda koja se trebaju ostvariti su analiziranje uputa i ispravljanje pogrešnog redoslijeda te stvaranje niza uputa u kojima se upotrebljava ponavljanje.

- B.2.1 analizira niz uputa koje izvode jednostavan zadatak, ako je potrebno ispravlja pogrešan redoslijed
- B.2.2 stvara niz uputa u kojemu upotrebljava ponavljanje. (Ministarstvo znanosti i obrazovanja, 2018)

Ovo su zapravo prvi koraci ka učenju o algoritmu i petlji, samo što se u primarnom obrazovanju uopće ne spominju ti pojmovi kao takvi, već se na vrlo jednostavan i štur način objašnjavaju i definiraju preko vizualnih primjera. U kurikulumu za 2. razred u razradi ishoda prvi je puta navedena eventualnost korištenja robota pri ostvarivanju prethodno navedenih ishoda, no, nema svaka škola mogućnost nabaviti i koristiti robote u svojoj nastavi. U trećem razredu navedeni su vrlo važni ishodi:

- B.3.1 stvara program korištenjem vizualnoga okruženja u kojem se koristi slijedom koraka, ponavljanjem i odlukom te uz pomoć učitelja vrednuje svoje rješenje
- B.3.2 slaže podatke na koristan način (Ministarstvo znanosti i obrazovanja, 2018)

Ovdje se prvi puta spominje Scratch – grafički program s naznakom programiranje za najmlađe. Učenici uz pomoć učitelja osmišljavaju korake koji vode prema rješenju nekog zadatka ili problema. Čop i Topolovec (2009) govore kako nastavnik informatike ne mora biti kreativan, nego samo treba osigurati slobodan razvoj učenika i stvoriti ugodnu atmosferu koja potiče razvoj interesa i stjecanje temeljnih znanja koji su u skladu s učenikovom dobi. Na kraju, u 4. razredu, razrade ishoda se dalje nadovezuju i šire, a ishodi koji se trebaju ostvariti su sljedeći:

- B.4.1 stvara program korištenjem vizualnog okruženja u kojem koristi slijed, ponavljanje, odluku i ulazne vrijednosti
- B.4.2 rješava složenije logičke zadatke s uporabom računala ili bez uporabe računala (Ministarstvo znanosti i obrazovanja, 2018)

Ovdje, učenici se prvi puta susreću sa svim elementima pisanja programa: ulaz, obrada podataka i izlaz. Važno je donijeti pravilne odluke i točan raspored pisanja programa kao što je važno pratiti i razumjeti što i kako program radi kako bi učenici na kraju mogli analizirati i raspravljati o problemima i rješenjima koja su imala prilikom pisanja programa.

Bitno je istaknuti jednu nepoželjnu rečenicu koja se ponavlja od prvog do trećeg razreda osnovne škole, a nalazi se u kurikulumu pod preporukama za ostvarenje odgojno-obrazovnih ishoda: „Aktivnosti se mogu izvoditi računalom ili bez njega.“ (Ministarstvo znanosti i obrazovanja, 2018). Ova rečenica daje odgovore na prethodno postavljena pitanja u ovom radu – zašto je informatika i dalje izborni predmet te zašto se ne programira dovoljno (kvalitetno) u hrvatskim školama.

