

Obrazac Metodičkih preporuka za ostvarivanje odgojno-obrazovnih ishoda predmetnih kurikuluma i međupredmetnih tema za osnovnu i srednju školu	
OSNOVNI PODATCI	
Ime i prezime	Zoran Hercigonja
Zvanje	mag.educ.inf.
Naziv škole u kojoj ste trenutano zaposleni	Druga gimnazija Varaždin
Adresa elektroničke pošte	zoran.hercigonja@gmail.com
Naslov Metodičkih preporuka	Programiranje (Python) – vizualizirano usvajanje računalnog razmišljanja
Predmet (ili međupredmetna tema)	Informatika – Petlje-Primjena for petlje
Za međupredmetnu temu navesti u okviru kojeg nastavnoga predmeta, sata razrednika ili izvannastavne aktivnosti se izvodi.	
Razred	1.razred prirodoslovno matematičke gimnazije
OBVEZNI ELEMENTI	
Odgojno-obrazovni ishod (oznaka i tekst iz kurikuluma predmeta ili međupredmetnih tema objavljenih u NN)	B. 1. 1 analizira problem, definira ulazne i izlazne vrijednosti te uočava korake za rješavanje problema B. 1. 2 primjenjuje jednostavne tipove podataka te argumentira njihov odabir, primjenjuje različite vrste izraza, operacija, relacija i standardnih funkcija za modeliranje jednostavnoga problema u odabranome programskom jeziku B. 1. 3 razvija algoritam i stvara program u odabranome programskom jeziku rješavajući problem uporabom strukture grananja i ponavljanja
Tijek nastavnog	Uvod→Glavni dio→Završni dio



sata	
Opis svih aktivnosti (što rade učenici, a što učitelj/nastavnik)	<p>Uvod</p> <p>Nastavnik uvodi motivacijski primjer korištenja for petlje u stvarnom životu. S učenicima utvrđuje potrebu za programskom strukturom ponavljanja uvodeći primjere funkcioniranja ponavljajućeg uključivanja i isključivanja svjetala na semaforu u prometu. Primjer pametnih obavijesti u mobilnim uređajima.</p> <p>Nastavnik u raspravi s učenicima utvrđuje konceptualnu funkcionalnu shemu for petlje.</p> <p>Nastavnik obrazlaže funkcionalne dijelove for petlje u programskom jeziku Python i povezuje ih s prethodnom konceptualnom shemom.</p> <p>Glavni dio sata - frontalni i grupni rad</p> <p>-Demonstracija</p> <p>Nastavnik demonstrira inicijalizaciju for petlje i parametara te objašnjava funkcionalnost i svrhu pojedinih dijelova for petlje.</p> <p>Učenici prate demonstraciju i postavljaju pitanja.</p> <p>Nastavnik demonstrira rješenja problemskog zadatka korak po korak koristeći Pythontutor- online visualization s ciljem što boljeg predočavanja funkcionalnosti for petlje.</p> <p>Učenici prate demonstraciju i usvajaju algoritam funkcioniranja for petlje.</p> <p>-Vođena vježba</p> <p>Nastavnik definira problemski zadatak.</p> <p>Zajedno s učenicima analizira tekst zadatka te osmišljenim sustavom potpitanja zajedno s učenicima konstruira rješenje u obliku dijagrama tijeka. Za dijagram tijeka koristi Flowgorithm aplikaciju za stvaranje interaktivnog dijagrama tijeka.</p> <p>Učenici sudjeluju u raspravi i konstrukciji rješenja u Flowgorithm aplikaciji.</p> <p>Učenici zajedno s nastavnikom konstruiraju programsko rješenje preko Pythontutor vizualizatora. Nastavnik nakon svakog koraka u vizualizatoru obrazlaže događaj petlje. Učenici prate i po potrebi postavljaju pitanja.</p> <p>-Samostalan rad 1. korak</p> <p>Nastavnik zadaje problemski zadatak s poveznicom iz stvarnog života</p> <p>Zajedno s učenicima proanalizira zadatak i utvrđuje teorijsku osnovu rješenja te daje smjernice za rješavanje.</p> <p>Učenici samostalno preko Flowgorithm interaktivnog dijagrama tijeka kreiraju rješenje.</p>



