

# Didaktička načela u nastavi informatike

---

**Bursać, Mirko**

**Master's thesis / Diplomski rad**

**2019**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Pula / Sveučilište Jurja Dobrile u Puli**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:137:624662>

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

*Download date / Datum preuzimanja:* **2025-02-07**



*Repository / Repozitorij:*

[Digital Repository Juraj Dobrila University of Pula](#)



Sveučilište Jurja Dobrile u Puli  
Fakultet informatike

**MIRKO BURSAĆ**

**DIDAKTIČKA NAČELA U NASTAVI INFORMATIKE**

Diplomski rad

Pula, listopad, 2019.

Sveučilište Jurja Dobrile u Puli  
Fakultet informatike

**MIRKO BURSAC**

**DIDAKTIČKA NAČELA U NASTAVI INFORMATIKE**

Diplomski rad

**JMBAG: 0140003479, redoviti student**

**Studijski smjer: Nastavni smjer informatike**

**Predmet: Didaktika**

**Znanstveno područje: Društvene znanosti**

**Znanstveno polje: Pedagogija**

**Znanstvena grana: Didaktika**

**Mentori: doc.dr.sc. Marina Diković, doc.dr.sc. Siniša Sovilj**

Pula, listopad, 2019.



## IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani \_\_\_\_\_, kandidat za magistra \_\_\_\_\_ovime izjavljujem da je ovaj Diplomski rad rezultat isključivo mogega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Diplomskog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

\_\_\_\_\_

U Puli, \_\_\_\_\_, \_\_\_\_\_ godine



## IZJAVA

### o korištenju autorskog djela

Ja, \_\_\_\_\_ dajem odobrenje Sveučilištu  
Jurja Dobrile

u Puli, kao nositelju prava iskorištavanja, da moj diplomski rad pod nazivom

\_\_\_\_\_

\_\_\_\_\_ koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, \_\_\_\_\_ (datum)

Potpis

\_\_\_\_\_

## Sadržaj

Uvod.....	7
1. Osnovni pojmovi.....	8
1.1. Kurikulum.....	8
1.2. Nacionalni kurikulum.....	10
1.3. Školski kurikulum.....	12
1.4. Kognitivna taksonomija.....	12
1.5. Digitalna kompetencija.....	14
1.6. Didaktička načela.....	15
1.6.1. Načelo zornosti i apstraktnosti.....	16
1.6.2. Načelo sistematičnosti i postupnosti.....	17
1.6.3. Načelo individualizacije i socijalizacije.....	18
1.6.4. Načelo racionalizacije i ekonomičnosti.....	18
1.6.5. Načelo jednakosti šansi.....	19
1.6.6. Načelo aktivnosti.....	19
2. Analiza predmetnih kurikuluma.....	21
2.1. Nacionalni okvirni kurikulum Republike Hrvatske.....	21
2.2. Predmetni kurikulum Republike Hrvatske.....	21
2.2.1. Kurikulum od V. do VIII. razreda.....	23
2.2.2. Kurikulum prirodoslovno-matematičke gimnazije.....	27
2.3. Predmetni kurikulum Republike Slovenije za osnovnu školu.....	30
2.4. Predmetni kurikulum Velike Britanije.....	33
2.4.1. <i>Key stage</i> 3 i 4.....	33
2.4.1. <i>Key stage</i> 5.....	38
2.5. Didaktička načela u predmetnom kurikulumu Republike Hrvatske.....	38
2.6. Didaktička načela u predmetnom kurikulumu Republike Slovenije.....	40

2.7. Didaktička načela u predmetnom kurikulumu Velike Britanije .....	42
2.8. Poredba hrvatskog i slovenskog kurikuluma .....	44
2.9. Poredba hrvatskog i britanskog kurikuluma .....	45
3. Implementacija ishoda.....	47
3.1. Sadržaj kao putokaz do ishoda .....	47
3.1.1. Algoritam i algoritamski postupci .....	47
3.1.2. Varijable i tipovi podataka .....	48
3.1.3. Upisivanje i ispisivanje vrijednosti varijabli.....	49
3.1.4. Naredbe za kontrolu toka.....	50
3.2. Izazovi pri ostvarivanju ishoda .....	57
Zaključak .....	59
Literatura .....	60
Mrežni izvori .....	61
Sažetak .....	63
Ključne riječi .....	63
Abstract .....	64
Key words .....	64

## Uvod

Od kada postoji ljudska civilizacija čovjek je kroz znanje i spoznaje o svijetu pokušavao stvoriti alate koji bi mu omogućili olakšati svakodnevno funkcioniranje. U tom procesu traženja odgovora koji proizlazi iz vlastite prirodne znatiželje i željom za razvojem, stvoreno je kolektivno znanje koje nam omogućava opstanak, napredak, ali je i donijelo neke negativne posljedice. Tako su u razvojnem procesu čovječanstva stvorena mjesta i institucije koje su mogle preuzeti i implementirati stečene informacije o svijetu, te tako izgrađivati kolektivnu svijest i civilizaciju.

Kako bi primijenili cjelokupno postojeće ljudsko znanje, ono je moralo biti organizirano, strukturirano i prilagođeno čovjeku s obzirom na njegovu životnu dob, želje, potrebe i mogućnosti, ali i ograničenja. U povijesnom kontekstu takve institucije su poprimale različite nazive, a danas su takva mjesta odgojno-obrazovne ustanove: vrtići, škole, sveučilišta i sl. Tako smo dobili institucije za različite razine odgoja i obrazovanja gdje su se uz pomoću stručnjaka iz svih područja ljudskog znanja i stručnjaka vezani za odgoj, obrazovanje i razvoj čovjeka razvili određeni programi i okviri u kojima se znanje prenosi generacijama.

Kako opisati što se uči, kojim sadržajem se baviti, kako takav sadržaj organizirati i na koji način prenijeti onima kojima je to potrebno? Takav okvir za učenje nazvan je kurikulumom. U ovom radu ćemo se baviti jezgrom predmetnog kurikulumu - ishodima, a to su znanja, vještine i kompetencije koje treba učenik usvojiti nakon nekog nastavnog, odnosno odgojno-obrazovnog procesa, u ovom slučaju iz predmeta informatike. Navest ćemo osnovne pojmove koji se vežu za pojam kurikulumu, kako njegove teorije tako i prakse.

Nadalje, navest ćemo internacionalne kurikulume Republike Slovenije i Velike Britanije, opisati didaktička načela, ponajviše vezana za ishode, koja možemo pronaći u istima, te ih usporediti s kurikulumom Republike Hrvatske. Na posljetku ćemo navesti jedan ishod, kako ga ostvariti, te na koje sve izazove možemo naići pri ostvarivanju nekog ishoda.



# 1. Osnovni pojmovi

## 1.1. Kurikulum

Lingvistički, pojam kurikulum dolazi od latinskog *currere* ili *curo*, a znači trčati i slijediti, označava tijek, slijed, optimalan put djelovanja i dolaska do nekog cilja. Povijesno, svoje temelje ima u sedam antičkih slobodnih vještina, koje se dalje nastavljaju na sedam srednjovjekovnih viteških vještina. U 16. te 17. stoljeću pojavila se potreba učenja gradiva redosljedom po godištima. Wolfgang Ratke (1571. - 1635.) i Jan Amos Komensky (1592. - 1670.) po prvi puta spominju kurikulum u svojim djelima *Methodus didactica*, odnosno *Didactica magna*, a sredinom 20. stoljeća javlja se kao pojam koji donekle odgovara pojmu nastavnog plana i programa. U suvremenom smislu sam naziv pojma kurikulumu je preuzet iz američke teorije kurikulumu.

Snažan utjecaj na razvoj modernog pristupa učeniku, nastavi, školi, sadržaju i razvoju nacionalnog kurikulumu dao je Benjamin Samuel Bloom (1913. - 1999.) u prvoj knjizi o klasifikaciji i taksonomiji kognitivnog razvoja i ciljeva učenja *Taxonomy of Educational Objectives* (1956.). Tijekom 20. stoljeća kritizirala se isključiva usmjerenost na kognitivne ciljeve i ishode pa je njegova taksonomija kasnije proširena i na afektivne i psihomotoričke ciljeve razvoja.

Kroz 20. stoljeće postojale su tendencije odmicanja od ideološki utemeljenog, jednostranog, jednournog, konzervativnog i znanstvenog neutemeljenog kurikulumu, također radi različite filozofske tradicije europskog humanizma i idealizma nasuprot američkom pragmatizmu razvili su se različiti pogledi na kurikulum te su se iskristalizirala dva tipa koja sadržavaju različite pedagoške pozicije. Govorimo o humanističkom i funkcionalističkom kurikulumu. Humanistički je više orijentiran na razvoj i na samog učenika, dok je funkcionalistički orijentiran na proizvod. Iz prethodno navedenih kurikulumu razvijaju se tri strukturirane vrste: otvoreni, zatvoreni i mješoviti, gdje otvoreni odgovara humanističkom, zatvoreni funkcionalističkom, a mješoviti čini kombinaciju otvorenog i zatvorenog.

U kontekstu modernog i suvremenog shvaćanja „kurikulum suvremenog odgoja, obrazovanja i škole podrazumijeva znanstveno zasnivanje cilja, zadataka, sadržaja, plana i programa, organizaciju i tehnologiju provođenja te različite oblike

evaluacije učinaka“, odnosno „skup planiranih i implicitnih odrednica koje usmjeravaju odgojni i obrazovni proces prema zadacima i sadržajima koji su dosljedno izvedeni iz cilja te upućuju na organizacijske oblike i načine rada, postupke, provjere uspješnosti u zavisnosti od mnogobrojnih procesnih faktora i okolnosti“ (Previšić, 2007: 20) .

Kurikulum je razvojni dokument koji sadrži smjernice, ciljeve i ishode za učenike i studente koji su nužni za napredovanje od najnižih do najviših razina u procesu obrazovanja. Obuhvaća nekoliko procesa: planiranje, organizaciju, izvođenje i evaluaciju koji se izvode u kontekstu odgoja, obrazovanja, nastave i škole.

Takav dokument je okvir za učitelje, nastavnike, profesore, predavače i odgajatelje koji bi trebao služiti da u optimalnim uvjetima maksimiziraju potencijale svakog učenika s obzirom na njegove mogućnosti, sposobnosti, želje i potrebe za osobnim napretkom u kontekstu akademske nadogradnje, procesa učenja i cjeloživotnog razvoja. Dokument koji bi trebao dati određene kognitivne, emocionalne, motoričke alate kao i alate koje potiču kritičko mišljenje, logičko razmišljanje, cjeloživotno učenje i nove kompetencije i pismenosti, a učenike bi učinili odgovornijima, samostalnijima, kreativnijima i spremnima na suradnju.

Za stvaranje kurikulumu odgovornost imaju i znanost i struka. Kroz interdisciplinarni pristup strukturira se metodološki procesom kroz sadržaje, program i nastavnu tehnologiju i obuhvaća sve od predškolskog odgoja, preko osnovnoškolskog i srednjoškolskog te do viših razina obrazovanja.

Ono što je prema teoriji zajedničko svakom kurikulumu jeste (Previšić, 2007: 21):

1. potreba utvrđivanja što učenici moraju naučiti,
2. određivanje sadržaja kao izvora informacija,
3. utvrđivanje pedagoških standarda,
4. predviđanje različitih sposobnosti i tempa rada učenika,
5. organizacijski i metodički naputci,
6. stjecanje predviđene kompetencije,
7. razrađeni postupci vrednovanja, i
8. izbjegavanje definicija (korištenje ključnih pojmova, riječi, kategorija, modula).

## 1.2. Nacionalni kurikulum

Nacionalni kurikulum se sastoji od pet glavnih ciljeva koji usmjeravaju obrazovnu politiku (Previšić, 2007: 165):

1. Osigurati svakom djetetu odnosno učeniku, bez obzira na, spol, narodnost, socijalno podrijetlo, razlike u sposobnostima, odrastanje, obrazovanje i razvoj s jednakim pravim, uvjetima, mogućnostima i djetetovim, odnosno učenikovim sposobnostima.

2. Legalno i legitimno povjeravanje odgojno-obrazovne djelatnosti ili ovlašćivanje neposrednih čimbenika i sudionika u sustavu odgoja i obrazovanja, s pravima, dužnostima, odgovornostima (svih sudionika u tom procesu) učinkovito ostvarivanje prvog cilja.

3. Utvrđivanje odgojno-obrazovnih standarda koji osiguravaju djeci i učenicima jednake mogućnosti za optimalan razvoj i postizanje što viših rezultata u učenju i postignuća u školi.

4. Osiguravanje uvjeta za stalni razvoj nacionalnog okvira omogućujući fleksibilnost s obzirom na učenikovo napredovanje i njegove interese, osiguravanje lakšeg prijelaza iz jedne škole u drugu ili prijelaz s jednog stupnja obrazovanja na drugi, na nacionalnoj i internacionalnoj razini, osiguravanje uvjeta cjeloživotnog učenja.

5. Transparentnost prikazivanja djelovanja odgojno-obrazovnih ustanova javnosti te uključivanje javnosti u rad odgojno-obrazovnih ustanova, jačanje povjerenja javnosti u rad odgojno-obrazovnih ustanova i stjecanja obrazovanja.

Osnovne sastavnice kurikulumu su: odgojno-obrazovne vrijednosti, odgojno-obrazovni ciljevi, kurikulumski načela, odgojno-obrazovni sadržaj, organizacija, metode i načini rada te vrednovanje i samovrednovanje. One se temelje na načelima konzistentnosti i koherentnosti, a izražuju se u dokument koji zovemo *Okvir nacionalnog kurikulumu*. Donosi ga nadležno tijelo - ministarstvo. Takvim pristupom daje kriterije i razinu kvalitete, a sadrži smjernice, uvjete i načine ostvarenja vezani za formalni odgoj i obrazovanje djece i mladih.

Pridržavanjem zadanog okvira, kao smjernica ili uputa u odgojno-obrazovnom procesu, štiti se i sam kurikulum od nekvalitetnih improvizacija, nestručnih djelovanja na jezgrovne cjeline nekog predmeta ili nastavnog udžbenika te medijskih utjecaja, a

služi kao regulacijski mehanizam u ostvarenju nacionalnih i internacionalnih ciljeva u odgojno-obrazovnom kontekstu.

Pored osnovnih sastavnica kurikulum čine i: izborni i fakultativni programi, suradnja s drugim obrazovnim čimbenicima (obitelj, lokalna zajednica, ustanove za obrazovanje), profesionalno informiranje, programi za djecu i učenike s posebnim potrebama, programi za djecu i učenike nacionalnih manjina, certificiranje, načini informiranja, plan razvoja prema rezultatima vanjskog vrednovanja i samovrednovanja.

Što sve utječe na stvaranje nacionalnog kurikulum? Tradicija i nacionalno-simbolički sadržaji, kao što su jezik, religija, povijest, teritorij, kulturna baština, stupanj znanstvenih dostignuća, stanje društvene svijesti, moral, pedagoške sposobnosti, stavovi, ponašanja. Nadalje: natjecateljstvo, solidarnost, menadžment, društveno-političke i gospodarske politike, multinacionalni biznis, interkulturalni odnosi, ljudska prava, aktualne promjene, suvremene tehnike i tehnologije, nove pismenosti i vrijednosti te globalizacija. To mogu biti neke vanjske ili internacionalne deklaracije i konvencije (*Europska konvencija o ljudskim pravima, Opća deklaracija o pravima čovjeka, Konvencija o pravima djeteta*) kao i neke univerzalne vrijednosti kao sloboda, jednakost, jednakopravnost, mir, društvena pravda, tolerancija, antidiskriminacija, inkluzivnost).