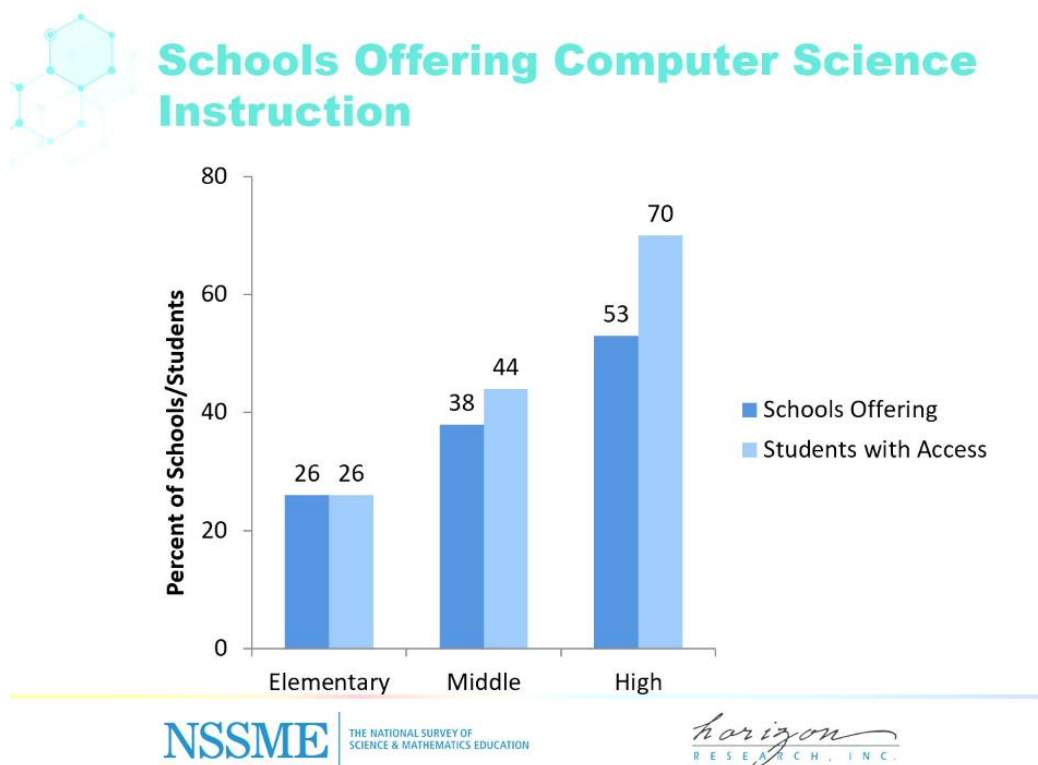
## **8. UVOĐENJE PROGRAMIRANJA U NASTAVU INFORMATIKE PRIMARNOG OBRAZOVANJA U SJEDINJENIM AMERIČKIM DRŽAVAMA**

Školsko obrazovanje u SAD-u je podijeljeno na sljedeći način:

- *Elementary School* u trajanju od 6 godina
- *Middle School* u trajanju od 3 godine
- *High School* u trajanju od 4 godine