	<p>Nastavnik obilazi učenika i prati njihov rad te po potrebi intervenira.</p> <p>Provjerava pojedinačan rješenja u Flowgoritmu. Zatim prikazuje rješenje također u Flowgoritmu.</p> <p>Učenici po potrebi korigiraju svoje rješenje u Flowgoritmu.</p> <p>-Samostalan rad 2. korak</p> <p>Učenici prevode Flowgorithm rješenje u programsko rješenje pomoću Pythontutora. Učenici prate korake izrade vlastitog programa u interaktivnom sučelju. Po potrebi pozivaju nastavnika i postavljaju pitanja.</p> <p>Nastavnik prati rad učenika i po potrebi intervenira. Po završetku izrade programskog rješenja, prikazuje rješenje i obrazlaže ključne korake u izradi preko Pythontutor interaktivne vizualizacije.</p> <p>-Grupni rad („natjecateljske aktivnosti“)</p> <p>Učenici se podijele u grupe po četiri.</p> <p>Nastavnik zadaje problemski zadatak sa smjericama. Učenici samostalno u grupi razrađuju problemski zadatak najprije u Flowgoritmu, a zatim u Pythontutoru.</p> <p>Sve grupe dobivaju isti zadatak. Grupa koja prva točno riješi problemski zadatak, dobiva ocjenu iz rubrike Digitalni sadržaji i suradnja.</p> <p>Nastavnik zadaje vrijeme rješavanja od 10 minuta.</p> <p>Nastavnik prati rad učenika i pazi na vrijeme. Učenici u grupama međusobno raspravljaju i konstruiraju rješenje.</p> <p>Grupa koja je u zadano vrijeme prva i točno riješila zadatak, dobiva ocjenu u e-dnevnik.</p> <p>Nakon ocjenjivanja, nastavnik pokazuje konačno rješenje i raspravlja s učenicima.</p> <p>Završni dio sata</p> <p>Nastavnik uvodi kratko ponavljanje gradiva i zadaje zadatak za domaću zadaću s uputama za rješavanje.</p>
<p>Sadržaji koji se koriste u aktivnostima</p>	<p>Pythontutor online vizualizator</p> <p>Flowgorithm interaktivni dijagram tijeka</p> <p>Powerpoint prezentacija</p> <p>Programski primjeri iz knjiga</p> <p>Rješavanje problema programiranjem u Pythonu,</p> <p>Uvod u programski jezik Python</p>



	Online knjiga: Vizualizacija postupaka rješavanja programskih zadataka u programskom jeziku Python
Primjeri vrednovanja za učenje, vrednovanja kao učenje ili naučenog uz upute	<p>Vrednovanje je izvedeno kroz grupni rad u izradi rješenja problemskog primjera tijekom samog nastavnog sata da bi se provjerila usvojenost, razumijevanje i praćenje napretka. Učenik će biti vrednovan kroz izradu praktičnog (programskog) primjera. Vrednovanje pripada kategoriji vrednovanja za učenje.</p> <p>Učenici nisu bili opterećeni strahom pred ispitivanjem. U tijeku nastavnog sata i izrade programskih rješenja, pokazali su znanje i umijeće računalnog razmišljanja. Vrednovanje njihovog znanja i umijeća poslužilo je za utvrđivanje njihove dosadašnje razine znanja i umijeća te poticaj za daljnje učenje.</p>
Razrađeni problemski zadaci, zadaci za poticanje kritičkog razmišljanja, kreativnosti i/ili istraživački zadaci; ovisno o predmetu i nastavnoj temi	<p>Problemske zadatke treba razraditi po principu od najlakšeg prema najtežem, od zadatka za usvajanje funkcionalnog koncepta osnovne for petlje do samostalnog kreativnog rješavanja složenih problemskih zadataka.</p> <p>Problemski zadaci u tijeku nastavnog sata vezani su uz konkretne primjere koji su vrlo bliski učenicima i mogu ih razumjeti te primijeniti u kontekstu svojeg života.</p> <p>Primjeri : Nastavnik ispituje učenike na satu informatike. Ispitati će prvih n učenika. Svakog nastavnog sata u svakom razredu nastavnik odabire neki prirodan broj n koji će poslužiti kao oznaka za broj učenika učenika. Nakon što nastavnik odabere neki prirodan broj n (npr. 5), iz cjelokupnog popisa učenika ispisuje prvih pet učenika.</p> <p><i>Ovaj problemski zadatak je ovisan o predmetu odnosno o nastavnoj temi i kontekstu u kojem se učenik trenutno nalazi kao dio nastavnog procesa. Učenik će stvarni primjer iz njegovog života i predmeta, učiniti programski i računalni mogućim i rješivim. Ujedno ova vrsta zadatka potiče kritičko razmišljanje i proširivanje programskog rješenja. Učenik bi mogao postaviti pitanje: „Da li je potrebno omogućiti nastavniku određeni broj pokušaja prilikom biranja prirodnog broja.“ ili „Što učiniti ako nastavnik ne unese prirodan broj?“ ili „Možemo li nastavnikov unos ograničiti na nekakav raspon vrijednosti?“</i></p> <p>Primjenjujući načelo postupnosti i pozivajući se na interdisciplinarnost predmeta i predmetnih područja, sljedeći primjer oslanja se na istraživački rad. Postupnost se ogleda u primjeni naučene programske strukture u novo usvojenoj programskoj strukturi (npr. u for petlju se uvodi if-else odluka s prethodnog nastavnog sata)</p> <p>Primjer: Napisati program za učitavanje temperature zraka u prethodnih deset dana te ispisati srednju temperaturu iznad nule i srednju temperaturu ispod nule.</p> <p><i>Interdisciplinarnost podrazumijeva prije svega povezivanje matematike te fizike s informatikom. Potrebno je poznavati pojam srednje vrijednosti odnosno aritmetičke sredine i znati izraditi algoritam računanja iste. Također potrebno</i></p>