Ako dobro postavimo kurikulum u svojim temeljima posljedice mogu biti značajne i ne samo na području teorije već i prakse, a to su: (Previšić, 2007: 12):

- a) optimalno opterećenje učenika i nastavnika,
- b) aktualizacija i preferencijalno biranje sadržaja, načina i rada,
- c) mogućnost transdisciplinarnih odnosa u znanju učenika,
- d) učinkovita znanstvena, sadržajna i pedagoška korelacija,
- e) jasna očekivanja razina usvojenih znanja od strane nastavnika i roditelja,
- f) promijenjeni socijalni odnosi među glavnim protagonistima,
- g) jasno stjecanje općih i posebnih kompetencija, i
- h) mogućnost primjene unutarnje i vanjske evaluacije.

### **1.3. Školski kurikulum**

Zašto spominjati školski kurikulum u ovom kontekstu? Jer on zapravo proizlazi iz nacionalnog kurikuluma i služi kao dodatak nacionalnom kurikulumu, u užem smislu, dok u širem smislu stavlja školu u središte odlučivanja o tome koje su to obrazovne potrebe učenika i lokalne sredine, dakle daje veću autonomiju školi u stvaranju kurikuluma škole. Zapravo, tamo gdje nacionalni kurikulum služi kao okvir na razini države, tu je školski kurikulum koji stavlja učenika, njegove interese i njegovu sredinu u središte odgojno-obrazovnog procesa.

Školski kurikulum isto tako i prokazuje jedinstveni identitet i profil pojedine škole te ukupnost aktivnosti koje su planirane za učenike. Sastoji se od toga koje će izborne predmete moći odabrati učenik kako bi dijelom upravljao odabirom predmetnog sadržaja kojim se želi baviti tijekom školovanja, dopunska nastava za učenike koji sporije dolaze do željenih ishoda, dodatna nastava za učenike koji žele znati više te se još više usmjeriti na željeno područje kao i izvannastavne te izvanškolske aktivnosti.

### **1.4. Kognitivna taksonomija**

Iako razlikujemo više vrsta kurikuluma, bit ćemo usmjereni na nacionalne i predmetne kurikulume iz područja nastave informatike. Nacionalni kurikulum kao temeljni dokument se formira kao okvir u kojem nastavnici mogu djelovati u ostvarivanju nastave kroz školski i predmetni kurikulum, dok je predmetni kurikulum usmjeren na samo jedno područje, odnosno predmet poučavanja. Svaki kurikulum sadrži opise ciljeva učenja, sadržaj poučavanja, nastavni stil rada učitelja, nastavne metode i oblike poučavanja, nastavna sredstva i pomagala, odgojno-obrazovne ishode i vrednovanje postignutih ishoda, ali nije nužno sve navedeno u samom kurikulumu, jer se nacionalni kurikulumi različitih država razlikuju, pa se tako razlikuju i predmetni kurikulumi.

Učenička postignuća ili odgojno-obrazovni ciljevi predstavljaju očekivana znanja, vještine i sposobnosti, vrijednosti i stavove koje učenici trebaju steći te ih moći pokazati nakon uspješnog završetka određene nastavne teme, programa, stupnja obrazovanja ili odgojno-obrazovnog ciklusa. Ciljevi su isključivo usmjereni na učenike i njihove aktivnosti koje iskazujemo aktivnim glagolima. Nas najviše zanima

postoji li pomak u učenikom razvoju, kroz stjecanje znanja i savladavanja gradiva, nakon što su prošli određeni predmetni program. S toga će nas zanimati i usvojeni ishodi koji napreduju s obzirom na razinu.

Sami ciljevi se odnose na tri domene: kognitivnu, emocionalnu i psihomotoričku. Razvojni model koji je uključivao samo kognitivnu domenu prvi je uveo Benjamin Samuel Bloom, 1956. godine.

Bloomovom taksonomijom možemo prikazati napredak znanja kroz ishode. Ako znamo i razumijemo osnovne koncepte i pojmove, onda kroz primjenu možemo naučiti analizirati rezultate, potom ih povezati i vrednovati. Npr. ako učenik savlada osnovne pojmove i koncepte iz programiranja, moći se samostalno osmisliti algoritam ili program, usporediti koji je najbolji način za riješiti određeni problem, na kraju ocijeniti je li takav program bio učinkovit, koliko je bio brz i računa li ono za što je bio namijenjen i tako odabrati najbolji i najupotrebljiviji algoritam. Navest ćemo revidiranu kognitivnu domenu<sup>1</sup> (tablica 1.1.).

tablica 1.1. Kognitivna domena

Kognitivna razina	Opis razine	Aktivni glagoli
<b>1. ZNANJE</b>	Reprodukcija ili prepoznavanje informacija, ideja, koncepata i načela	Definirati, opisati, identificirati, prepoznati, označiti, nabrojati, povezati, imenovati, ponoviti, reproducirati, izreći, odabrati, navesti, iskazati, poredati, sjetiti se zapamtiti
<b>2. RAZUMIJEVANJE</b>	Objašnjavanje ili interpretacija informacija	Opisati, objasniti, raspraviti, dati primjer, grupirati, svrstati, klasificirati, pretvoriti, obraniti, razlikovati, izdvojiti, procijeniti, izvesti, zaključiti, predvidjeti, sažeti, prevesti, preformulirati, smjestiti, pokazati

<sup>1</sup> revidirana verzija se nalazi u Anderson, L. W., Krathwohl, D. R., [et al] (2001.) *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. Allyn & Bacon. Boston: MA (Pearson Education Group)

<b>3. PRIMJENA</b>	Odabiranje i upotrebljavanje naučenih koncepata, načela, teorija i metoda u rješavanju problema ili zadataka u konkretnoj ili novoj situaciji	Primijeniti, izračunati, odabrati, prilagoditi, riješiti, otkriti, demonstrirati, pokazati, baratati, pripremiti, rabiti, koristiti, upotrijebiti, proizvesti, povezati, ilustrirati, skicirati
<b>4. ANALIZA</b>	Raščlanjivanje materijala na osnovne dijelove kako bi se mogla razumjeti organizacijska struktura	Analizirati, raščlaniti, skicirati, razlikovati, izdvojiti, identificirati, prikazati, ukazati na, usporediti, staviti u odnos s, klasificirati, sortirati, sučeliti, suprotstaviti, proračunati, ispitati, istražiti, eksperimentirati, provjeriti
<b>5. VREDNOVANJE</b>	Sposobnost prosuđivanja vrijednosti materijala u skladu s određenim kriterijima	Utvrđiti, procijeniti, predvidjeti, vrednovati, ocijeniti, prosuditi, usporediti, zaključiti, interpretirati, suprotstaviti, kritizirati, opravdavati, odabrati, podržati, preporučiti, argumentirati, potvrditi
<b>6. SINTEZA</b>	Povezivanje osnovnih dijelova u novu funkcionalnu cjelinu.	Urediti, povezati, integrirati, složiti, kreirati, stvoriti, razviti, kombinirati, prikupiti, sakupiti, dizajnirati, generirati, modificirati, organizirati, planirati, preurediti, uskladiti, napisati, predložiti, osmisлити, konstruirati, revidirati, rekonstruirati, formulirati

### 1.5. Digitalna kompetencija

Kako bi se ostvario kurikulum, u idealiziranom smislu, onaj koji ga izvršava mora posjedovati određene karakteristike, određene kompetencije. Kako bi se održala određena okvirna razina potrebni su određeni kriteriji po kojima bi trebali vrednovati takve kompetencije. Tako je 2006. Europska komisija došla do određenog kompetencijskog okvira koji bi optimizirao akademski, profesionalni, poslovni i osobni razvoj europskog građanina. Takav okvir sastoji se od osam glavnih kompetencija

koje bi trebao posjedovati europski građanin<sup>2</sup>: Komunikacija na materinskom jeziku, komunikacija na stranom jeziku, matematička i prirodoslovna pismenost, digitalne kompetencije, naučiti kako učiti, međuljudske i građanske kompetencije, smisao za inicijativu i poduzetništvo, kulturna osviještenost i kulturološko izražavanje.

Okvir za digitalne kompetencije<sup>3</sup> je 2013. objavila Europska komisija, a s obzirom na to da su usko vezane i za sadržaj nastave informatike navest ćemo domene i kratak opis:

1. Informacijska i podatkovna pismenost; Treba: artikulirati potrebne informacije, pronaći i izvući digitalni podatak, informaciju i sadržaj i vrednovati relevantnost izvora i njegov sadržaj.

2. Komunikacija i suradnja; Treba: sudjelovati, komunicirati i surađivati putem digitalnih tehnologija uz svjesnost, kulturalne i dobne raznolikosti, društveno djelovati kroz javne i privatne digitalne servise i suradničko građanstvo, održavati digitalni identitet i ugled.

3. Stvaranje digitalnog sadržaja; Treba: stvarati i mijenjati digitalni sadržaj radi poboljšanja i integriranja informacije i sadržaja u postojeća znanja uz razumijevanje kako se primjenjuju autorska prava i licence, znati kako dati razumljive instrukcije za računalni sustav.

4. Sigurnost; Treba: zaštititi uređaje, sadržaje, osobne podatke i privatnost u digitalnom okruženju, zaštititi fizičko i psihološko zdravlje i biti svjestan digitalnih tehnologija za društvenu dobrobit i inkluziju, biti svjestan utjecaja digitalnih tehnologija na okoliš.

5. Rješavanje problema; Treba: identificirati potrebe i probleme te rješavanje konceptualnih problema i problemskih situacija u digitalnom okruženju, koristiti digitalni alat za inovaciju procesa i produkta, biti u koraku s digitalnom evolucijom.

## **1.6. Didaktička načela**

Načelo je početak ili osnova od koje se polazi i koju valja uvažavati u različitim područjima ljudske djelatnosti. Prema Lavrnji (1996: 50): „Usmjerava nečiju

---

<sup>2</sup> (2006.) *European reference framework.*

<sup>3</sup> (2013.) *Digital competence framework.*



aktivnost, određene kriterije odlučivanja koji proizlaze iz zakonitosti procesa nekih djelatnosti, rezultat spoznajnih procesa određene djelatnosti. Odražava zakonite odnose i obuhvaća postavljene ciljeve konkretne djelatnosti”.

U kontekstu nastave i učenja takva načela su didaktička načela. Slijede iz spoznaja o zakonitostima odgojno-obrazovnoga procesa, prakse nastave i učenja, a odnose se na cjelokupni nastavni proces. Neka od načela su usmjereni na nastavni proces (kako je sadržaj organiziran u nastavi, struktura nastave, vremenska organizacija, optimizacija i maksimizacija učinkovitosti učenja i poučavanja), neka na sadržaj (kako ga logički organizirati, linearni napredak od najnižih do najviših kognitivnih, afektivnih i psihomotoričkih razina), neka na proces učenja i usvajanja informacija (kako učiti), a neka na samog učenika (socijalni aspekti učenja i poučavanja, pristup učenju i sudjelovanje u procesu učenja i poučavanja). Postoje različite podjele, ali mi ćemo spomenuti načela: zornosti i apstraktnosti, sistematičnosti i postupnosti, individualizacije i socijalizacije, racionalizacije i ekonomičnosti, jednakosti šansi te aktivnosti. U opisima će biti jasnije zašto neka načela idu u paru.

### **1.6.1. Načelo zornosti i apstraktnosti**

Načelo zornosti odnosi se na usvajanje činjenica, a načelo apstraktnosti na usvajanje generalizacija. Zornost omogućava “cjelovito osjetno doživljavanje svim osjetilnim organima, odnosno omogućava učenicima da tijekom nastave osjetilnim organima neposredno zahvaćaju objektivnu stvarnost koja se u nastavi proučava“ (Poljak, 1988: 201).

Osjetno doživljavanje pojačava psihičke funkcije: pamćenje, maštu, emocionalno doživljavanje i sl. Koristi se kod usvajanja novih znanja (usvajanje činjenica), posebice kod učenika mlađe školske dobi. Potrebno je onoliko koliko je učenicima dovoljno da usvoje činjenice i znanja koja će biti osnova za kasnije apstrahiranje i generalizaciju. Ovo načelo se temelji na induktivnom mišljenju, gdje se kreće od pojedinačnog k općem.

Zornost je načelo koje je preduvjet za apstraktnost i podržava proces apstrahiranja, zato i na temelju usvojenih činjenica učenike treba kognitivnim poticajima dovesti do generalizacije, tj. do formiranja određenih pojmova, zakona,

načela, pravila, kategorija, jednadžbi, simbola, zaključaka, teorija i sl. Zornost treba biti spoznajno i psihološki orijentirana prema izvođenju generalizacija. Osim naglašavanja povezanosti zornosti i apstraktnosti radi usvajanja činjenica i generalizacija, treba naglasiti i ona načela zornosti koja vode do određenih apstrakcija.

### **1.6.2. Načelo sistematičnosti i postupnosti**

Načelo sistematičnosti se odnosi na nastavni sadržaj koji je strukturiran logičkim redoslijedom, mora biti koherentan, a prethodi mu planiranje. Iz kolekcije naučenih činjenica i generalizacija trebaju se izdvojiti određeni sadržajni elementi kao uporišta, kako bi se ta uporišta rasporedila u logički red te da bi se ostali sadržajni elementi koncentrirali oko odgovarajućih uporišta. „Kriteriji za izdvajanje uporišta mogu biti različiti, što ovisi o unutarnjoj strukturi sadržaja i njezinoj logičnosti“ (Poljak, 1988: 206).

Kriteriji sistematiziranja se mogu mijenjati tako da se za obrađivanje novih sadržaja koristi jedna logička struktura, a za ponavljanje toga istoga sadržaja neka druga struktura. Tako učenici mogu samostalno uređivati nastavne sadržaje, a ne samo usvajati gotov sustav iz određenoga izvora znanja. Što je broj činjenica i generalizacija veći to je i veća potreba za logičkim uređivanjem tih sadržaja. Bez takvog logičkog sređivanja nastavni bi sadržaji ostali neuređeni i nagomilani u svijesti. Ovim načinom se vježba i stimulira praksa stvaranja kognitivnih mapa, složenih vizualnih mapa znanja i pojmova što olakšava učenicima lakše pretraživanje pamćenja.

Struktura sama po sebi ništa ne znači, ako ne postoji neka povezanost između podstruktura, tako se načelom postupnosti pokazuje da „učenici ne mogu odmah usvajati znanstvene sustave u njihovoj dubini i širini, nego ih do tih sustava treba dovesti postupnim radom“ (Poljak, 1988: 207). Taj postupan rad nastavnika usmjerava od lakšeg k težem, od jednostavnog k složenom, od bližeg k nepoznatom i od konkretnog k apstraktnom. Načelom postupnosti se povezuje strukturirani sadržaj, a nastavni sadržaj treba biti strukturiran tako da odgovara određenoj razini razumijevanja i složenosti sadržaja koje učenik može uz optimalni trud i napor

savladati i kretati se unaprijed prema daljnjim razinama povezujući razine napredovanja.

### **1.6.3. Načelo individualizacije i socijalizacije**

Individualne razlike postoje u svakom razredu. „Kako bi se individualne psihofizičke snage učenika razvile do maksimuma tada se individualizacija provodi na različite načine, a najlakše se ostvaruje u dopunskoj, dodatnoj i izvannastavnoj nastavi“ (Poljak, 1988: 212).

U redovitoj nastavi se najlakše provodi individualiziranim pristupom učenika kroz diferencirane zadatke. Također se postiže radom u skupinama ili radom u timu. U frontalnom radu je najzahtjevnije provesti individualizaciju jer se učitelj ne može svim učenicima jednako posvetiti. Individualizacija nastave postiže „se diferencijacijom nastave u sadržaju i načinu rada do te mjere da se zadovolje individualne razlike i potrebe učenika“ (Poljak, 1988: 213). Nastava bez individualizacije povećava individualne razlike među učenicima.

Svaki učenik se nalazi u socijalnoj skupini, razredu, pa kao takvog učenika promatramo u kontekstu tog kolektiva. Tako dolazimo do načela socijalizacije. Učenik u razredu razvija interpersonalne odnose s ostalim učenicima, pogotovo suradničke odnose u odgojno-obrazovnom procesu učenja i nastave. Načelo socijalizacije je usmjereno na sprječavanje konkurencije, a afirmira vrijednosti tolerancije i suradnje u nastavi i učenju. Ovim načelom učenik se integrira u razred, a da se pritom ne zanemaruju njegove individualne karakteristike.