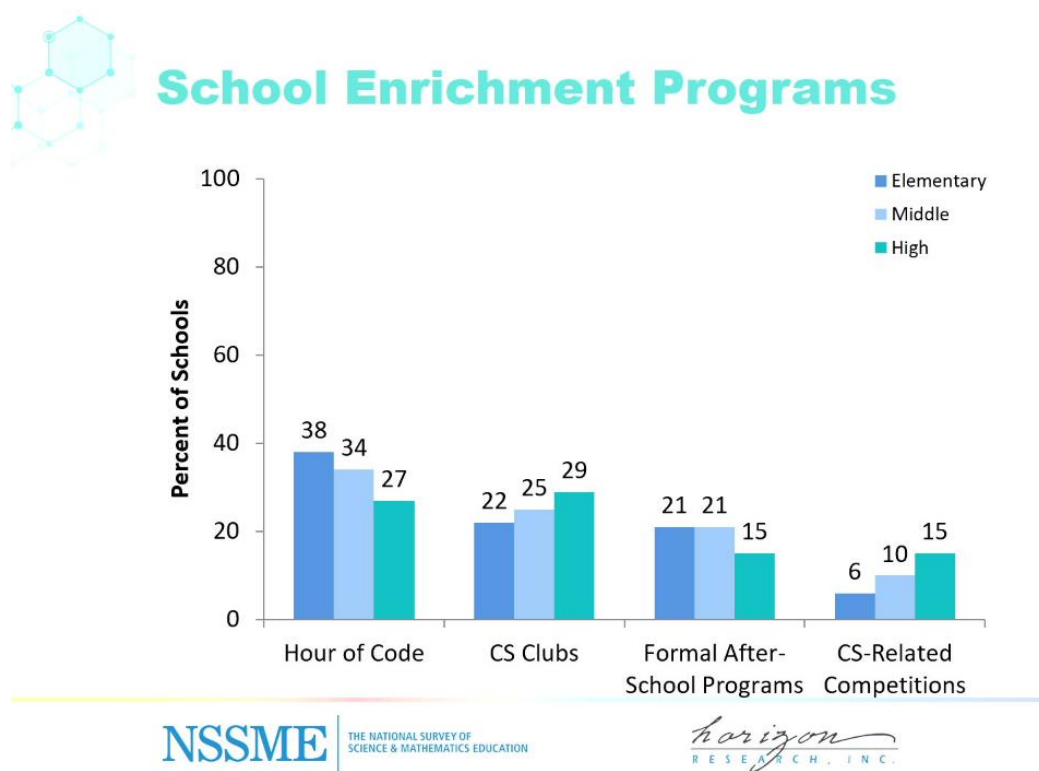
Prije pohađanja škole, američka djeca mogu pohađati vrtić, *preschool*. Dok, nakon završetka srednje škole (*High School*) oni koji žele nastaviti svoje obrazovanje, upisuju terciarni oblik, tj. fakultete ili više škole. Amerikanci imaju državno tijelo nalik hrvatskom Ministarstvu znanosti i obrazovanja pod nazivom *Department of Education*. Taj odjel dužan je osmisliti i nadzirati provođenje nacionalnih strateških smjernica u obrazovanju te provoditi kurikularne reforme ukoliko je potrebno. Kako se SAD sastoji od 50 saveznih država, svaka država ima veliku autonomnost i neovisnost u provođenju vlastita plana i programa nastave. Lokalna samouprava svake države sama odabire kojim će se udžbenicima učenici služiti te koje će nastavne sadržaje poučavati kojim redoslijedom. Zato je vrlo teško ispitati cjelokupan američki sustav obrazovanja,

čak i za jedan predmet kao što je informatika, kada se uzme u obzir taj velik broj lokalnih samouprava koje donose odluke. U primarnom obrazovanju (Elementary School) nema niti naznaka o informatici, a kamoli o programiranju. Glavni predmeti su matematika i engleski jezik. Ostali predmeti se spominju i crticama, ali se grade temelji povijesti, geografije, tjelesne i glazbene kulture kao i fizike, kemije i biologije (u SAD-u su zadnja tri predmeta objedinjena u jedan predmet pod nazivom *Science*). Informatika se u SAD-u kao predmet zove *Technology* ili *Computer Science* i prvi put se javlja u 9. razredu (High School). Zanimljiv je podatak da je u 2019. samo četvrtina nastavnika koji predaju predmet *Computer Science* imalo diplomu za to područje. U SAD-u se ne programira u primarnom obrazovnu niti se planira uvesti u skorije vrijeme. Američki *Department of Education* 2019. je provelo zanimljivo istraživanje u 2000 škola diljem Amerike s ciljem saznavanja koliko je predmet *Computer Science* usađen u američko obrazovanje, s naglaskom na srednjoškolsko, ali i s podacima o primarnom obrazovanju. Istraživanje pod nazivom *What Do We Know about Computer Science Education?* (2019) prikazalo je sljedeće:



Slika 9: Prikaz škola koje nude Computer Science

Iz viđenog se može zaključiti da vrlo mali broj škola primarnog obrazovanja nude neku vrstu instrukcija iz područja informatike, dok je u višim razredima i srednjoj školi taj broj nešto veći, s naglaskom na tome da nemaju svi učenici pristup informatičkoj opremi – bilo to doma ili u školi. Ovdje se i dalje ne misli na predmet informatike, nego samo dodatno, izborno poučavanje koje škole nude u okviru imena *Computer Science*.