	<i>je povezati i pojam ljestvice temperature zraka zbog unošenja vjerodostojnih vrijednosti.</i>
DODATNI ELEMENTI¹	
Poveznice na više odgojno-obrazovnih ishoda različitih predmeta ili očekivanja međupredmetnih tema	Matematika A.1.2., B.1.2., C.1.3. Fizika ABCD 1.8.
Aktivnost u kojima je vidljiva interdisciplinarnost	<p>Poznavanje matematičkih algoritama, formula i pravila za zbrajanje, množenje, dijeljenje i potenciranje potencija, za kvadrat zbroja i razlike i za razliku kvadrata te primjena za pojednostavnjivanje izraza i povezivanje s problemima iz drugih područja i života korelira s informatikom prilikom konstrukcije i kreiranja programskih algoritama pomoću for petlje za navedene operacije. Vrlo dobro poznavanje matematičkih pravila koristi i doprinosi konstrukciji programskog rješenja (algoritma) pomoću for petlje.</p> <p>Povezanost fizike s informatikom iz ishoda ABCD 1.8. vidljiva je u stečenom znanju vizualizacije situacije u zadanom problemu koji se rješava programski. Konstruiranje plana rješavanja problema, povezano je s planiranom izradom dijagrama tijekom postupka rješavanja problema. Vrednovanje postupka i rješenja, potrebno je za utvrđivanje točnosti i ispravnosti postupka i rješenja.</p> <p>Grupno rješavanje problemskog zadatka uključuje i poznavanje drugih predmeta sociologija i psihologija prije svega za komunikaciju unutar same grupe, podjelu dužnosti i zadataka.</p>
Aktivnosti koji obuhvaćaju prilagodbe za učenike s teškoćama	
Aktivnosti za motiviranje i rad s darovitim učenicima	Uključiti alate i tehnologije za interaktivniji rad i mogućnost studioznijeg uvida u algoritam rješavanja zadataka radi razvijanja kreativnosti u osmišljavanju rješenja. Poticati učenike da isti zadatak riješe na više načina kako bi razvili širi kut razumijevanja i razmišljanja na više načina. Zainteresirati darovite učenike

¹ Sastavni elementi prijave koji omogućuju dodanu vrijednost provedbi javnog poziva. Nisu obavezni, ali nose dodatne bodove u skladu s kriterijima procjene Metodčkih preporuka.



	za samostalno istraživanje, traženje i učenje preko interaktivnih web tečajeva čime provjeravaju i samostalno bilježe svoju razinu osobnog dostignuća i napredovanja (primjer web tečaja Codecademy).
Upute za kriterijsko vrednovanje kompleksnih i problemskih zadataka i/ili radova esejskoga tipa	
Projektne zadaci (s jasnim scenarijima, opisima aktivnosti, rezultatima projekta, vremenskim okvirima)	<p>Naziv projektnog zadatka: AUTENTIFIKACIJA KORISNIKA</p> <p>Projektan zadatak namijenjen je učenicima koji žele istraživati. Projekt povezuje prethodno stečena znanja, trenutno znanje te novostečeno znanje temeljem samostalnog istraživanja od strane učenika. Svi zainteresirani učenici za izradu projektnog zadatka koriste stečeno znanje for petlje te ranije naučenih programskih struktura kao što je struktura odluke elif i if-else. Učenici također primjenjuju znanje varijabli, tipova podataka te operatora. Također učenici trebaju sami temeljem stečenog znanja for petlje istražiti funkcionalnost while petlje za izradu projektnog zadatka. Na taj način učenik je potaknut na istraživačko učenje i samostalnost u učenju i traženju rješenja.</p> <p>Vremenski rok za izradu programskog rješenja je dva tjedna brojeći od dana zadavanja projektnog zadatka. Učenici projektno rješenje mogu razvijati kod kuće, ali i u školi koristeći se školskom opremom i literaturom.</p> <p>Nakon isteka zadanog vremena učenici projektna rješenja prezentiraju na satu informatike gdje će ujedno biti i ocijenjena. Kriteriji ocjenjivanja su: točnost i funkcionalnost programskog rješenja.</p> <hr style="border: 1px solid black; margin: 10px 0;"/> <p>Učenici dobivaju detaljni scenarij rješavanja projektnog zadatka.</p> <ul style="list-style-type: none"> • Napišite program koji će omogućiti unos podataka o učeniku • Prije unosa podataka o učenicima, korisnik treba proći autentifikaciju (provjera korisničkog imena i pripadajuće lozinke) • AUTENTIFIKACIJA KORISNIKA: <ul style="list-style-type: none"> ○ korisnik unosi korisničko ime i lozinku u program <ul style="list-style-type: none"> ▪ korisničko ime treba biti formata ime.prezime ▪ lozinka treba imati ukupno 6 znakova (obavezno kombinacija velikih i malih slova, brojeva i specijalnih znakova) ○ program provjerava točnost korisničkog imena i lozinke ○ ukoliko jedno od njih (korisničko ime ili lozinka) nije točno program treba omogućiti ponovni unos korisničkog imena i