### **1.6.4. Načelo racionalizacije i ekonomičnosti**

Cilj ovoga načela je postići najveći mogući učinak sa što manjim utroškom vremena, sredstava i snaga, pritom da nije smanjena obrazovna i odgojna vrijednost. „U ostvarivanju ekonomičnosti treba imati na umu da svaki nastavni postupak zahtijeva određeni optimalni utrošak vremena“ (Poljak, 1988: 215). Ono što omogućava ekonomičnost je dobro i pravilno osmišljeni izvedbeni nastavni plan i program.

Racionalizirati nastavu znači provesti racionalne promjene u nastavnim postupcima radi postizanja kvalitetnijeg rezultata i većeg učinka u propisanom

nastavnom vremenu, čime povećava ekonomičnost nastavnoga rada. Tu je racionalizaciju nastave potrebno provoditi u svim dijelovima nastavnoga procesa: „racionalizacijom izvora znanja (izbor adekvatnih izvora znanja), organizacijskom strukturom nastavnoga sata, primjenom efikasnih nastavnih metoda i strategija, stvaranjem povoljnijega razrednoga ozračja itd.“ (Poljak, 1988: 215). Pretpostavka za racionalizaciju nastave je sposobnost učitelja za mogućnost pronalaženja originalnih rješenja u načinu izvođenja nastave. Potreba za poznavanje nastavnoga procesa je preduvjet za racionalizaciju i ekonomičnost, pogotovo u onim dijelovima koji se odnose na učenika.

Uz ovo načelo se može povezati i načelo evaluacije. Eksplicitno se misli na učenikovo davanje povratne informacije o napretku - samoevaluaciji, što može uvelike olakšati samu organizaciju, tempo i smjer samog nastavnog procesa.

#### **1.6.5. Načelo jednakosti šansi**

Ovo načelo je temelj svake demokratski uređene društvene zajednice. „Nastava i učenje se tako organiziraju da svaki učenik u optimalno mogućim uvjetima postiže svoj maksimum“ (Lavrnja, 1996: 53). Ostvarivanje ovoga načela razmatramo u sklopu dvaju aspekata, u aspektu (Lavrnja, 1996: 54) „mogućnosti individualizacije učenja i nastave u kojemu učenik dobiva mogućnost svojega osobnoga uspjeha i društvene pretpostavke koja opravdava jednakosti šansi“.

Dakle, odgojno-obrazovni proces i aktivnosti moraju biti dostupni učenicima bez obzira na njihove individualne karakteristike. Ovo načelo je sadržano i povezano s načelima individualizacije i socijalizacije, jer se upravo kroz ova dva načela pruža svima jednaka šansa za razvoj i napredak u odgojno-obrazovnom procesu.

#### **1.6.6. Načelo aktivnosti**

Ovo načelo se ogleda u „samostalnom djelovanju učenika u odnosu na cjelokupan proces nastave i učenja“ (Lavrnja, 1996: 61). Ne obuhvaća samo intelektualne aktivnosti, već i druge dimenzije procesa učenja. Samostalnom djelatnošću u procesu učenja učenika stavljamo u situaciju da „otkriva istine“, a ne da mu namećemo gotove sadržaje. Aktivno učenje dovodi do razumijevanja, shvaćanja i prihvaćanja što pospješuje trajnije zadržavanje naučenoga uz

moćnost daljnega samostalnoga ućenja i otkrivanja. Aktivnost i svjesno djelovanje dovode do samostalnosti te uće ućenike odgovornosti i odgovornom djelovanju prema svom razvoju. Razmatranje načela aktivnosti treba sagledati i s aspekta aktivnosti učitelja – poučavanja i s aspekta ućenika – ućenja. Poučavanje i ućenje su povezani procesi (neki put i istovremeni) pa tako i aktivnost učitelja, koja potiče i organizira proces ućenja, utječe na ućenikovu aktivnost, pa i na njegovu samoaktivnost i samodjelatnost.

Ućenje je uvijek individualni čin, nitko ne može učiti umjesto ućenika, ali mu učitelj može pomoći i tako olakšati proces ućenja i usvajanja novih znanja i pomoći u racionalizaciji procesa ućenja. Pod aktivnošću ućenika se podrazumijeva djelovanje u svim etapama nastavnoga procesa. Aktivnost ućenika je zapravo tako povezana i za evaluaciju postignutoga ishoda, ali to podrazumijeva i samoevaluaciju ućenika, evaluaciju razrednih kolega te evaluaciju ućenika od strane nastavnika.

Neki put se uz ovo načelo spominje i načelo svjesnog djelovanja, a ono se odnosi prema Marius-Costel (2010: 26) na: „jasno i dublje značenje“ te „povezivanje i stvaranje konceptualno-teorijske korelacije.“ Navedeni autor spominje i da motivacija mora igrati bitnu ulogu u ohrabivanju ućenika radi razvijanja samostalnosti te da je srž ovog načela, zajedno s aktivnosti, razumijevanje nastavnog sadržaja. Pogotovo intrinzična motivacija, koja pojačava usmjerenost da se znanja iz teorije primjene u praksi, tj. određenoj aktivnosti.

## **2. Analiza predmetnih kurikuluma**

### **2.1. Nacionalni okvirni kurikulum Republike Hrvatske<sup>4</sup>**

Nacionalni okvirni kurikulum je dokument iz kojeg proizlazi predmetni kurikulum. Službena verzija je prihvaćena 2011. godine. Informatika je područje koje obuhvaća informacijske znanosti i računarstvo, a nacionalni kurikulum je kao temeljni dokument usmjeren na kompetencije koje učenik treba razviti ovim predmetom. Informatika se nalazi u tehničkom i informatičkom području, a napredak znanja je podijeljen na cikluse.

Odgojno-obrazovni proces je podijeljen na četiri ciklusa. Prvi ciklus obuhvaća osnovnu školu od I. do IV. razreda, drugi ciklus osnovnu školu V. i VI. razreda, treći ciklus zadnja dva razreda osnovne škole, VII. i VIII. te četvrti ciklus se odnosi na srednjoškolske razrede od I. do IV. ovisno o potrebama, tj. radi li se o gimnazijskim ili strukovnim programima. U ovom radu ćemo se baviti drugim, trećim i četvrtim ciklusom. Nas zanimaju koja su to informatička znanja koja učenici trebaju usvojiti kako bi postali samostalni, osposobljeni za rad i cjeloživotno učenje.

Imamo zadatak, imamo početni problem. Kako doći do rješenja ili rezultata? Koja znanja i metode upotrijebiti da bi se došlo do tog cilja. To nam omogućuju odgojno-obrazovni ciljevi koji su operacionalizirane vrijednosti koje se iskazuju na obrazovnim razinama, a povezani su s očekivanim rezultatima sa stupnjevima postignuća koje zovemo ishodom. Očekivani ishodi učenika u nastavi informatike su podijeljeni po područjima unutar samog predmeta. U ovom dijelu ulazimo u kurikulume osnovne škole od V. do VIII. razreda, te srednjoškolski program prirodoslovno-matematičke gimnazije.

### **2.2. Predmetni kurikulum Republike Hrvatske**

Ovdje ćemo navesti novi prijedlog predmetnog kurikuluma iz informatike za osnovnu i srednju školu koji kronološki slijedi nakon nacionalnog i navedenih predmetnih kurikuluma. Ovaj prijedlog je stavljen u javnu raspravu 2016. godine i

---

<sup>4</sup> (2011.) *Nacionalni okvirni kurikulum*. Zagreb: Ministarstvo obrazovanja i športa.

prihvaćen je 2018. kao službeni dokument. U škole ulazi 2019. godine. Kao obvezni predmet se pojavljuje u V. i VI. razredu, dok je do kraja osnovne škole izborni, kao i u srednjoškolskom obrazovanju. Cjeloviti kurikulum obuhvaća četiri domene iz predmeta informatike, a to su: informacije i digitalna tehnologija, računalno razmišljanje i programiranje, digitalna pismenost i komunikacija te e-društvo<sup>5</sup>. Opisi domena slijede:

a) Informacije i digitalna tehnologija

Ova domena obuhvaća sadržaj računalnih podataka, a uključuje različite radnje koje se mogu izvršavati s podacima: pretraživanje, dohvaćanje, pohranjivanje, manipulaciju, prenošenje, brisanje, kopiranje i sl. Različiti podatci mogu poprimiti različite načine digitalnog prikazivanja, to mogu biti brojevi, tekst, zvuk, video i slike. Između podataka se mogu uočiti razni odnosi, kao npr. različiti tipovi zvukovnih zapisa. Isti ti podatci se mogu različito mijenjati i prezentirati putem digitalne tehnologije i programa za obradu i predstavljanje podataka. Mogu se modelirati nove strukture podataka, te vizualizacijom i simulacijom utjecati na razvoj apstraktnog mišljenja.

b) Računalno razmišljanje i programiranje

Kada govorimo o računalnom razmišljanju mislimo na način razmišljanja koji je primjenjiv ne samo na računalu, već u različitim područjima stvarnoga života. Uključuje različite aspekte: logičko zaključivanje, modeliranje, apstrahiranje, rješavanje problema, razvijanje metakognitivnih vještina, razlaganje složenog problema na više jednostavnijih, kreativnost, inovativnost, poduzetnost, preuzimanje inicijative, razvoj komunikacijskih vještina, timski rad. Razvijanjem računalnog razmišljanja postaje se ne samo korisnik, već i samostalno i samopouzdana se stvara računalni alat programiranjem što omogućava stvaranje novih resursa koji utječu na mijenjanje svijeta.

---

<sup>5</sup> (2018.) *Kurikulum za nastavni predmet informatike za osnovne škole i gimnazije*. Zagreb: Ministarstvo obrazovanja i športa.

### c) Digitalna pismenost i komunikacija

Kao i prethodna domena, učenici bi trebali razvijati kritičko mišljenje i kreativnost stvarajući različite digitalne artefakte. Razlikuju pojmove hardvera i softvera i znaju kako funkcionira digitalno društvo. Upoznaju se s različitim tipovima digitalne tehnologije, kao i programima, kako aktualnih tako i onih koji bi se mogli koristiti u budućnosti. Samostalno ili u timu mogu prezentirati na koji način tehnologija utječe na naš život i kako ga učiniti boljim razmjenjujući ideje, stavove i različite vrijednosti. Tako razvijaju komunikacijske i suradničke vještine.

### d) e-Društvo

Učenici će ovom domenom učiti kako koristiti mrežne alate interneta. Pratit će novosti vrednovanjem različitih izvora informacija. Naučit će kako zaštititi svoj identitet - osobne podatke, kako prepoznati digitalne *online* prevare, koji su to neprikladni sadržaji, neprikladni oblici ponašanja, različite oblike elektroničkog nasilja i kome se obratiti za pomoć ako do istog dođe. Ova domena potiče razvoj aktivnog sudjelovanja u demokraciji, gdje će učenici koristiti različite e-usluge. Primjenjivat će različita ergonomska načela te će se naposljetku brinuti za svoj digitalni ugled, zaštitu *online* privatnosti i okoliš.

## 2.2.1. Kurikulum od V. do VIII. razreda

Nadalje ćemo navesti koji su to ishodi iz predmetnog kurikuluma informatike za osnovnu školu po razredima krećući se od V. razreda na dalje. Ishodi za V. razred osnovne škole su sljedeći:

### 1. Informacije i digitalna tehnologija

Učenik: pronalazi i vrednuje informacije; istražuje glavne komponente uobičajenih digitalnih sustava, određuje osnovne funkcije i veze s drugima, istražuje kako se takvi sustavi mogu povezivati mrežom i kako razmjenjivati podatke; analizira način na koji računalo pohranjuje sve vrste podataka.



## 2. Računalno razmišljanje i programiranje

Učenik: koristi se programskim alatom za stvaranje programa u kojemu se koristi ulaznim i izlaznim vrijednostima te ponavljanjem; stvara algoritam za rješavanje jednostavnoga zadatka, provjerava ispravnost algoritma, otkriva i popravlja pogreške.

## 3. Digitalna pismenost i komunikacija

Učenik: prilagođava korisničko sučelje operacijskog sustava svojim potrebama, samostalno otkriva i pokazuje dodatne mogućnosti operacijskog sustava; koristi se mogućnostima sustava za pohranjivanje i organizaciju datoteka; osmišljava plan izrade digitalnog rada, izrađuje ga, pohranjuje u mapu digitalnih radova (e-portafolio) i vrednuje ga; upotrebljava multimedijske programe za ostvarivanje složenijih ideja u komunikacijskome ili suradničkom okruženju.

## 4. e-Društvo

Učenik: analizira etička pitanja koja proizlaze iz korištenja računalnom tehnologijom; argumentira i procjenjuje važnost zbrinjavanja električkog otpada te objašnjava postupke njegova zbrinjavanja.

Ishodi za VI. razred su:

### 1. Informacije i digitalna tehnologija

Učenik: planira i stvara vlastite hijerarhijske organizacije te analizira organizaciju na računalnim i mrežnim mjestima; opisuje načine povezivanja uređaja u mrežu, analizira prednosti i nedostatke mrežnoga povezivanja te odabire i primjenjuje postupke za zaštitu na mreži.

### 2. Računalno razmišljanje i programiranje

Učenik: stvara, prati i preuređuje programe koji sadrže strukture grananja i uvjetnoga ponavljanja te predviđa ponašanje jednostavnih algoritama koji mogu biti prikazani dijagramom, riječima govornoga jezika ili programskim jezikom; razmatra i rješava složeniji problem rastavlajući ga na niz potproblema.

### 3. Digitalna pismenost i komunikacija

Učenik: izrađuje, objavljuje te predstavlja digitalne sadržaje s pomoću nekoga *online* i/ili *offline* programa pri čemu poštuje uvjete korištenja programom te postavke privatnosti; koristi se *online* pohranom podataka i primjerenim programima kao potporom u učenju i istraživanju te suradnji; surađuje s drugim učenicima u stvaranju *online* sadržaja.

### 4. e-Društvo

Učenik: objašnjava ulogu i važnost digitalnih tragova, stvara svoje pozitivne digitalne tragove; prepoznaje vrste elektroničkoga nasilja, analizira ih i odabire preventivne načine djelovanja na različite slučajeve elektroničkog nasilja; pronalazi mrežne zajednice učenja koje su od osobnog interesa i pridružuje im se (online kolegij, grupe i sl.).

Ishodi za VII. razred su:

#### 1. Informacije i digitalna tehnologija

Učenik: prepoznaje i opisuje ulogu glavnih komponenti računalnih mreža, istražuje kako obilježja strojne opreme utječu na mrežne aktivnosti, koristi se zajedničkim dijeljenjem resursa na mreži; primjenjuje strategije za prepoznavanje i rješavanje rutinskih hardverskih/softverskih problema do kojih može doći tijekom uporabe računalne tehnologije; prikuplja i unosi podatke kojima se analizira neki problem s pomoću odgovarajućega programa, otkriva odnos među podacima koristeći se različitim alatima programa te mogućnostima prikazivanja podataka; opisuje, uspoređuje i koristi se različitim formatima zapisivanja grafičkih i zvučnih podataka te videopodataka na računalu.

#### 2. Računalno razmišljanje i programiranje

Učenik: razvija algoritme za rješavanje različitih problema koristeći se nekim programskim jezikom pri čemu se koristi prikladnim strukturama i tipovima podataka; primjenjuje algoritam (sekvencijalnog) pretraživanja pri rješavanju problema; dizajnira i izrađuje modularne programe koji sadrže potprograme u programskom jeziku; koristi se simulacijom pri rješavanju nekoga, ne nužno računalnog problema.

### 3. Digitalna pismenost i komunikacija

Učenik: koristi i upoznaje se s različitim platformama i programima koje prema potrebi pronalazi i instalira; priprema, izrađuje te objavljuje vlastite mrežne stranice u skladu s dobrom praksom u području intelektualnoga vlasništva, kritički prosuđuje dobra i loša obilježja pojedinih mrežnih sadržaja.