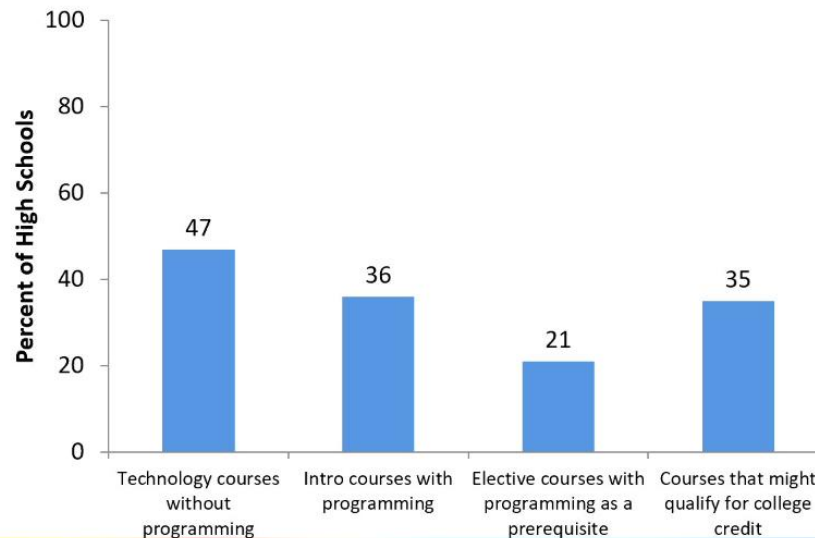


Slika 10: Programi kojima se potiče programiranje

Nadalje, može se uočiti kako neke škole pokušavaju u svoj plan i program uvesti programiranje u svakodnevnu nastavu preko raznih platformi i projekata. Jedna od tih je i *Hour of Code*, spominjan ranije u ovom radu. Ta platforma najviše je zastupljena u primarnom obrazovanju jer su njeni sadržaji prikladni upravo za taj uzrast. Dalje su navedene neke vrste programerskih klubova i izvannastavnih aktivnosti vezanih za programiranje, ali u manjem broju nego što bi to trebalo biti.



## High Schools Offering Computer Science and Technology Courses



NSSME THE NATIONAL SURVEY OF SCIENCE & MATHEMATICS EDUCATION

horizon RESEARCH, INC.

Slika 11: Srednje škole i programiranje

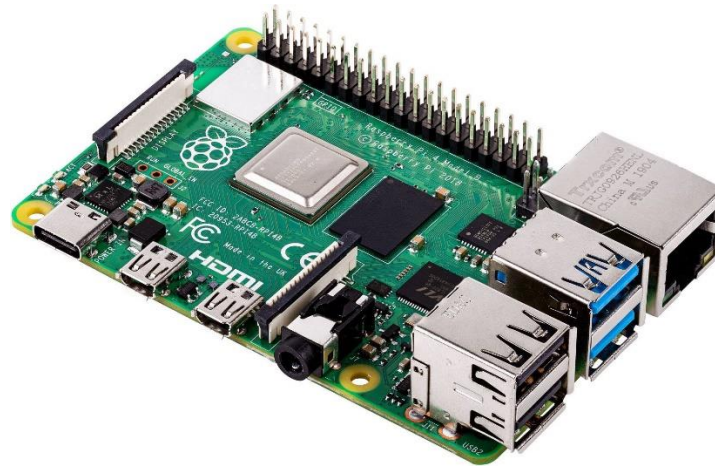
Premda ovaj grafikon prikazuje podatke za srednje škole, a ne za primarno obrazovanje, prikazan je zbog negativne činjenice da čak 47% srednjih škola ima smjerove ili neku vrstu predmeta sličnog informatici u kojima se uopće ne poučava o programiranju. Nadalje, tek 36% ispitanih škola nudi samo osnovni, uvodni dio o programiranju. Dok učenici dođu u srednju školu, trebaju imati kakvo-takvo znanje i iskustvo u programiranju. Dalje se ne može govoriti o prednostima i manama uvođenja programiranja u nastavu informatike u SAD-u jer u službenim dokumentima nema naznake o provođenju istog.