lozinke ****ukupno tri puta****

- Kada je korisničko ime ili lozinka netočna ispisati poruku:
 - „Pokušajte ponovno“
 - „Imate još **N** pokušaja“ (na mjestu **N** treba ispisati koliko je korisniku još pokušaja preostalo od ukupno 3 pokušaja)
- Kada korisničko ime i lozinka budu točno uneseni ispisati poruku:
 - „Podaci su valjani“
 - „***DOBRODOŠLI***“
- **UNOS NOVOG UČENIKA:**
 - Nakon uspješne autorizacije, program treba omogućiti unos podataka o novome učeniku, zatim ispis podataka o unesenom učeniku i izlazak iz programskog sučelja
 - Nakon što je korisnik prošao autentifikaciju te nakon što se ispisala poruka o valjanosti podataka s pozdravnom frazom, **potrebno je prikazati opcije izbornika**
 - primjer izbornika
 - 1.Unos podataka o učeniku
 - 2. Ispis podataka o unesenom učeniku
 - 3. Izlazak iz programa
 - Kada korisnik odabere opciju *1.Unos podataka o učeniku* program treba omogućiti unos sljedećih podataka za učenika sa zadanim kriterijima
 - **OIB** → brojevi ≥ 1 & ≤ 10
 - **IME** → znakovni tip podataka
 - **PREZIME** → znakovni tip podataka
 - **BORAVIŠTE** → znakovni tip podataka
 - Kada korisnik odabere opciju 2. Ispis podataka o unesenom učeniku ispisuju se sljedeći podaci:
 - **OIB** → brojevi
 - **IME** → znakovni tip podataka
 - **PREZIME** → znakovni tip podataka
 - **BORAVIŠTE** → znakovni tip podataka
 - Opcija *Izlazak iz programa* treba omogućiti potpuni izlazak iz cijelog programa, a ne povratak u glavni izbornik ili na početak forme za autentifikaciju korisnika (**koristiti break**)



	<p>PRIMJER ISPISA:</p> <pre>***** ***SUČELJE ZA AUTORIZACIJU KORISNIKA***** ***UNOS KORISNIČKOG IMENA I LOZINKE***** ***** Unesite korisničko ime pero.peric Unesite lozinku D394s! ----- Podaci su valjani ****DOBRODOŠLI**** Izbornik 1.Unos podataka o učeniku 2.Ispis posljednjeg učenika 3.Izlazak iz programa Vaš odabir</pre> <pre>***** ***SUČELJE ZA AUTORIZACIJU KORISNIKA***** ***UNOS KORISNIČKOG IMENA I LOZINKE***** ***** Unesite korisničko ime pero.peric Unesite lozinku 1234 ----- Pokušajte ponovno Imate još 3 pokušaja Unesite korisničko ime </pre> <pre>OIB 3 Ime Pero Prezime Peric Boraviste Makarska Izbornik 1.Unos podataka o učeniku 2.Ispis posljednjeg učenika 3.Izlazak iz programa</pre>
<p>Poveznice na multimedijske i interaktivne sadržaje</p>	<p>http://www.flowgorithm.org/</p> <p>http://www.pythontutor.com/visualize.html#mode=display</p> <p>http://www.znanje.org/knjige/computer/python/python_abc_zadaci.htm</p> <p>http://info-lux.net/python-vjezbe-grananja-petlje/</p> <p>https://pythonosnoveblog.wordpress.com</p>
<p>Prijedlozi vanjskih izvora i literature</p>	<p>Vizualizacija postupaka rješavanja programskih zadataka u programskom jeziku Python (https://issuu.com/zoranhercigonja/docs/vizualizacija_postupaka_rje_avanja)</p> <p>Hercigonja Z., Uvod u programski jezik Python, 2018.</p> <p>L.Budin, P. Brođanac, Z. Markučić, S. Perić-Rješavanje problema programiranja u Pythonu, udžbenik za prirodoslovno-matematičke gimnazije</p>