### 4. e-Društvo

Učenik: štiti svoj elektronički identitet i primjenjuje pravila za povećanje sigurnosti korisničkih računa; demonstrira i argumentirano opisuje primjere dobrih strana dijeljenja informacija na internetu i njihova brzog širenja te primjenjuje pravila odgovornoga ponašanja; analizira proces suradnje među članovima virtualnih zajednica te njezin utjecaj na sve članove grupe, provjerava i proučava mogućnosti i načine otvaranja virtualne zajednice; prepoznaje i proučava interdisciplinarne poslove koji su poboljšani razvoje informatike i informacijsko-komunikacijske tehnologije.

Ishodi za VIII. razred su:

#### 1. Informacije i digitalna tehnologija

Učenik: kritički procjenjuje točnost, učestalost, relevantnost i pouzdanost informacija i njihovih izvora (znati izvući najbolje iz bogate ponude informacijskih portala, enciklopedija, knjižnica i obrazovnih računalnih programa); opisuje i planira organizaciju baza podataka, koristi se nekim programom za upravljanje bazama podataka za lakše pretraživanje i sortiranje podataka; opisuje građu računalnih uređaja, objašnjava načine prijenosa podataka u računalu te analizira i vrednuje neka obilježja računala koja značajno utječu na kvalitetu rada samoga računala; prepoznaje i proučava interdisciplinarnu primjenu računalnoga razmišljanja analiziranjem i rješavanjem odabranih problema iz različitih područja učenja.

#### 2. Računalno razmišljanje i programiranje

Učenik: identificira neki problem iz stvarnoga svijeta, stvara program za njegovo rješavanje; dokumentira rad programa i predstavlja djelovanje programa drugima; prepoznaje i opisuje algoritam sortiranja, primjenjuje jedan algoritam sortiranja za rješavanje zadanoga problema u programskom jeziku; prepoznaje i

opisuje mogućnost primjene rekurzivnih postupaka pri rješavanju odabranih problema te istražuje daljnje mogućnosti primjene rekurzije.

### 3. Digitalna pismenost i komunikacija

Učenik: pronalazi, opisuje te uspoređuje različite servise za objavljivanje mrežnoga sadržaja, opisuje postupak objavljivanja mrežnoga sadržaja; samostalno pronalazi informacije i programe, odabire prikladne izvore informacija te uređuje, stvara i objavljuje/dijeli digitalne sadržaje; dizajnira, razvija, objavljuje i predstavlja radove s pomoću sredstava informacijske i komunikacijske tehnologije primjenjujući suradničke aktivnosti.

### 4. e-Društvo

Učenik: Učinkovito se koristi dostupnim e-uslugama u području odgoja i obrazovanja; aktivno sudjeluje u sprečavanju elektroničkog nasilja i govora mržnje.

## **2.2.2. Kurikulum prirodoslovno-matematičke gimnazije**

Kurikulum za srednje škole koji će biti naveden odnosi se na prirodoslovno-matematičke gimnazije s najopsežnijim sadržajem i najvećom satnicom iz predmeta informatike (4\*105 sati godišnje).

Za prvi razred ishodi su:

### 1. Informacije i digitalna tehnologija

Učenik: objašnjava glavne komponente računalnoga sustava i njihove funkcije; primjenjuje načela hijerarhijske organizacije datoteka u računalnim memorijama te razlikuje formate datoteka; analizira i primjenjuje sažimanje datoteka; analizira ulogu binarnog i heksadekadskog brojevnog sustava u prezentaciji digitalnoga prikaza različitih tipova podataka; definira logički izraz za zadani problem; dizajnira, razvija i objavljuje strukturu povezanih mrežnih stranica s pomoću alata i tehnologija koje se izvode na računalu korisnika.

## 2. Računalno razmišljanje i programiranje

Učenik: Analizira problem, definira ulazne i izlazne vrijednosti te uočava korake za rješavanje problema; primjenjuje jednostavne tipove podataka te argumentira njihov odabir, primjenjuje različite vrste izraza, operacija, relacija i standardnih funkcija za modeliranje jednostavnoga problema u odabranome programskome jeziku; razvija algoritam i stvara program u odabranome programskom jeziku rješavajući problem uporabom strukture odluke i ponavljanja; primjenjuje standardne algoritme definirane nad cijelim brojevima.

## 3. Digitalna pismenost i komunikacija

Učenik: pronalazi podatke i informacije, odabire prikladne izvore informacija te uređuje, stvara i objavljuje/dijeli svoje digitalne sadržaje; istražuje usluge interneta i mogućnosti učenja, poslovanja, budućega razvoja; u online okruženju surađuje i radi na projektu.

## 4. e-Društvo

Učenik: U suradničkom *online* okruženju na zajedničkom projektu analizira etička pitanja koja proizlaze iz korištenja računalnom tehnologijom; opisuje probleme koje mogu prouzročiti zlonamjerni programi te probleme koji nastaju kao rezultat električnog napada i krađe električnog identiteta te odgovorno primjenjuje sigurnosna pravila; analizira ulogu koju pomoćna tehnologija i prilagođeni digitalni sadržaji mogu imati u životima osoba s poteškoćama.

Drugi razred obuhvaća:

### 1. Informacije i digitalna tehnologija

Učenik: Opisuje temeljne koncepte računalnih mreža; objašnjava binarno zbrajanje cijelih brojeva kao temeljnu operaciju u računalu; konstruira smisleni logički sklop; opisuje načelo kriptiranja te važnost primjene enkripcije u svakodnevnome životu; istražuje različite vrste ulaznih i izlaznih podataka te pretvorbu u obliku pogodan za računalnu obradu.

## 2. Računalno razmišljanje i programiranje

Učenik: Analizira osnovne algoritme s jednostavnim tipovima podataka i osnovnim programskim strukturama i primjenjuje ih pri rješavanju novih problema; u zadanome problemu uočava manje cjeline, rješava ih potom integrira u jedinstveno rješenje problema; razlikuje složene tipove podataka u zadanome programskom jeziku te se pri rješavanju problema koristi funkcijama i metodama definiranim nad njima; analizira sortiranje podataka kao važan koncept za rješavanje različitih problema; u suradnji s drugima osmišljava algoritam, implementira ga u odabranome programskom jeziku, testira program, dokumentira i predstavlja drugima mogućnosti i ograničenja programa.

## 3. Digitalna pismenost i komunikacija

Učenik: U suradničkom *online* okruženju na zajedničkom projektu istražuje utjecaj ugradnje računalnih sustava u razne uređaje na svakodnevni život; analizira programe s obzirom na licenciju i na preduvjete za instalaciju programa; uspoređuje strategije prikupljanja podataka prema relevantnosti i pouzdanosti izvora podataka odabranim programom učinkovito analizira i prikazuje podatke i rezultate.

## 4. e-Društvo

Učenik: Aktivno pridonosi unapređenju kvalitete života podizanjem ekološke svijesti, analizira i procjenjuje utjecaj informacijske i komunikacijske tehnologije na učinkovitost i produktivnost u raznim područjima i poslovima.

U trećem razredu ishodi su sljedeći:

### 1. Informacije i digitalna tehnologija

Učenik: Za jednostavni problem iz stvarnoga života oblikuje bazu podataka te ju realizira u nekom sustavu za rad s bazama podataka;

### 2. Računalno razmišljanje i programiranje

Učenik: koristeći neki grafički modul vizualizira i grafički prikazuje neki problem iz svoje okoline; rješava problem primjenjujući složene tipove podataka definirane zadanim programskim jezikom; rješava problem primjenjujući rekurzivnu funkciju; uspoređuje različite algoritme sortiranja i pretraživanja podatka; vrednuje algoritme

prema njihovoj vremenskoj složenosti; osmišljava objektni model s pripadnim složenim strukturama podataka, implementira ga u zadanome programskom jeziku; analizira tradicionalne kriptografske algoritme i opisuje osnovnu ideju modernih kriptografskih sustava; definira problem stvarnoga života i stvara programsko rješenje prolazeći sve faze programiranja, predstavlja rješenje i vrednuje ga.

### 3. Digitalna pismenost i komunikacija

Učenik: Planira, stvara, predstavlja i vrednuje multimedijски projekt.

Četvrti razred nam donosi:

#### 1. Informacije i digitalna tehnologija

Učenik: Istražuje mogućnosti različitih programskih jezika; istražuje moderne kriptografske sustave.

#### 2. Računalno razmišljanje i programiranje

Učenik: Rješava problem koristeći se apstraktnim strukturama podataka; stvara aplikaciju s grafičkim korisničkim sučeljem za problem iz stvarnoga života; koristi se modeliranjem i simulacijom za predstavljanje i razumijevanje prirodnih fenomena; koristi se različitim programskim paradigmatama za rješavanje problema iz stvarnoga života; definira problem iz stvarnoga života i stvara programsko rješenje prolazeći sve faze programiranja, predstavlja programsko rješenje i vrednuje ga.

## 2.3. Predmetni kurikulum Republike Slovenije za osnovnu školu

Informatika se u slovenskom školstvu pojavljuje kao izborni predmet u osnovnim školama. Kurikulum za 4., 5. i 6. razred je sastavljen prema smjernicama *Ministrstva za izobraževanje, znanost in šport*, 2013. godine. Od 4. do 6. razreda gradivo je podijeljeno na pet domena: algoritmi, programi, podatci, rješavanje problema, komunikacije i usluge, ishodi su navedeni općenito za sva tri razreda, te su uspostavljene razine usvojenosti znanja - "minimum znanja" koje učenik mora usvojiti na odabranom predmetu, viša razina - "temeljna znanja" i najviša - "zahtjevnija znanja"<sup>6</sup>.

---

<sup>6</sup> Krajnc, R. [et. al] (2013.) *Računalništvo*. Ljubljana: Ministrstvo za izobraževanje, znanost in šport: Zavod RS za šolstvo.

Tako u 5. i 6. razredu učenik:

Pronalazi primjer iz svakodnevnog života za zadani algoritam; opisuje svakodnevne aktivnosti algoritmom; zapisuje algoritam za rješenje problema dijagramom tjeka jednostavnim jezikom; koristi grananje i petlje te povezuje nekoliko algoritama kako bi riješio problem; mijenja ili ispravlja algoritam na način koji rješava problem; stvara jednostavan program bez grananja i pokreće ga; stvara program koji sadrži konstante i varijable; umeće u program različite tipove podataka, logičke operatore, uvjete i grananja; stvara program koji čita ulazne podatke, a varijable mijenjaju vrijednosti s obzirom na određene uvjete; stvara program koji ispisuje vrijednosti varijabli tijekom i nakon pokretanja; uklanja pogrešku ili poboljšava izvedbu programa; prepoznaje različite vrste podataka: tekst, broj i dr.; bilježi podatke u binarnom sustavu; kodira i dekodira poruku; sustavno prezentira podatke; raspravlja i opisuje potrebu za uređivanjem podataka; objašnjava kako zaštititi svoje osobne podatke; planira rješavanje problema; rastavlja glavni problem na potprobleme i pronalazi odgovarajući alat za njega; realizira planirano rješenje te vrednuje i kritički promišlja neuspjele pokušaje; na kreativan način pridonosi rješavanju u grupi; navodi osnovne ideje važne za rad računalnih mreža; objašnjava osnovne pojmove u računalnim mrežama; koristi glavne usluge računalne mreže (e-mail, web, internet); pretražuje podatke online procjenjuje njihovu relevantnost; koristi različite i napredne tehnike pretraživanja na mreži; poznaje i primjenjuje pravila ponašanja na mreži.

Kurikulum od 7. do 9. razreda sastavljen je 2002. godine, a gradivo je podijeljeno na tri glavne domene: uređivanje teksta, računalne mreže i multimedija, od kojih svako je podijeljeno na dodatne tri: osnovne informatike i računarstva, obrada podataka i komunikacija pomoću informacijske tehnologije i programiranje<sup>7</sup>.

Dakle, ishodi su:

## 1. Uređivanje teksta

### a) Osnove informatike i računarstva

---

<sup>7</sup> Batagelj, V. [et al] (2002.) *Računalništvo*. Kočevje: Tiskarna Kočevski tisk d. d.



Učenik: pronalazi informacije i određuje podatke; analizira način zapisa podatka; upoznaje računalo; analizira funkcije računala i poznaju njegovu opremu; pozna osnovne hardverske jedinice računala; razumije zadaće pojedinih jedinica i razlikuje njihova obilježja; stvara kriterije koji se odnose na izbor hardvera i softvera pri kupovanju računala;

#### b) Obrada podatka i komunikacija pomoću informacijske tehnologije

Učenik: koristi računalo za izradu i komunikaciju jednostavnih informacija prezentiranjem odabranih informacija u obliku teksta i slika koje istražuju mogućnosti svakog medija i učenja da je moguće prezentirati informacije i komunicirati na različite načine; vrednuje i analizira različite načine pružanja informacija pri utvrđivanju uspješnosti komunikacije i razumije da naš svakodnevni život ovisi o primljenim informacijama; prepoznaje osnovne tipove računalnog softvera i određuje njezinu svrhu; stječe znanja i vještine za samostalno korištenje računala;

#### c) Programiranje

Učenik: analizira jednostavan problem; upotrebljava osnovne korake u programiranju;

### 1. Računalne mreže

#### a) Osnove informatike i računarstva

Učenik: analizira ono o čemu zavisi razumijevanje informacija; prepoznaje temeljne vrste softvera; razumije zadaće određenih softvera.

#### b) Obrada podatka i komunikacija pomoću informacijske tehnologije

Učenik: Dovršava tekst i sprema dokument s uređivačem teksta; izrađuje sliku s programom za crtanje koju umeće u tekstualni dokument; unosi podatke u proračunsku tablicu, crta grafikon i umeće ga u tekstualni dokument; umeće dokument na internet; koristi javno dostupne podatke; razvija kritički stav prema korištenim podacima i stav prema zaštiti autorskih prava i zaštiti podataka; uspoređuje i procjenjuje prednosti i nedostatke interneta; razvija znanja i vještine za samostalno rješavanje problema na računalu.

### c) Programiranje

Učenik: Analizira jednostavan problem; koristi osnovne korake programiranja

## 2. Multimedija

### a) Osnove informatike i računarstva

Učenik: pretražuje informacije u različitim medijima; uspoređuje kvalitetu različitih informacija;

### b) Obrada podatka i komunikacija pomoću informacijske tehnologije

Učenik: Koristi računalo za oblikovanje i komuniciranje svojih ideja; multimedijском prezentacijom istražuje mogućnosti svakog medija i spoznaje da se ideja može predstaviti na različite načine; analizira prezentaciju svojih ideja s kolegama te evaluira uspjeh komunikacije i prepoznaje da osim sadržaja to ovisi i o načinu i obliku prijenosa informacija; razvija znanja i vještine kako bi samostalno pomoću računala riješio probleme;

### c) Programiranje

Učenik: Analizira poznati i razumljiv, ali složeniji problem; razumije zahtjevnije korake pri programiranju.

## 2.4. Predmetni kurikulum Velike Britanije

### 2.4.1. Key stage 3 i 4

Britanski kurikulum je doživio mnoge promjene od 90-ih prošlog stoljeća, programiranje i računalno razmišljanje su prepoznati kao ključni alati za buduće generacije, te se od klasičnog informacijsko-komunikacijskih tehnologija okrenulo prema modelu kojoj u svojoj srži sadrži upravo *computing*. Britansko osnovnoškolsko i srednjoškolsko obrazovanje je podijeljeno po razinama - *key stages* - a samo obvezno obrazovanje se sastoji od četiri razine. Prva obuhvaća djecu od 5. do 7.,

druga od 7. do 11., treća od 11. do 14., a četvrta od 14. do 16. godine. Bavit ćemo se posljednjim dvjema.

*Department for Education* je 2012. godine odredilo kako su sve škole obvezne objaviti one informacije koje su vezane za školski i predmetni kurikulum, ali i načine kako ga ostvariti, pa je tako 2013. godine objavljena verzija za informatičko predmetno područje koje su nazvali - *computing* - što bi doslovno značilo obrada podataka putem računala ili radnje koje obavlja računalo dok izvršava instrukcije. U kontekstu engleskog jezika i u duhu hrvatskog nazovimo taj predmet informatikom. Ovaj predmet je obavezan na svim razinama.