### 9. UVOĐENJE PROGRAMIRANJA U NASTAVU INFORMATIKE PRIMARNOG OBRAZOVANJA U ENGLESKOJ

U Engleskoj primarno obrazovanje počinje u 4. ili 5. godini djetetova života. Primarno obrazovanje u osnovnoj školi dijeli se na dvije faze (*key stages*). Isti je slučaj i u srednjim školama (*Secondary school*). Prva tzv. ključna faza u osnovnoj školi traje dvije, dok druga traje četiri godine. U obje faze obvezni predmet je i informatika. U početku je naglasak na shvaćanju principa rada računala, kasnije se poučava i o pisanju programa. Veliki naglasak je dan na važnost razumijevanja programiranja umjesto samo pisanja linije kodova. Ministarstvo obrazovanja u

Engleskoj je od početka poduzelo prave korake ka kvalitetnom obrazovanju mladih umova. Jedan od koraka je i ponovna edukacija učitelja. Prvi naglasak je bio obučiti ih o nastavi na daljinu i online nastavi, no edukacija se širila i na komponente IKT-a u obrazovanju i digitalne pismenosti. Nažalost, kako je edukacija bila izborna za učitelje, većina ih je odlučila ne pohađati danu edukaciju.

No, Engleska je prva država, uz Estoniju, koja je u Europi uvela programiranje u primarno obrazovanje. Jedan od ciljeva engleskog kurikulumu na području računarstva bilo je pripremiti učenike za pravilno korištenje logičkog zaključivanja pri stvaranju i analizi jednostavnih programa. U prijevodu, učenik bi trebao znati i razumjeti zašto i na koji način program radi. Ako ne radi, treba razumjeti kako i gdje je došlo do pogreške. Engleski kurikulum je vrlo cijenjen i hvaljen u Europi, no i dalje najveći problem predstavlja upravo programiranje. Iako dovoljno zastupljen u srednjoj školi, u osnovnoj ipak u manjoj mjeri. 2013. u Engleskoj je pokrenuta inicijativa gdje su se školama poklanjali Raspberry Pi malih računala. Njegova glavna svrha je promicanje nastave informatike u osnovnim školama. U engleskim školama koristio se pri programiranju u Scratchu i nešto manje u Pythonu.



*Slika 12: Raspberry Pi*

Kroz ovu ideju, puno više učenika se privuklo prema programiranju jer su mogli preko vizualnog pomagala vidjeti kako i što napisan program radi i izvršava.

Novi kurikulum donesen 2013. godine odredio je glavne stupnjeve usvajanja iz kurikuluma računarstva:

- 1. ključna faza osnovne škole – učenje o algoritmima, pisanje i objašnjavanje skupova instrukcija
- 2. ključna faza osnovne škole – pisanje programa i otkrivanje pogrešaka (debugiranje), prvi koraci pisanja sekvenci i petlji
- 3. ključna srednje škole – samostalno pisanje programa u određenom programskog jeziku, korištenje Booleove algebre, učenje o raznim softverima

Programiranje u primarnom obrazovanju u engleskim školama iz godine u godinu doživljava novi preporod te, kao i sama tehnologija, raste i razvija se. Potrebno je konstantno usavršavanje učiteljskog kadra kako bi ostali u koraku s razvojem i novim načinima programiranja. Engleski učitelji, s obzirom da žive i rade u tako razvijenoj zemlji, imaju puno više mogućnosti i izgleda metodički pravilno uvesti nastavu programiranja već od samih početaka primarnog obrazovanja.

## **10. UVOĐENJE PROGRAMIRANJA U NASTAVU INFORMATIKE PRIMARNOG OBRAZOVANJA U ŠPANJOLSKOJ**

U Španjolskoj se, isto kao i u Republici Hrvatskoj, sport odvojio od ministarstva obrazovanja. Tijelo nadležno za provođenje politike obrazovanja u Španjolskoj zove se Ministarstvo obrazovanja i strukovnog osposobljavanja (MEFP). Pod njegovu nadležnost pada osnovnoškolsko i srednjoškolsko obrazovanje, dok za tercijarno obrazovanje, tj. sveučilišno stoji zasebno Ministarstvo sveučilišta. 2018. godine u Španjolskoj je zabilježeno 550 000 učitelja u radnom odnosu (47,42 milijuna stanovnika).