Podijeljen je u tri glavne domene<sup>8</sup>:

1. Računalna znanost (*computer science*),
2. Informacijske tehnologije (*information technology*) i
3. Digitalna pismenost (*digital literacy*).

U nastavku slijedi opis domena:

#### 1. Računalna znanost:

Domena računalne znanosti se odnosi na znanstveni i praktičan pristup proučavanja računala, na ono što računala mogu obrađivati, na koji način to čine i kako se mogu koristiti za rješavanje problema.

#### 2. Informacijske tehnologije:

Ova domena obuhvaća načine na koji računala i telekomunikacijska oprema rade, kako se mogu koristiti za spremanje, pristup, prijenos i manipulaciju podatcima.

#### 3. Digitalna pismenost:

Odnosi se na sposobnost za učinkovito, odgovorno, sigurno i kritičko upravljanje, evaluiranje i stvaranje digitalnih artefakata koji uključuju računalni sustav, medije za pohranu podataka (tvrdi disk, DVD, USB), elektronički dokument i mrežne pakete, korištenjem raznih digitalnih tehnologija.

---

<sup>8</sup> (2013.) *The national curriculum in England*. London: Department for Education

U britanskoj inačici informatike na trećoj razini učenik na području računalne znanosti: dizajnira, koristi i evaluira računalne apstrakcije koje modeliraju stanje i ponašanje problema i fizičkih sustava stvarnoga svijeta; razumije nekoliko ključnih algoritama koje odražavaju računalno razmišljanje (sortiranje, pretraživanje); koristi logičko razmišljanje pri uspoređivanju korisnosti alternativnih algoritama za isti problem; koristiti dva ili više programska jezika, od kojih je barem jedan tekstualan u rješavanju raznih računalnih problema; koristi odgovarajuću strukturu podataka (lista, tablica, polje); dizajnira i razvija modularne programe koji koriste procedure i funkcije; razumije osnove Boolove logike (AND, OR, NOT) i korištenje u sklopovlju i programiranju; razumije binarno predočavanje brojeva i računati osnovne operacije binarnih brojeva (zbrajanje binarnih brojeva, pretvaranje iz binarnog u dekadski); razumije sklopovne i programske komponente koje sačinjavaju računalni sustav i kako međusobno komuniciraju s drugim sustavima; razumije kako se instrukcije spremaju i izvršavaju unutar računala; razumije kako može različite tipove podataka (tekst, zvuk, slike) predstavljati i digitalno manipulirati u obliku binarnih znamenki.

U informacijskoj tehnologiji učenik: koristi kreativne projekte koje uključuju odabir, korištenje i kombiniranje više aplikacija, po mogućnosti širok raspon uređaja za ostvarivanje zahtjevnih ciljeva, uključujući sakupljanje i analiziranje podatka te ostvarivanje potreba poznatih korisnika; stvara, revidira i ponovno iskorištava digitalne artefakte za odabranu publiku s naglaskom na vjerodostojnost, dizajn i uporabljivost.

Na području digitalne pismenosti učenik: razumije koristiti tehnologiju na siguran, pošten, odgovoran, zaštićen način, uključujući zaštitu identiteta (online) i privatnost; prepoznaje neprikladan sadržaj, način komuniciranja i ponašanja i zna kako prijaviti sumnjive aktivnosti.

Na trećoj razini u središtu plana i programa je računalno razmišljanje. Ono se ne odnosi isključivo na programiranje kao takvo, već se shvaća kao alat i vještina koje učenici trebaju razvijati, a koje im može poslužiti u daljnjem školovanju i razvoju. Radi se o vještini koja zahtjeva da se problem iz stvarnog života rastavi na manje dijelove te pomoću određenog algoritma dođe do rješenja. Iako je u kontekstu informatike ovaj način razmišljanja primjenjiv u programiranju, on se može koristiti pri konstruiranju različitih modela u drugim područjima ako npr. biologija, fizika ili statistika. Npr. predviđanje rasta populacije, računanje vremena pada utega ili pera s

određene visine ili računanje prosječne plaće u zadnjih 10 godina. Dakle, kroz razvijanje modela koji je nastao kao rezultat rješavanja problema iz stvarnog života učenici uče kako pristupiti problemu, kako koristiti računalo koje će im omogućiti evaluirati i usporediti koliko zapravo takav model odgovara stvarnosti, te omogućiti predikciju za određene pojave. Takvo modeliranje zovemo modularno dizajniranje.

Kroz pojmove algoritma, programa i koda, razumjet će razlikovati iste i shvatiti da algoritam koristimo u svakodnevnom životu (npr. algoritam kuhanja čaja). Također, treba imati na umu da se neki problemi mogu riješiti na više načina ili da se korištenjem različitih algoritama može doći do istog rješenja te da i algoritam i rješenje ovise o složenosti problema. Nas zanimaju i uređivanje i pronalaženje podataka, a takav problem rješavamo korištenjem algoritama za pretraživanje i sortiranje. Neke od tema je moguće predstaviti *unplugged* aktivnostima - bez upotrebe računala ili neke pomoćne tehnologije.

Učenici bi tako trebali naučiti koristiti najmanje dva programska jezika iz razloga jer se tako lakše mogu razumjeti određeni koncepti iz programiranja i ono omogućuje uopćavanje te olakšava učenje drugih programskih jezika (sličnost u naredbama i sintaksi). To nas dovodi do korištenja osnovnih koncepata pri programiranju: Slijed (sekvenca), odabir (selekcija) i ponavljanje, te složene strukture podataka tj. polje, tablica i lista.

Učenjem Boolove logike, učenik razumije i objašnjava osnove funkcioniranja računala, jer se ono bazira na izjavama o istinitosti i lažnosti, odnosno napona koje prolazi kroz računalne čipove. Najmanja veličina spremljena u računalu je jedan bit, oni su u računalu zabilježeni u obliku koji je razumljiv za računalo, u binarnom brojevnom sustavu, nizom nula i jedinica. Osim što će učenik naučiti pretvarati binarni broj u dekadski ili heksadecimalni, također će izvršavati operacije zbrajanja i oduzimanja binarnih brojeva.

Računalo se sastoji od fizičkog dijela - sklopovlja, te programskog dijela, popularno nazvan softver koji uključuje operacijski sustav, programe i aplikacije. Osnova funkcija računala jest da pri određenom unosu podataka, da ih obradi te nam ponudi određeni odgovor putem izlaznih podataka.

Računala mogu biti povezana u mrežu, te putem pametnog sklopovlja - usmjernika ili preklopnika komunicirati s drugim računalima, a sami podatci se prenose putem paketa. Internet protokol adresa je numerička oznaka u mreži koja

nam govori odakle je stigla poruka ili gdje poruka treba stići. Učenici će tako naučiti i razlikovati *WWW (World Wide Web)* i internet, gdje je *WWW* kolekcija povezanih podataka, a internet skup umreženih računala koji komuniciraju zajedno. Za razliku od programskih jezika, u mrežnim stranicama se koristi *HTML (Hypertext Markup Language)* jezik. Ono što nam omogućava računalo i povezanost na internet jest komunikacija s drugim korisnicima putem elektroničke pošte.

Računalo služi i spremanju podataka, pa će tako učenici naučiti čemu služi primarna i sekundarna memorija, te na koji način se spremaju tekst, zvuk i sliku u računalo. Učenici će također trebati savladati i stvaranje digitalnih artefakata koje će moći prezentirati publici, te prilagoditi osobama s invaliditetom i uključiti osobe kojima engleski nije prvi/materinski jezik.

Učenici će učiti i o određenim pravnim, moralnim i drugim aspektima vezani za ponašanje na internetu koje uključuje: korištenje elektroničke pošte i mrežnih stranica, komunikaciju preko interneta, zaštitu računala od virusa i trojanskog konja, zaštitu identiteta i osobnih podataka i intelektualno vlasništvo.

Od učenika se očekuje prijavu svakog uznemiravanja koje je usmjereno na njih osobno ili ako primijete u *online* okruženju ili putem mobitela postojanje ponašanja koja upućuju na isto.

Na četvrtoj razini navedeni su opći ciljevi kratkim opisom, pa učenik: razvija mogućnosti, kreativnost i znanja u računalnim znanostima, digitalnim medijima i informacijskim tehnologijama; razvija i primjenjuje analitičke, problemske, dizajnerske i računalne vještine; razumije kako promjene u tehnologiji utječu na sigurnosti, kao i nove načine zaštite *online* privatnosti i identiteta te kako identificirati i prijaviti sumnjiva ponašanja.

Na ovoj razini se očekuje da se teme iz treće razine podignu ili prošire na neki drugi način ili kroz uvođenje zasebnih predmeta tematski usmjereni na informatiku ili međupredmetne teme odnosno obrađivanje informatičkih tema u drugim predmetima.

### 2.4.1. Key stage 5<sup>9</sup>

Iako nakon 16. godine obrazovanje više nije obvezno, navest ćemo primjer sadržaja aktualnog kurikulumu za petu razinu prema školi *UTC Leeds*. Peta razina uključuje učenike od 16. do 18. godina.

U opisu kurikulumu se navodi da je glavni cilj *computer science* nastave razvoj generičkih načela programiranja koje nadilazi sve jezike, a uključuje: apstraktno mišljenje, rješavanje problema, matematičko rezoniranje i načela izgradnje softvera.

Prvom godinom (12. godina školovanja) učenik: razlikuje tipove podataka koji su predstavljeni binarnim brojevima; razumije selekciju, iteraciju i potprograme; koristi binarne i tekstualne datoteke te enkripcijske tehnike; prepoznaje unutarnju arhitekturu računala, osnovne digitalne elektronike (logička vrata i Boolova algebra) i mrežne sustave; praktično koristi *C#*; proceduralno programira na jezicima više razine i programira u jezicima niže razine (asemblersko kodiranje).

Na drugoj godini (13. godina školovanja) učenik: poznaje objektno orijentirane jezike; praktično koristi *C#*, *Haskell* i *SQL* (modeli baze podataka); koristi napredne algoritme pretraživanja i uspoređuje njihovu učinkovitost; razumije pojmove rekurzije, nasljeđivanja, polimorfizma; razumije ograničenja binarnog spremanja u računalu; prepoznaje arhitekturu računala i koristi znanja iz digitalne tehnologije za stvaranje dijelova računalne memorije (bistabili).

Na kraju druge godine učenik je evaluiran i putem projekta (koji iznosi 20% ocjene), a uključuje stvaranje modela iz stvarnog života putem: poslovne aplikacije, edukacijske računalne igrice ili simulacije matematičkog ili nekog drugog znanstvenog načela.

## 2.5. Didaktička načela u predmetnom kurikulumu Republike Hrvatske

Ako se usmjerimo samo na ishode kao jedne od bitnih sastavnica kurikulumu za nastavu informatike, vidimo da je najviše zastupljen onaj dio koji se odnosi na načelo zornosti i apstraktnosti, te sistematičnosti i postupnosti. Preostala načela su prepuštena školskom i nastavnom kurikulumu. Tako postoji napredak od općih i osnovnih pojmova k složenijim.

---

<sup>9</sup> *KS5 Computer Science*

U domeni programiranja u 5. razredu tako je naveden ishod s kojim učenik treba osmisлити jednostavan program upisivanjem i ispisivanjem rezultata, u 6. razredu taj ishod se podrazumijeva uz korištenje naredbi za grananje i ponavljanje. 7. razred zahtjeva nešto složenije radnje, uz korištenje algoritama pretraživanja (binarni, linearni), te proširuje program, koji postaje složeniji radi stvaranja potprograma, da bi u 8. razredu to rezultiralo prepoznavanjem i korištenjem algoritama sortiranja (zamjena elemenata, umetanje) i rekurzivnog programiranja. Već sada je očito da se složenost ishoda promijenila od 5. do 8. razreda.

Nadalje, to ćemo pokazati i na primjeru srednjoškolske domene programiranja. U 1. razredu se koriste jednostavniji tipovi podataka, ali obuhvaća složenije probleme i operacije, koristeći grananje i ponavljanje. U 2. razredu se ide korak dalje pa se, pored složenih tipova podataka, po prvi put uvode i funkcije (što su zapravo složeniji oblici potprograma). Sve se objedinjuje, ali se ne zahtijeva samostalno stvaranje algoritama sortiranja, već samo njihovo objašnjavanje i implementacija (kada koristiti, koji koristiti i sl.).

Složeni tipovi podataka i složene strukture, uz sortiranje i pretraživanje dolaze u 3. razredu, dok na kraju srednjoškolskog školovanja obrađuje se apstraktna struktura podataka (klasa objekata čije logičko ponašanje određuje skup vrijednosti i skup operacija) što uvodi učenika u objektno orijentirano programiranje. Dakle od 1. do 4. razreda opet smo se pomicali od najjednostavnijih tipova podataka, struktura i algoritama do najsloženijih.

U kurikulumu imamo i eksplicitno navedena načela koja se odnose na individualizaciju gdje se od učenika očekuje razvijanje navedenih računalnih kompetencija, te samostalnost, odgovornost i poduzetnost. Načelo individualizacije je vidljivo iz činjenice da su ishodi usmjereni isključivo na samog učenika. Napominje se i rad s djecom s posebnim obrazovnim potrebama, te nadarenom djecom.

Načelo socijalizacije je vidljivo u navođenju timskog rada i suradnja koji su motivirajući faktori za učenje. Socijalizaciju karakteriziraju i različite vrste interakcije s drugim učenicima, nastavnicima, roditeljima, stručnjacima i širom zajednicom. Također, učitelj facilitira odnose u razredu kroz otvorenu komunikaciju koja se temelji na partnerskim odnosima, međupoštovanju, toleranciji, prihvaćanju i empatiji. Kada govorimo o socijalizaciji možemo i govoriti o tome da učenik izgradnjom svog identiteta integrira i osjećaj pripadnosti razredu i školi.



O načelu racionalizacije i ekonomičnosti možemo govoriti o informatici kao sadržaju koji se može proširiti i na druga područja, odnosno predmete, napominjući da su usvojena znanja, vještine i stavovi podrška ostalim predmetima i međupredmetnim temama, ponajviše iz praktičnog razloga, izbjegavajući redundantnost. U organizaciji nastave svaki učenik u umreženoj učionici bi trebao imati svoje mjesto za računalom, a radi održavanja motivacije potrebno bi bilo koristiti se različitim metodama.

Načelo aktivnosti odnosi se na uvjerenje da se najbolje uči aktivno sudjelujući u nastavi što facilitira razvoj i korištenje kreativnog mišljenja, a to se najbolje ogleda u stvaranju digitalnih podataka kao što su prezentacije, video uradci, računalni modeli ili programi. Takva aktivnost bi trebala i omogućiti da sam učenik postane demokratski aktivan građanin (npr. korištenje digitalnih javnih usluga, kao npr. e-građana, briga za zdravlje i okoliš - zbrinjavanje elektronskog i električnog otpada). Učenik bi trebao tako i aktivno sudjelovati u izboru sadržaja, digitalnih alata i programa, koristeći primjere iz stvarnog života te predstavljanjem svog kreativnog rada.

Načelo aktivnosti i socijalizacije su vidljivi u projektnoj nastavi gdje učenik samostalno, u paru ili grupi istražuje, razvija komunikacijske vještine, kritički vrednuje svoj i rad drugih učenika, razmišlja o problemu. Učenik također može osim sudjelovanja u razrednim i školskim projektima sudjelovati i u sklopu izvanškolske aktivnosti na nacionalnim i međunarodnim projektima.

## **2.6. Didaktička načela u predmetnom kurikulumu Republike Slovenije**

U slovenskom kurikulumu za osnovnu školu programiranje je podijeljeno na ciklus od 4. do 6. razreda u sklopu algoritama i programa, navodi se grananje, petlje i različite tipovi podataka. Tu su ishodi općeniti kroz cijeli ciklus i nije specifično navedeno kako se kreću od osnovnih prema složenijim sadržajima. Od 7. do 9. razreda postoji posebna domena programiranja u kojem je jako kratko navedeno na što se odnosi, dakle krećemo se od rješavanja jednostavnog problema i osnova u 7. razredu, neznačajno se mijenja ishod u 8. razredu, pa je uglavnom sličan i kreće se do rješavanja složenijih problema u 8. razredu.

Ovako pojednostavljeni i kratak opis ishoda možemo pripisati tome ili da se nastavniku ostavlja prostor određivanja na koji način da se postavi prema sadržaju, smatranje programiranja nebitnim (što je jako malo vjerojatno), nedovoljno razrađenim ili nepotrebnim prije srednjoškolskog obrazovanja, a moguće je i da njegova izbornost utječe na sadržaj kurikuluma. Ako govorimo o didaktičkim načelima, možemo samo govoriti o nekoj blagoj sistematičnosti i postupnosti, ako govorimo o programiranju, jer od preostalih domena, ona je najmanje razrađena kroz ishode.

U ciklusu od 4. do 6. razreda ostala načela možemo prepoznati izvan ishoda: Načelo aktivnosti je navedeno kao usmjerenost učenika približavanju znanstvenom načinu razmišljanja, a motiviranost, koju treba poticati nastavnik, kroz aktivnost treba približiti učenika računalnoj znanosti tražeći primjere iz stvarnoga života, što će utjecati pozitivno na učenikovo samopouzdanje. Također aktivnost se može vidjeti kroz zanimanje za nove informacije, promatranje detalja, postavljanje zanimljivih i smislenih pitanja, upotrebu digitalnih izvora informacija, kritičko mišljenje (vrednovanje izvora informacija, funkcionalnost programa ili algoritama, moralno-etički problemi o zaštiti identiteta i digitalnom nasilju).

Načelo individualnosti se posebno napominje i naglašava razlikovanje učenika po računalnim vještinama prema svojim sposobnostima i drugim posebnostima, kao što su nadareni učenici, učenici s poteškoćama u učenju te obrazovanju strane djece drugih nacionalnosti.

Načelo socijalizacije dolazi do izražaja kada se radi o projektnoj nastavi, tada se može izvoditi rad u parovima, pa čak i u manjim skupinama. Učenici se međusobno potiču, uče jedni od drugih i izgrađuju timsko rješenje. Rješavaju probleme i pitanja koja se pojavljuju u grupi, nakon toga traže ili osmišljavaju rješenja koja su nastala kao rasprava i dogovor između članova.

Po potrebi (opet se vraćamo na načelo aktivnosti), ako je potreban učitelj, kao mentor, potiču ih i pomažu im da se vrate konstruktivnom rješavanju problema tako što im postavljaju dodatna pitanja i savjete. Cilj je dakle usmjeravati učenika, kako bi samostalno došao do završnog proizvoda ili rješenja.

Načelo sistematičnosti i postupnosti se ne zadržava samo na nastavi informatike, već se kroz međupredmetne teme proširuje i na matematiku (formalni postupci, dokazi, logički i aritmetički izrazi, problemski zadaci, logičko i algoritamsko

razmišljanje), likovnu umjetnost (razumijevanje koncepata u računalnoj grafici, smisao za dizajn i digitalni dizajn), slovenski (opisivanje postupaka, vježbanje razumijevanja tekstova, izvještavanje, pretvaranje teksta u drugi format, npr. iz tekstualnog zadatka u tablicu ili dijagram, nadalje: dijeljenja teksta na logičke jedinice, rezimiranja i formuliranja zaključaka), društvo (timski rad, utjecaj na okoliš, intelektualno vlasništvo, društvene vještine, grupna suradnja, kritičkog mišljenja, argumentacija i timska rješenja), prirodnu znanost i tehnologiju (osnovni pojmovi rada računala, planiranje rada, ispitivanje procesa, precizno i sustavno istraživanje i promatranje), engleski jezik (računalni izrazi na slovenskom i engleskom jeziku).

U ciklusu koji slijedi, a obuhvaća 7., 8. i 9. razred pojavljuje se također načelo aktivnosti, gdje učenik sadržajnu građu, osim samostalno, pokušava pronaći konzultirajući se s učiteljem i knjižničarima. Aktivno učenje podrazumijeva i stvaranje određenih digitalnih dokumenata, kao izrada web stranice, prezentacije te dokumentiranje i komentiranje nekog programskog rješenja.

## **2.7. Didaktička načela u predmetnom kurikulumu Velike Britanije**

Britanski kurikulum za srednju školu nam donosi u kontekstu programiranja računalnu znanost (*computer science*) u čijoj jezgri je računalno razmišljanje. Ovdje je okvirno navedeno što treći ciklus treba ishakovati za učenika i pored ostalog postoji specifična uputa za poznavanje barem dva programska jezika, od kojeg barem jedan je tekstualan, a obuhvaća i algoritme sortiranja i pretraživanja, te složene strukture podataka. Ovdje se jasno daje do znanja, da se programiranje eksplicitno radi kroz računalne znanosti, ali da bi nakon druge godine računalni način razmišljanja trebalo proširiti i na druge predmete - dakle način razmišljanja koji služi ne samom uskom pojmu isključivo vezan za programiranje, već i rješavanje problema i zadataka u drugim područjima. Ovdje je očito da se informatičko područje smatra alatom i da se do kraja trećeg ciklusa trebaju savladati pojmovi iz ovog područja, od osnovnih do složenijih.

S obzirom na to da je obrazovanje u Velikoj Britaniji obvezno do 16. godine, tako je i informatički kurikulum isprogramiran. Očigledno su ovdje vidljiva načela zornosti i apstraktnosti, kao sistematičnosti i postupnosti: u domeni *computer science* zahtjeva se da se zadana razina podigne na višu u i to je u potpunosti

prepušteno nastavniku. Sam kurikulum je strukturiran tako da govori samo o sadržaju predmeta.

U ovom dijelu potrebno je i spomenuti priručnik<sup>10</sup> za nastavnike u kojem se uočavaju (čak i ako se odmaknemo od programiranja) različita načela koja obogaćuju kurikulum. Načelo aktivnosti je najviše zastupljeno, tako se zahtjeva da učenik aktivnije uči pisanjem programa i stvaranjem teorijskih pitanja koja bi mogla biti korisna u poučavanju drugih. Smatra se da bi bolje učili ako bi stvarali digitalne artefakte (prezentacije, *web* stranice, video uradci) i iste prezentirali publici, živoj (npr. mlađim učenicima) i kada bi učinili materijale dostupnima u *online* okruženju.

Učenik isto tako može obogatiti digitalnu pismenost kroz dokumentiranje znanja i činjenica. Može se kroz individualne interese zanimati za međupredmetne teme, projekte i aktivnosti izvan škole. Modeliranje tako može pomoći u stvaranju profesionalne budućnosti i karijere, tako što se projektiranjem nekog prototipa može dizajnirati, stvarati, testirati i vrednovati uradak.

Naglašava se i učenje koje zahtjeva učenje na greškama, ali i činjenica da se neki problemi mogu riješiti na više načina, a ne samo na jedinstven način. Učenje istraživanjem - gdje najviše dolazi do izražaja načelo aktivnosti - kao npr. u procesu *debugginga*, gdje učenik provjerava u računalnom programu koje su to pogreške i kako ih ispraviti da bi program računao ono za što je namijenjen ili učenje podržano tehnologijama, gdje se u *online* okruženju pretražuju digitalni sadržaji kao npr. blogovi. Postavljanjem otvorenih pitanja se ruše zastarjela obrasci i mišljenja o nastavi kao o procesu učenja u kojem se trebaju pamtit samo činjenice. U kontekstu aktivnosti imamo i vrednovanje od strane učenika.

Učenik treba moći samostalno zadati si cilj, što može utjecati na njegovu motivaciju i samopoštovanje. Shvatit ćemo to kao samovrednovanje, gdje zapravo učenik procjenjuje koliko mu vlastita znanja, vještine i iskustvo mogu pomoći ostvariti cilj (riješi zadatak, završi projekt, dođe do odgovora i sl.). Učenici mogu vrednovati jedni druge i tako dobivati povratne informacije o vlastitom radu, ne samo od nastavnika već i od strane svojih razrednih kolega. Za razvijanje metakognicije primjenjuje se pristup *KWL* (*what I know, what I want to know, what I have learned*), gdje učenik može sam procijeniti što zna, što želi znati i što je naučio.

---

<sup>10</sup> Kemp, P. (2014.) *Computing in the national curriculum*. Bedford: Newnorth Print, Ltd.

U kurikulumu kao i u ovom priručniku napominje se i načelo jednakosti šansi. Do zadnjeg ciklusa u obveznom obrazovanju (tj. *key stage 3*) učenici će doći s različitim znanjima, iskustvima, vještinama i kompetencijama, pa se smatra da je dobro provjeriti kakvo je stanje na terenu. Ako je potrebno provesti provjeru te intervenirati. Općenito postoje zahtjevi za inkluzijom i socijalnom jednakosti, pa tako sve tehnologije moraju biti dostupne svima, odnosno pristupačne bez obzira na rod, socijalnu pripadnost, nacionalnu pripadnost, prilagođene učenicima s posebnim obrazovnim potrebama ili invaliditetom, učenicima kojima engleski nije prvi jezik te nadarenim i talentiranim učenicima.

## **2.8. Poredba hrvatskog i slovenskog kurikuluma**

Usporedimo sada hrvatski i slovenski kurikulum. Ako promatramo samo slovenski, unutar zadnja dva osnovnoškolska ciklusa se razlikuje po domenama i manjem dijelu je nedosljedan, nema opise domena, samo navodi koji su ciljevi, ishodi, tri razine znanja koje se nazivaju standardima a navedene su kao minimalno, temeljno i zahtjevno, navode se upute za nastavu i načine vrednovanja.

Hrvatski je dosljedan kroz cijelo osnovnoškolsko i srednjoškolsko obrazovanje, daje strukturalnu podjelu na četiri domene, koje su i opisane. Kurikulum se također sastoji od odgojno-obrazovnih ishoda, razrade i razina usvojenosti, preporuka za ostvarivanje odgojno-obrazovnih ishoda po razredima, odnosno osnovnih škola i različitim tipovima srednjih škola i gimnazija s brojem sati i oblicima izvođenja samog predmeta te vrednovanje (za učenje, kao učenje, naučenoga). Na poslijetku se navode mogućnosti onih koji mogu obavljati nastavnički odgojno-obrazovni rad.

Hrvatski i slovenski kurikulum su domenski slični, iako je slovenski drugačije strukturiran pa među domenama možemo prepoznati digitalne tehnologije i programiranje. Ako govorimo o području programiranja, hrvatski kurikulum ima opširnije i detaljnije opisane ishode. Slovenski ne spominje koje metode i načine računalnog razmišljanja bi učenik trebao usvojiti, te je ishod na ovoj domeni općenit. Npr. u hrvatskom se spominju eksplicitno ključne riječi rekurzija i algoritmi sortiranja, dok u slovenskom se govori o jednostavnim i složenim problemima.

U oba kurikulumu dobro su uravnoteženi ishodi koji se odnose na sadržaj i učenika i spominje se međupredmetna važnost informatičkog sadržaja. Nadalje,

podjednako ističu važnost kritičkog mišljenja i logičkog razmišljanja, usmjereni su na razvoj individualnosti učenika, ali i kontekstu grupe povezujući razvoj s timskim radom i suradnjom.

U kontekstu didaktičkih načela, oba kurikulumata nastoje pokriti sva navedena načela i gledajući između redaka vidljive su ideje građanskog odgoja i razvoja učenika kao demokratskog bića.

Upravo radi navedenih različitosti možemo reći da je hrvatski kurikulum koherentniji i sveobuhvatniji (u vertikalnom i horizontalnom smislu) nego što je slovenski.

## **2.9. Poredba hrvatskog i britanskog kurikulumata**

Hrvatski kurikulum je na tragu britanskog. Kao što je već prije bilo navedeno, britanski kurikulum već odavno prepoznaje potrebu računalnog razmišljanja i vidljiva je razina zahtjeva za ishode kod učenika na svim razinama. Takvu ideju možemo i prepoznati u hrvatskom kurikulumu u domeni računalnog razmišljanja i programiranja.

Zanimljivo je i da ishodi za *key stage* 3 britanskog kurikulumata odgovaraju razini ishoda za drugi razred hrvatskog kurikulumata, odnosno možemo reći da ishodi za britanskog učenika s 14 godina odgovaraju ishodima hrvatskog učenika sa 16 godina. To nam najbolje pokazuje primjer poznavanja složenih tipova podataka i poznavanje najmanje dva programska jezika, od kojih je jedan tekstualan, koji se u britanskom kurikulumu javljaju ranije, nego u hrvatskom. Britanski kurikulum podrazumijeva da su neki računalni sadržaji integrirani u obrazovanje kao npr. informacijsko-komunikacijske tehnologije, odnosno podrazumijeva svojevrsnu međupredmetnost.

Postoje neke podudarnosti u domenskoj strukturiranosti, pa tako računalna znanost odgovara računalnom razmišljanju i programiranju, digitalna pismenost i komunikacija i e-Društvo prepoznajemo u digitalnoj pismenosti, a informacijske i digitalna tehnologija su analogne informacijskim tehnologijama. Dakle, oba kurikulumata imaju specifične domene koje su opisane i razrađene kroz ishode. Intuitivno možemo zaključiti da je odgovornost i obveza nastavnika veća u britanskom kurikulumu s obzirom na to da navodi samo svrhu, ciljeve i ishode (iako

priručnik razrađuje detaljnije i složenije određena očekivanja koja bi trebao ponuditi nastavnik) kao okvir za nastavu informatike.

U oba kurikulumu su podjednako zastupljeni ishodi koji se odnose na sadržaj i učenika. Britanski više spominje samostalnost i individualnost, i kako ona najviše dolazi do izražaja kroz projektnu nastavu i timski rad. Iako su poprilično ujednačeni na području didaktičkih načela u britanskom kurikulumu više dominira ideja o aktivnosti učenika koja će mu omogućiti veću samostalnost i razvoj. Vrednovanje je prepušteno školama, tj. nastavniku na kojem je odrediti metode, načine, izvore i kako vrednovati ishode i postignuća učenika. Također, u hrvatskom kurikulumu podjednaka je važnost stavljena i na vrednovanju, dok u britanskom, pored aktivnosti su više usmjereni na rezultate ishoda.

## 3. Implementacija ishoda

### 3.1. Sadržaj kao putokaz do ishoda

U ovom dijelu ćemo objasniti kako doći do željenog ishoda. Za primjer ćemo uzeti ishod iz domene računalnog razmišljanja i programiranja za četvrti ciklus, prvi razred prirodoslovno-matematičke gimnazije: Učenik razvija algoritam i stvara program u odabranome programskom jeziku rješavajući problem uporabom strukture odluke i ponavljanja. Ako promotrimo samo sadržaj ovog ishoda, možemo izdvojiti ključne riječi: algoritam, (računalni) program, (odabrani) programski jezik, struktura odluke i (struktura) ponavljanja.

S obzirom na to da se radi o prvom razredu srednje škole potrebno je primijeniti jedno od načela didaktike - načelo jednakih šansi. Potrebno je saznati, dobiti povratnu informaciju o ujednačenosti kurikuluma na razini osnovnih škola, najbolje je sastaviti kratak upitnik s kojim možemo saznati jesu li se učenici sreli s programiranjem, u kojoj mjeri i tako saznati neke osnovne informacije.

Da bi došli do navedenog ishoda, potrebno je proći kroz različiti spektar pojmova koji se odnose na računalno razmišljanje i programiranje. Primjenjujemo načelo sistematizacije i postupnosti, u manjoj mjeri zornosti i apstraktnosti, jer možemo pretpostaviti da su učenici sigurno upoznati s nekim pojmovima iz ovog područja i nalaze se u fazi apstraktnog mišljenja. Trebamo se usmjeriti približavanju računalnog razmišljanja učeniku, a uključuje: logičko ili algoritamsko razmišljanje, dekompoziciju, rješavanje problema i apstrakciju<sup>11</sup>.