Školski obrazovni sustav u Kraljevini Španjolskoj je podijeljen na sljedeći način:

- predškolska ustanova (neobvezno)
- Primarno - osnovna škola (obvezno)
- Obvezno srednje obrazovanje - educación secundaria obligatoria (obvezno)
- Srednje obrazovanje - bakalaureat (neobavezno)

Primarno, osnovno obrazovanje traje šest godina, nakon čega slijedi 4 godine srednjoškolskog obrazovanja. Nakon završenih 10 godina obrazovanja učenicima se dodjeljuje Svjedodžba o



obrazovanju. Tu prestaje obvezno obrazovanje, nakon čega slijedi tzv. viša srednja škola u trajanju od dvije godine. Nakon završene dvije godine učenici dobivaju diplomu. Konačno, mogu izabrati između stručne obuke ili sveučilišnog obrazovanja, a za to je potrebno položiti stručni ispit (Prueba de Acceso a la Universidad).

Do 2013. godine programiranje se nije poučavalo u primarnom obrazovanju. Kako je rasla potreba za programerima i kvalitetnom spremom, ministarstvo je uvelo testni projekt uvođenja programiranja u primarno obrazovanje. Nije stvoren novi predmet, nego se programiranje poučavalo u sklopu likovnih i prirodnih znanosti. U toj praksi koristio se Scratch kao idealan početak, pogotovo za korelaciju s drugim predmetima. Unutar nastavnog programa, učilo se o svim konceptima programiranja, krenuvši od algoritma. Učenici su postepeno učili o naredbama, sljedovima i petljama. Nakon dvije godine učenički rezultati su bili izvrsni te je ministarstvo odlučilo uvesti programiranje u osnovnoškolsko obrazovanje, no, krenuvši tek od 5. razreda. Nakon završetka projekta, 2015. godine u srednjim školama uveden je izborni predmet pod nazivom *kodiranje*. Kako sve autonomne jedinice u Španjolskoj imaju veliku slobodu pri kreiranju vlastita plana i programa nastave, u najvećim i najrazvijenijim jedinicama (Katalonija i Madrid) u primarno osnovnoškolsko obrazovanje uveden je pojam *Računalno razmišljanje*, koji se također nalazi i u hrvatskom kurikulumu. Zadnjih godina, španjolske škole se dosta zalažu i provode *Code Week* sa svojim učenicima. Dakle, u primarnom obrazovanju postoji informatika kao izborni predmet u kojem se, između ostalog, poučava i programiranje, ali u vrlo malim dijelovima. Jedan od glavnih autora kurikularne reforme u Hrvatskoj, učitelj i pedagog Boris Jokić, rekao je, kada se hrvatska informatika uspoređivala s ostalim europskim zemljama (uključujući i Španjolsku) – „Djeca će, ako se uključe u izbornu nastavu informatike, od prvog razreda moći učiti programiranje.“

Zaključno, niti u Španjolskoj kao razvijenoj zemlji, nastava programiranja u 2023. i dalje nije dovoljno zastupljena i promovirana. Postoje obećavajuće naznake, ali čini se da je problem u nedovoljno obrazovanom učiteljskom kadru koje nije podobno poučavati i prenositi znanje o programiranju svojim učenicima.

Iako je u europskim zemljama vidno zastupljenije programiranje od početka školovanja naspram programiranja u SAD-u, Amerikanci imaju veći broj stručnjaka i programera. Također, europski i američki pogled na informatiku nije u potpunosti jednak:

„Informatika označava mladu znanost za koju nemamo jedinstvenu definiciju. Današnji pristup informatici nastao je u 20. stoljeću utjecajem visoke tehnologije. U svijetu postoje različiti nazivi i odnosi prema informatici. Europa, za razliku od Amerike, upotrebljava naziv informatika i u njemu ujedinjuje dvije znanosti: informacijsku (information science) i znanost o računalu (computer science).“ (Gugić i sur., 1997, str. 47)