#### 3.1.1. Algoritam i algoritamski postupci

Algoritam je prvi pojam koji se nadovezuje na dva načina na koje ga možemo zapisati: dijagram toka, koji grafički predstavlja ono što želimo programirati, te pseudo-kod koji govornim jezikom zapisuje korake u algoritmu. Potrebno je uvesti osnovne pojmove iz matematičke logike (npr. Boolova algebra, tablice istinitosti). Ovaj dio se odnosi na one probleme koji su sastavljeni od više dijelova, a potrebni su

---

<sup>11</sup> Brennan, K., & Resnick, M. (2012.) New frameworks for studying and assessing the development of computational thinking. *In Proceedings of the 2012 Annual Meeting of the American Educational Research Association*. Vancouver, Canada. Vancouver, Canada



pri programiranju, pogotovo ako postoje određeni uvjeti koji odgovaraju operacijama sa skupovima a to su: presjek, unija i negacija. Osim toga i pokazati da računalo i u programiranju u osnovi funkcionira na binarnom načelu istinitih i lažnih izjava. Potrebno je uvesti pojmove koje se odnose na varijable i vrste operatora: relacijski, aritmetički i logički.

Krećemo s uvođenjem različitih algoritamskih postupaka, a to su: slijed, uvjetno grananje i ponavljanje ili petlja. Slijed se odnosi na upisivanje ulaznih i ispisivanje izlaznih podataka. Grananje se koristi kada imamo jedan ili više uvjeta koji trebaju biti zadovoljeni da bi se izvršio blok naredbi, a petljom omogućavamo programu ponavljanje bloka naredbi dok je zadovoljen određeni uvjet, uz korištenje po potrebi naredbi za prekid i nastavljanje izvođenja programa.

### 3.1.2. Varijable i tipovi podataka

U prvom razredu gimnazije spominju se samo jednostavni tipovi podataka, pa ćemo raditi uglavnom s prirodnim, cijelim i realnim brojevima, te nizom znakova (*string* podatci).

Računalni jezik koji smo odabrali je *Python*. Program ili razvojni alat u kojem radimo je verzija od *Python* 3.0 na dalje., imamo dvije radne površine na koju možemo upisivati podatke i naredbe: interaktivni dio ili glavni prozor (*Python Shell*) i skriptni ili sporedni prozor.

Varijable su podatci koje ulaze u program. Ono što je karakteristično za naziv varijable jeste da smiju započinjati sa slovom ili *underscore* znakom “\_”, ostatak naziva se može sastojati od brojeva, slova i *underscorea*, a treba i pripaziti na korištenje velikih i malih slova (*case sensitive*) jer pravi razliku među varijablama (npr. Varijabla i varijabla smatra dvjema različitim varijablama). Bitno je i napomenuti da nazivi varijabli ne mogu biti nazivi naredbi i funkcija koji su već ugrađeni u *Python*.

Varijable mogu biti numeričke: cijeli brojevi ili *int* i realni brojevi ili *float*, niz znakova (*string* tip varijable) ili logički (*True* ili *False*). Tekstualne varijable uvijek navodimo u jednostrukim, dvostrukim, ili trostrukim jednostrukim navodnicima.

Nad brojevima, a i znakovima možemo izvršavati različite operacije. Brojeve možemo zbrajati (+), oduzimati (-), množiti (\*), podijeliti (/), potencirati (\*\*), tražiti ostatak dijeljenja brojeva (%) cjelobrojni ostatak dijeljenja brojeva (//). Nizove

znakova možemo spajati (+), ponavljati (\*), dohvaćati indekse (mjesto na kojem se nalazi znak) [n], dohvaćati dio niza omeđenog indeksima [n:n+1], naredba *in* provjerava nalazi li se znak u nizu, odnosno *not in* ako se ne nalazi. Logički operatori su: i (*and*), ili (*or*) i negacija (*not*).

### 3.1.3. Upisivanje i ispisivanje vrijednosti varijabli

Zanima nas kako upisati podatke. Za upis podataka se koristi operator pridruživanja jednako “=”. Na lijevoj strani stoji naziv varijable, sa znakom jednako “=” pridružujemo vrijednost ili niz znakova. Kao što vidimo na primjeru:

```
naziv_varijable=10 ili naziv_varijable="niz_znakova"
```

Treba upisati podatak i ispisati ga na ekranu: `a=5`.

Naše programe ćemo upisivati u *Python* skripti, pa ćemo se i voditi time. U skriptu upišemo `a=5`. Odabiremo izbornik *run*, pa odaberemo *run module* ili to činimo s tipkom F5. Pri pokretanju programa pojavit će se zahtjev za spremanjem (osim ako to već nismo prethodno učinili) i imenovanjem datoteke u koju ćemo spremiti program. Program ima ekstenziju *.py* pa će se u glavnom prozoru ispisati:

```
= RESTART: C://...Local/Programs/Python/Python37-32/proba.py =
```

Ova putanja nam pokazuje koju smo datoteku pokrenuli. Upišemo li varijablu *a* u glavni prozor ispisat će se broj 5. U varijablu koju smo nazvali *a* upisali smo neku vrijednost, a na dva načina možemo promijeniti vrijednost te varijable, to možemo učiniti tako da promijenimo vrijednost te varijable ako joj pridružimo neku drugu vrijednost (recimo `a=10`) ili ako izbrišemo samu varijablu iz memorije (pomoću naredbe *del a*).

Ako želimo upisati proizvoljnu vrijednost ili niz znakova (korisniku je dana mogućnost samostalnog upisivanja neke vrijednosti ili niza znakova), to činimo s naredbom *input()*. Ako znamo da ćemo upisivati (cijeli) broj, varijabli *a* ćemo pridružiti naredbu *int(input())*, u slučaju ako se radi o nizu znakova izostavljamo *int* prije *input* naredbe. Moguće je i da želimo korisnika obavijestiti koji korak ili radnju treba

napraviti, tako ispisujemo poruku unutar *input* naredbe koja će se ispisati prilikom pokretanja programa, kao npr. `a=int(input("upiši cijeli broj:"))`.

Ako želimo kontrolirati što program treba ispisati, to činimo naredbom *print()*. U naredbu *print* možemo upisati bilo koju varijablu ili niz varijabli npr. `print(a, var_1, b2)`, operacije nad varijablama ili niz operacija nad varijablama, npr. `print(a+b2, a%b)` ili tekst, npr. `print("zbroy brojeva a i b je:")` ili operacije nad nizovima znakova, npr. `print(c[0])`. Ako navodimo više elemenata unutar naredbe *print*, onda te elemente odvajamo zarezom.

Sada je poznato kako upisati i ispisati podatke, to možemo provjeriti i za *string* podacima, ako umjesto broja kojeg smo pridružili varijabli a pridružimo "niz znakova". Varijable možemo upisivati i u nizu, pazeći pritom da svakoj varijabli pridružimo točno jednu vrijednost ili niz znakova. Ako upišemo u skriptu:

```
a,b,c,d=1,2,"ne","da"  
print(a+b,",", a-b,",", a/b,",", a*b,",", a**b,",", a%b,",", a//b)  
print(c+d,",",c[0],",",d*4)
```

ispisuje:

```
3 , -1 , 0.5 , 2 , 1 , 1 , 0  
neda , n , dadadada
```

### 3.1.4. Naredbe za kontrolu toka

Naši programi se neće sastojati samo od upisivanja i računanja. Neki put će postojati određeni uvjeti koji dovode do određenog rezultata (trebat će ispisati parne brojeve, brojeve koji su djeljivi sa 7 u nekom rasponu, izračunati zbroj ili umnožak dva broja i ispisati koji je veći, ispisati sve petice u nekom nizu ocjena i sl.). To postizemo naredbama za kontrolu toka: *if (elif, else)*, *while*, *for*, *continue* i *break*.

*If* koristimo kada postoje različiti uvjeti koji se baziraju na binarnoj odluci, ako je ova tvrdnja točna izvršavaj blok naredbi inače izvršavaj drugi blok naredbi (ili n-ti blok – moguće je da postoji i više od jednog uvjeta).

*While* je naredba koja se koristi kada ne znamo koliko će se puta neki zahtjev ponavljati, dok za *for* petlju točno znamo, koristeći brojač ili nepoznanicu.

*Continue* služi tome da se blok naredbi unutar petlje nastavlja izvršavati dok je zadovoljen određeni uvjet (čiji rezultat preskače), dok *break* zaustavlja program čim se zadovolji uvjet.

Ono što moramo napomenuti jeste da se pri ispisivanju koda mora pripaziti koji dio koda pripada kojem bloku naredbi, to nam omogućuje dvotočka ":" koja slijedi nakon naredbe kontrole tijeka, te uvlačenjem (četiri *space* udaljenosti od lijevog ruba prozora skripte).

Pogledajmo na sljedećim primjerima kako se implementiraju naredbe toka. Možemo konstruirati kalkulator koji provjerava točan odgovor. Koristit ćemo naredbu *if* čija je sintaksa: *if* uvjet: blok naredbi. Upisujemo dva broja te njihov rezultat, program provjerava je li taj rezultat točan, ako jeste ispisuje tekst o točnosti, u suprotnom ispisuje da je odgovor pogrešan, operaciju koju smo odabrali je množenje, slijedi:

```
a=int(input("unesite prvi broj:"))
b=int(input("unesite drugi broj:"))
c=int(input("unesite rješenje:"))
if a*b==c:
    print("Točan odgovor!")
else:
    print("Netočan odgovor!")
```

Ispisuje:

```
unesite prvi broj:24
unesite drugi broj:3
unesite rješenje:72
Točan odgovor!
```

Ako možemo primijetiti, program podržava znakove hrvatskog jezika, tako da u pisanju uputa ili nekog niza znakova možemo koristiti i dijakritičke znakove.

Neki put će se desiti da nismo u potpunosti točno ispisali sintaksu, tada će nam program javiti pogrešku. Za program:

```
a=ivan
if a==ivan
```

```
print(a)
```

pojaviti će se prozorčić *SyntaxError: Invalid syntax*. Niz znakova "ivan" neće shvaćati kao niz znakova, jer nije označen navodnicima, pa tako niti ne može ispisati varijablu. Kada je ispravljen (crveno naznačeno je ono što je nedostajalo da bi se program pravilno izveo), program ispravno ispisuje a, pa slijedi:

```
a="ivan"  
if a=="ivan":  
    print(a)
```

U slučaju da je izostala dvotočka nakon otvaranja bloka naredbi iza naredbe *if*, označit će crvenom trakom liniju koda koju treba ispraviti i izbaciti *SyntaxError prozor*.

```
a="ivan"  
if a==ivan  
    print(a)
```

Želimo sada provjeriti nalazi li se neki broj u rasponu brojeva. To možemo učiniti pomoću naredbe *while*, njezina sintaksa slijedi: *while* uvjet: blok naredbi. Upisujemo gornju i donju granicu i broj za kojeg se pitamo nalazi li se u rasponu ili ne, u slučaju da se odabere 0 program se prekida. Slijedi:

```
a=int(input("upisi donju granicu:"))  
b=int(input("upisi gornju granicu:"))  
print("raspon je:[",a,",",b,"]")  
while 1:  
    broj=int(input("unesi broj:"))  
    if broj>=a and broj<=b:  
        print("broj je u rasponu!")  
        break  
    elif broj==0:  
        print("program je prekinut!")  
        break  
    else:  
        print("broj nije u rasponu, ponovi...")
```

Koristili smo ovaj put *while* petlju, dokle god je tvrdnja istinita izvršavaj sljedeće: ako se broj nalazi u segmentu [a, b] ispiši da se nalazi u njemu, u slučaju da smo unijeli 0 naredbom *break* prekidamo program, dok u slučaju da nije u rasponu, to nam i ispisuje.

Moguće je i da se *while* petlja vrti u beskonačno, ako nismo postavili dobro uvjete, kao na primjeru linije koda koja je označena crvenim:

```
lozinka=""
while lozinka!="mojalozinka":
    if lozinka=="mojalozinka":
        lozinka=input("Unesi ponovno lozinku:")
        print("Upisali ste točnu lozinku!")
    else:
        print("Upisali ste pogrešnu lozinku!")
```

Ako upisivanje varijable *lozinka* postavimo unutar *if* uvjeta, neće biti “lozinke” koja će biti upisana u program, pa će se beskonačno ispisivati “Upisali ste pogrešnu lozinku!”, s toga ta linija koda mora ići nakon *while* (moramo unijeti neku vrijednost!), a prije *if* uvjeta (odnosno u trećoj liniji koda).

Petlja *for* se sastoji od različitih elemenata. Koristimo brojač *i* ili broj koji se nalazi u rasponu, naredbu *in* koja označava u čemu se nalazi i naredba *range()* koja označava raspon brojeva, a obuhvaća polusegment, tako što uzima donju granicu, a gornju izuzima (npr. raspon (1,4) je niz brojeva: 1,2,3). Sintaksa je sljedeća: *for* i *in range()*: blok naredbi.

Npr.:

```
for i in range(0,3):
    print(a)
```

Ispisuje:

```
0
1
2
```

Ispišimo a="Python", b="je super" koristeći for petlju. Pokazat ćemo da različitim sintaksama dobijemo različite rezultate:

```
a="Python"
b="je super"
for i in range(0,3):
    print(a)
for j in range(0,3):
    print(b)
```

Ove dvije petlje zapisane na ovaj način programa shvaća kao svaku za sebe, te na sljedeći način ispisuje niz znakova. Ispiši prvo "Python" tri puta, pa onda "je super" tri puta:

```
Python
Python
Python
je super
je super
je super
```

Ako isti program zapišemo samo s jednim brojačem, ispisuje nizove znakova kao rečenice:

```
for i in range(0,3):
    print(a,b)
```

ispisuje:

```
Python je super
Python je super
Python je super
```

Ako uvučemo (ugnijezdimo) drugu for petlju, program će ponavljati petlju unutar petlje. Tada ćemo reći programu da ispiše četvorku nizova znakova (a, b, b, b) koja će se ispisati tri puta:

```
for i in range(0,3):
    print(a)
    for j in range(0,3):
        print(b)
```

Ispis će biti:

```
Python
je super
je super
je super
Python
je super
je super
je super
Python
je super
je super
je super
```

Pogledajmo koja je razlika između *continue* i *break* na primjeru. U programu je moguće i ostavljati komentare, ali kako bi ih program razlikovao od onoga što je u kodu, stavljamo ili znak # ili kao tekst navodimo u jednostrukim, dvostrukim ili trostrukim jednostrukim navodnicima:

```
#continue primjer

for i in range(1,21):    #za neki broj u rasponu od 1 do 21
    if i%5==0:          #dok je broj djeljiv s 5
        continue       #preskoči brojeve djeljive s 5
    else:               #inače
        print(i,"nije djeljiv s 5!") #ispiši brojeve koji nisu djeljivi s 5
i ispiši da nije djeljiv s 512
```

---

<sup>12</sup> svaki razlomljeni komentar inače ostaje u istom redu i povezan je s #.



```

#break primjer

for i in range(1,21):           #za neki broj u rasponu od 1 do 21
    if i%5==0:                 #dok je broj djeljiv s 5
        print(i, "je djeljiv s 5!")#ispiši broj koji je djeljiv s 5 i
ispiši da je djeljiv s 5
        break                  #prekini pretragu
    else:                       #inače
        print("nije djeljiv s 5!") #ispiši poruku i ispiši da nije djeljiv
s 5

```

Program s *continue* naredbom će pretražiti cijeli raspon i ispisat će svaki broj koji nije djeljiv s 5, dok pri *break* naredbi zaustavlja pretragu čim pronade prvu peticu. Za *continue* ispisat će:

```

1 nije djeljiv s 5!
2 nije djeljiv s 5!
3 nije djeljiv s 5!
4 nije djeljiv s 5!
6 nije djeljiv s 5!
7 nije djeljiv s 5!
8 nije djeljiv s 5!
9 nije djeljiv s 5!
11 nije djeljiv s 5!
12 nije djeljiv s 5!
13 nije djeljiv s 5!
14 nije djeljiv s 5!
16 nije djeljiv s 5!
17 nije djeljiv s 5!
18 nije djeljiv s 5!
19 nije djeljiv s 5!

```

Za *break* će ispisati:

```

1 nije djeljiv s 5!
2 nije djeljiv s 5!
3 nije djeljiv s 5!
4 nije djeljiv s 5!
5 je djeljiv s 5!

```

Da smo htjeli ispisati samo brojeve djeljive s 5, koristili bismo samo jednu naredbu *if* s uvjetom djeljivosti broja s 5, te *print* za ispis tih brojeva.

Ishodi koji dovode do finalnog ishoda će omogućiti da učenik: koristi dijagram tijeka i pseudo-kod za sastavljanje algoritma; prepoznaje različite tipove podataka; upisuje i ispisuje jednostavne tipove podataka nad kojima vrši aritmetičke operacije zbrajanja, oduzimanja, množenja, dijeljenja, potenciranja, traženja ostataka i cjelobrojnog dijela ostatka; upisuje i ispisuje nizove znakova koristeći se indeksima, spaja nizove znakova i multiplicira ih; koristi naredbe za kontrolu toka: uvjetno grananje, naredbe ponavljanja, naredbu prekida i nastavljanja programa.

### **3.2. Izazovi pri ostvarivanju ishoda**

Iako imamo nekakav idealizirani put poučavanja do ostvarivanja ishoda, možemo se susresti s različitim oblicima prepreka ili ometajućih faktora u njihovom ostvarivanju. Takve izazove možemo podijeliti u tri kategorije, a odnose se na nastavnika, učenika i resurse koji su potrebni za ostvarivanje nastave.

Nastavnik može tako zbog svojih karakteristika ometati daljnje napredovanje učenika u nastavi informatike, a to mogu biti: niska stručnost na području predmeta kojeg poučava, niska odgojno-obrazovna kompetentnost u održavanju nastave, a odnosi se na pedagoške kompetencije, način poučavanja, organiziranja nastave, organiziranja sadržaja, nepripremljenost i sl., osobni stavovi i uvjerenja kao rodni stereotipi, socioekonomski stereotipi ili neki drugi oblici diskriminacije, nezainteresiranost i niska motivacija, loše komunikacijske vještine, niska prilagodljivost na nova znanja, tehnologije i oblike poučavanja, loša suradnja s nadređenima, institucijama odgovornima za odgojno-obrazovni proces i stručnjacima predmetnog područja te neke osobine ličnosti kao nisko samopouzdanje i samopoštovanje.

Ono što bi moglo ometati učenika u napredovanju kroz nastavu informatike je: nezainteresiranost, niska motivacija, nepripremljenost za nastavu kao nevježbanje na računalu i nepoznavanje teorijskog sadržaja itd., nerazumijevanje predmetnog sadržaja, školski stresori (odnosi u razredu, s nastavnikom, školske obveze), sporije napredovanje po kognitivnim razinama, nekvalitetno predznanje (pri prelasku u viši

razred ili promjenu škole), kao i navedene osobine ličnosti koje vrijede za nastavnika.

Osim nastavnika i učenika, u nastavnu jednadžbu ulaze i određeni resursi, a to mogu biti: problemi s mrežom, nedostatak računala za samostalni rad, loša infrastruktura i logistika za tehničku podršku, nedostupnost ili neodržavanje određenih softvera koji su potrebni za ostvarivanje nastave programiranja.

Pored navedenih primjera možemo navesti i one iz britanskog istraživanja koje se 2014. zainteresiralo za nastavu informatike prije nego što je postala obvezan predmet, upravo iz razloga, jer su se nastavnici susreli s poteškoćama, pogotovo nakon što se prelazilo iz formata komunikacijsko-informacijskih tehnologija na područje računalnog razmišljanja i programiranja. Postavljalo se pitanje kako prijeći na način razmišljanja za koji se zahtjeva istovremeno korištenje dedukcije i indukcije.

Navedimo neka opažanja iz spomenutog istraživanja<sup>13</sup>: Učenik bi trebao savladati gradivo aktivno sudjelujući u procesu učenja i poučavanja, istražujući putem pogrešaka rješava problem iz stvarnog života koji je primjenjiv na računalu, u paru ili grupi razvija logičko mišljenje i sekvencijalno razmišljanje i razvija upornost u postizanju dobivanja algoritma koji je upotrebljiv. Pored ostalog, potrebne su mu kompetencije koje se odnose na pismenost, matematičko znanje i razmišljanje, nov pristup učenju - usklađen s računalnim razmišljanjem kojeg čine: apstrakcija, algoritamsko razmišljanje, generalizacija i evaluacija.

Određene strategije koje su se do sada koristile djelomično su zadovoljavale postizanje ishoda koji su se očekivali od učenika, pa se tako smatra da bi strategije za uspješno učenje sadržavale:

1. učenje bez računala - aktivnosti nevezane za računalo
2. suradničko učenje - rad u paru ili grupi ili mentorstvo
3. računalno razmišljanje
4. kontekstualno učenje - međupredmetno povezivanje i povezivanje sa svakodnevnim životom
5. *scaffolding* - analiza i opis koda putem traženja pogrešaka u kodu (*debugging, troubleshooting*)

---

<sup>13</sup> Sentence, S. & Csizmadia, A. (2013.) Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Educ Inf Technol* (2017) 22:469–495.

## Zaključak

Postojanje predmeta informatike nam omogućuje integrirati način razmišljanja koji nam omogućuje istraživanje, različite oblike kritičkog mišljenja, logike, algoritamskog mišljenja, matematičkog razmišljanja i kreativnost u svakodnevni život. Svi mi posjedujemo neki potencijal, koji dolazi do izražaja i prepoznaje se u nastavi informatike. Manipulacija podacima, tekstom, slikom, zvukom, stvaranje različitih alata, aplikacija koje nam mogu pomoći, učiniti život zabavnijim, lakšim i zanimljivijim. Sve to možemo dobiti iz područja znanja koje objedinjuje informatika.

Svaka država ima svoj tempo i percepciju o ovom području, pa se neravnomjerno razvija svaki dio civilizacije. Neki su prepoznali računalno razmišljanje kao osnovu koja nam može pomoći u svakodnevnom životu, na osobnoj i profesionalnoj razini, pritom pazeći na korištenje određenih odgojno-obrazovnih načela. Kroz ovo područje stupamo na teritorij mogućnosti i ostvarivanja i ako ugradimo u naše vrijednosti i ovaj nekonvencionalni način razmišljanja, možda nas to dovede do nečega što će nas uistinu gurnuti još dalje.

Predmetni kurikulum informatike dakako treba prepustiti stručnjacima i onima koji žele ulagati u sebe, a pritom biti usmjereni i na učenika. Iako postoji dugotrajna tradicija školstva, zbog količine znanja i promjene tehnologija, potrebno je doslovno ažuriranje na svakodnevnoj bazi kako bi se svi sudionici odgojno-obrazovnog procesa mogli snaći u instituciji kao što je škola. Teorijski i praktični aspekt nam govori da se još uvijek treba raditi na tome da se poboljša bilo koji dio odgojno-obrazovnog procesa nastave, naravno pritom uzimajući u obzir i učenika i nastavnika i sadržaj koji ga obuhvaća.

Dijelom obrazovanje ovisi o političkoj klimi i ekonomskoj situaciji u državi, ali i o onima koji to obrazovanje konzumiraju ili direktno ili indirektno sudjeluju u tom procesu. Trebamo sami sebi dozvoliti da učinimo obrazovanje prioritetom i vratimo u bolje sutra, na podjednak način kao i što su ona područja ljudskog djelovanja i znanja koja prepoznajemo kao potrebna i posebna, a kroz kvalitetno ulaganje u njih ona neće imati samo lokalni utjecaj već i utjecaj na svjetskoj razini.

## Literatura

1. Anderson, L. W., Krathwohl, D. R., [et al.] (2001.) *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. Allyn & Bacon. Boston: MA (Pearson Education Group) Dostupno na: [https://quincycollege.edu/content/uploads/Anderson\\_and\\_Krathwohl\\_Revised\\_Blooms\\_Taxonomy.pdf](https://quincycollege.edu/content/uploads/Anderson_and_Krathwohl_Revised_Blooms_Taxonomy.pdf) [Pristupljeno: 09.11.2017.]
2. Baranović, B. (2015.) *Školski kurikulum: Teorijski i praktični aspekti*. Zagreb: Institut za društvena istraživanja u Zagrebu. Dostupno na: <http://idiprints.knjiznica.idi.hr/259/1/%C5%A0kolski%20kurikulum%20-%20theoretical%20chapters.pdf> [Pristupljeno: 10.09.2019.]
3. Batagelj, V. [et al.] (2002.) *Računalništvo*. Kočevje: Tiskarna Kočevski tisk d. d. Dostupno na: [https://www.gov.si/assets/ministrstva/MIZS/Dokumenti/Osnovna\\_sola/Ucni\\_nacrti/izbirni/3\\_letni\\_lahko\\_krajsi/Racunalnistvo\\_izbirni.pdf](https://www.gov.si/assets/ministrstva/MIZS/Dokumenti/Osnovna_sola/Ucni_nacrti/izbirni/3_letni_lahko_krajsi/Racunalnistvo_izbirni.pdf) [Pristupljeno: 05.09.2017.]
4. Brennan, K., & Resnick, M. (2012.) New frameworks for studying and assessing the development of computational thinking. *In Proceedings of the 2012 Annual Meeting of the American Educational Research Association*. Vancouver, Canada. Vancouver, Canada
5. Galešev, V. [et al.] (2018.) *Informatika i računalstvo*. 2. izdanje. Zagreb: Sysprint. Dostupno na <https://sysprint.hr/eudzbenici/infGIM2018/#> [Pristupljeno: 27.09.2019.]
6. Hruška, M. (2018.) *Osnove programiranja (Python)*. Zagreb: Srce. Dostupno na: [https://www.srce.unizg.hr/files/srce/docs/edu/osnovni-tecajevi/d450\\_polaznik.pdf](https://www.srce.unizg.hr/files/srce/docs/edu/osnovni-tecajevi/d450_polaznik.pdf) [Pristupljeno: 27.09.2019.]
7. Kemp, P. (2014.) *Computing in the national curriculum*. Bedford: Newnorth Print, Ltd. Dostupno na: [https://www.computingatschool.org.uk/data/uploads/cas\\_secondary.pdf](https://www.computingatschool.org.uk/data/uploads/cas_secondary.pdf) [Pristupljeno: 09.01.2018.]
8. Krajnc, R. [et al.] (2013.) *Računalništvo*. Ljubljana: Ministrstvo za izobraževanje, znanost in šport: Zavod RS za šolstvo. Dostupno na:

[http://www.fmalgaja.si/files/2017/05/Racunalnistvo\\_izbirni\\_neobvezni.pdf](http://www.fmalgaja.si/files/2017/05/Racunalnistvo_izbirni_neobvezni.pdf)

[Pristupljeno: 28.12.2017.]

9. Lavrnja, I. (1996.) *Poglavlja iz didaktike*. Rijeka: Pedagoški fakultet Rijeka.

10. Marius -Costel, E. (2010.) The didactic principles and their applications in the didactic activity. *Sino -US English Teaching*. Vol. 7. (Br.9). str. 24 -34. Dostupno na: <https://files.eric.ed.gov/fulltext/ED514739.pdf> [Pristupljeno: 25.09.2019.]

11. Poljak, V. (1988.) *Didaktika*. Zagreb: Školska knjiga.

12. Previšić, V. (2007.) *Kurikulum: teorije, metodologija, sadržaj, struktura*. Zagreb: Školska knjiga.

13. Sentence, S. & Csizmadia, A. (2013.) Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Educ Inf Technol* (2017) 22:469-495. Dostupno na: <https://link.springer.com/content/pdf/10.1007%2Fs10639-016-9482-0.pdf> [Pristupljeno: 06.10.2018.]

## **Mrežni izvori**

14. (2013.) *Digital competence framework*. <https://ec.europa.eu/jrc/en/digcomp/digital-competence-framework> [Pristupljeno: 26.04.2018.]

15. (2006.) *European reference framework*. Dostupno na: <https://www.erasmusplus.org.uk/file/272/download> [Pristupljeno: 25.09.2018.]

16. *KS5 Computer Science*. Dostupno na: <https://www.utcleeds.co.uk/curriculum/ks5/ks5-computer-science/> [Pristupljeno: 06.10.2019.]

17. (2018.) *Kurikulum za nastavni predmet informatike za osnovne škole i gimnazije*. Zagreb: Ministarstvo obrazovanja i športa. Dostupno na: [https://narodne-novine.nn.hr/clanci/sluzbeni/2018\\_03\\_22\\_436.html](https://narodne-novine.nn.hr/clanci/sluzbeni/2018_03_22_436.html) [Pristupljeno: 18.09.2019.]

18. (2011.) *Nacionalni okvirni kurikulum*. Zagreb: Ministarstvo obrazovanja i športa. Dostupno na: [http://mzos.hr/datoteke/Nacionalni\\_okvirni\\_kurikulum.pdf](http://mzos.hr/datoteke/Nacionalni_okvirni_kurikulum.pdf) [Pristupljeno: 09.11.2017.]

19. (2013.) *The national curriculum in England*. London: Department for Education

Dostupno na:

[https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment\\_data/file/381754/SECONDARY\\_national\\_curriculum.pdf](https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/381754/SECONDARY_national_curriculum.pdf) [Pristupljeno:

11.01.2018.]

## **Sažetak**

Organizirani i strukturirani okvir za učenje i poučavanje zovemo kurikulumom. Kurikulum je razvojni dokument koji sadrži smjernice, ciljeve i ishode za učenike i studente koji su nužni za napredovanje od najnižih do najviših razina u procesu obrazovanja. Odgojno-obrazovni ciljevi nam omogućuju da se usmjerimo na rezultate učenja, a one su operacionalizirane vrijednosti koje se iskazuju na obrazovnim razinama, a povezani su s očekivanim rezultatima sa stupnjevima postignuća koje zovemo ishodima. Načelo je početak ili osnova od koje se polazi i koju valja uvažavati u različitim područjima ljudske djelatnosti. Didaktička načela su usmjerena na nastavni proces, sadržaj, proces učenja i usvajanja informacija, a neka na samog učenika. Nacionalni predmetni kurikulumi iz informatike Republike Hrvatske, Republike Slovenije i Velike Britanije integriraju na različite načine didaktička načela. Na primjeru ishoda za 1. razred prirodoslovno-matematičke gimnazije Republike Hrvatske i domene računalnog razmišljanja i programiranja ponuđen je sadržaj koji dovodi do ishoda i navedeni su izazovi do kojih može doći u tom procesu.

## **Ključne riječi**

kurikulum, nacionalni okvirni kurikulum, predmetni kurikulum, školski kurikulum, didaktička načela, računalno razmišljanje, programiranje, informatika



## **Abstract**

A curriculum is an organized and structured framework for learning and teaching. It's a developmental document that contains guidelines, aims and outcomes for students which are necessary to progress from the lowest to the highest levels of the educational process. The educational aims are operationalized values that are expressed through the educational levels and are associated with the expected results of the levels of achievement which are called the outcomes. These aims allow us to focus on the the learning objectives.

The principle is the beginning or the basis which should be valued through the various fields of human activity. The didactic principles are focused on the various parts of the educational process such as the teaching process, the content, the learning process and the acquisition of information and some are focused on the student. A computer science curriculum of the Republic of Croatia, the Republic of Slovenia and the United Kingdom integrate the didactic principles in different ways. In the context of highschool first grade in the Republic of Croatia and the domain of computational thinking and programming shows the outcome of the content that leads to its realization and outlines the challenges that may arise in the process.

## **Key words**

curriculum, national curriculum, subject-centered curriculum, school curriculum, didactic principles, computational thinking, programming, computer science