## ZAKLJUČAK

Programiranje sve više ulazi u škole zahvaljujući blistavim umovima mladih i novopečenih učitelja i nastavnika informatike koji razumiju kako se svijet ubrzano mijenja i koji shvaćaju kako treba ostati u koraku s vremenom, neprestano obrazovati sebe i svoje učenike. Iako su programski zadaci i problemi u udžbenicima informatike u ograničenom broju, danas postoje razni projekti koji se mogu ukomponirati u nastavu informatike te se mogu korelirati s drugim predmetima. Učenje programiranja pomoću Scratcha u primarnom obrazovanju označilo je veliki iskorak i napredak prema boljoj budućnosti, učenici od malena shvaćaju samu srž programiranja – zašto program radi na način koji radi.

Usporedba implementacije programiranja u Republici Hrvatskoj s implementacijom ostalih razvijenijih zemalja donijelo je zanimljive zaključke. Hrvatska kurikulumom iz 2018. godine prednjači, barem na papiru, po kvaliteti i zastupljenosti programiranja i računalnog razmišljanja u nastavi informatike. U Španjolskoj i SAD-u veliki problem predstavlja informatički kadar koji, prema svim rezultatima, nije dovoljno kvalitetan kako bi prenio znanje mladim generacijama. Iako je programiranje apstraktan pojam, bitno je naći pravo tumačenje i kvalitetan metodički način kako bi se mladim umovima usadili pojmovi algoritma, zatim i programiranja. Englesko školstvo je primjer na koji se treba ugledati, pomoću Scratcha i malih računala vizualno se učenicima prikazuje što se točno i na koji način programira. Naglasak se mora staviti na praksu, projekte i problemske zadatke, a ne na teoriju, definicije i učenje pojmova. Jer programiranje je ništa drugo no novi jezik kojim se izražava pojedinčeva kreativnost i inovativnost. A djeca zaslužuju biti upravo ti kreatori, a ne potrošači i puki korisnici tehnologije.

## LITERATURA

1. Anđelić, V. (2015). Digitalne kompetencije za nastavnike. Agencija za strukovno obrazovanje i obrazovanje odraslih, URL: [https://radovi2015.cuc.carnet.hr/modules/request.php?module=oc\\_program&action=view.php&a=&id=24&type=4](https://radovi2015.cuc.carnet.hr/modules/request.php?module=oc_program&action=view.php&a=&id=24&type=4) (pristupljeno: 27.6.2023.)
2. Babić Hajduković, D. (2018). The Analysis of the Relationship of Students' Towards Traditional and Contemporary Teaching Methods. *Pannoniana*, 2 (1-2), 177-191. Preuzeto s <https://hrcak.srce.hr/206860>
3. Bahč, T. (2023). Digitalne kompetencije u osnovnoj školi. *Varaždinski učitelj*, 6 (11), 325-328. Preuzeto s <https://hrcak.srce.hr/287440>
4. Čop, M. i Topolovec, V. (2009). Upotreba informacijske i komunikacijske tehnologije (ict) u obrazovanju djece s posebnim potrebama. *Informatologia*, 42 (4), 304-313. Preuzeto s <https://hrcak.srce.hr/42352>
5. Gugić, I., Seršić, S., Hrdka, S., Musser, E., Mirković, M., Bagarić, Z. (1997). Priručnik metodike za nastavu računalstva i informatike. Vinkovci: Pentium d.o.o.
6. Hutinski, Ž. i Aurer, B. (2009). Informacijska i komunikacijska tehnologija u obrazovanju: stanje i perspektive. *Informatologia*, 42 (4), 265-272. Preuzeto s <https://hrcak.srce.hr/42347>
7. Krelja Kurelović, M.s.E., Vasiljević, B. i Bodiš, S. (2013). Odgojna komponenta u nastavi informatike. *Metodički obzori*, 8(2013)1 (17), 5-13. Preuzeto s <https://hrcak.srce.hr/106245>
8. Lazić\_Lasić, J., Špiranec, S. i Banek Zorica, M. (2012). Izgubljeni u novim obrazovnim okruženjima – pronađeni u informacijskom opismenjivanju. *Medijska istraživanja*, 18 (1), 125-142. Preuzeto s <https://hrcak.srce.hr/85384>
9. Ministarstvo znanosti i obrazovanja (2018). Kurikulum nastavnog predmeta Informatike za osnovne škole i gimnazije u Republici Hrvatskoj. Preuzeto s: [https://narodne-novine.nn.hr/clanci/sluzbeni/2018\\_03\\_22\\_436.html](https://narodne-novine.nn.hr/clanci/sluzbeni/2018_03_22_436.html)
10. Mrđen, T., Livaja, I. i Acalin, J. (2019). Programski jezik Scratch - primjena u edukaciji. *Zbornik radova Veleučilišta u Šibeniku*, (3-4/2019), 115-123. Preuzeto s <https://hrcak.srce.hr/235564>

11. Nadrljanski, Đ. (2006). Informatička pismenost i informatizacija obrazovanja. *Informatologia*, 39 (4), 262-266. Preuzeto s <https://hrcak.srce.hr/9254>
12. Radošević, D. (1992). Informatika i uloga fakulteta organizacije i informatike u njenom razvoju. *Journal of Information and Organizational Sciences*, (16), 169-178. Preuzeto s <https://hrcak.srce.hr/79861>
13. Resnick, M. (2012). Let's teach kids to code. TED Talk, URL: [https://www.ted.com/talks/mitch\\_resnick\\_let\\_s\\_teach\\_kids\\_to\\_code](https://www.ted.com/talks/mitch_resnick_let_s_teach_kids_to_code) (pristupljeno: 27.05.2023.)
14. Scratch – službena stranica, URL: <https://scratch.mit.edu/> (pristupljeno: 4.6.2023)
15. Šimović, V., Maletić, F., Afrić, W. (2010). Osnove informatike – uvod. Zagreb: Golden marketing – Tehnička knjiga
16. Tutek, Ž. (2006). E-učenje - vežite se, polijećemo!. *Ekscentar*, (8), 120-121. Preuzeto s <https://hrcak.srce.hr/6188>
17. UNICEF (2017). Stanje djece u svijetu 2017: Djeca u digitalnom svijetu. New York: UNICEF, Odjel za komunikaciju
18. Vrkić Dimić, J. (2014). Suvremeni oblici pismenosti. *Školski vjesnik*, 63 (3), 381-394. Preuzeto s <https://hrcak.srce.hr/136084>
19. Vrkić Dimić, J. (2013). Kompetencije učenika i nastavnika za 21. stoljeće. *Acta Iadertina*, 10 (1), 0-0. Preuzeto s <https://hrcak.srce.hr/190113>
20. Department of Education (2019). What Do We Know about Computer Science Education? URL: <https://edstream.ed.gov/webcast/Play/8ef409fc273547d6b86149a05d0a4bf71d?catalog=82d9933c-1256-4cb2-8783-89599eb97fd8> (pristupljeno: 25.6.2023.)

## POPIS SLIKA

|                                                                                   |    |
|-----------------------------------------------------------------------------------|----|
| Slika 1: Odgojno-obrazovni ishodi prema ishodima i domenama za 2. razred OŠ ..... | 3  |
| Slika 2: Hour of Code - događanja u RH.....                                       | 8  |
| Slika 3: Hour of Code - primjeri projekata .....                                  | 9  |
| Slika 4: Hour of Code - certifikat .....                                          | 10 |
| Slika 5: Code Week tabela zemalja.....                                            | 11 |
| Slika 6: Run Marco - primjer zadatka.....                                         | 12 |
| Slika 7: Scratch - primjer jednostavnog koda.....                                 | 15 |
| Slika 8: Python – logo.....                                                       | 15 |
| Slika 9: Prikaz škola koje nude Computer Science .....                            | 19 |
| Slika 10: Programi kojima se potiče programiranje .....                           | 20 |
| Slika 11: Srednje škole i programiranje .....                                     | 21 |
| Slika 12: Raspberry Pi .....                                                      | 22 |

### **Izjava o samostalnoj izradi rada**

Izjavljujem da je moj diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onih koji su navedeni.

---