

Pregled

- 1 Uvod
- 2 Nastava zasnovana na ishodima
- 3 Osnovni pojmovi i ideje
- 4 Raspoloživi nastavni materijali
- 5 Algoritmi i primeri
- 6 Između dva dela obuke
- 7 Zaključci

Pregled

- 1 Uvod
 - O projektu
 - O ovoj obuci
 - Uvođenje informatike i programiranja u osnovnu školu
- 2 Nastava zasnovana na ishodima
- 3 Osnovni pojmovi i ideje
- 4 Raspoloživi nastavni materijali
- 5 Algoritmi i primeri
- 6 Između dva dela obuke
- 7 Zaključci

Uvod

Nastava zasnovana na ishodima
Osnovni pojmovi i ideje
Raspoloživi nastavni materijali
Algoritmi i primeri
Između dva dela obuke
Zaključci

O projektu

O ovoj obuci

Uvođenje informatike i programiranja u osnovnu školu

O projektu

Ova obuka održava se u okviru projekta

Podrška nastavi informatike u OŠ

Projekat organizuje

Fondacija Petlja

Pomoć i podršku projektu pružaju

Vlada Republike Srbije

- **Ministarstvo Prosvete, Nauke i Tehnološkog razvoja**
- **Ministarstvo Trgovine, Turizma i Telekomunikacija**

IT kompanija

Microsoft

Neprofitno udruženje vodećih IT kompanija

Inicijativa Digitalna Srbija

Univerzitet u Beogradu

Matematički fakultet

Akcioni plan Vlade Srbije o informatici

Svi su složni da je potrebna nova akcija

Vlada, IT kompanije, univerziteti, nastavnici, đaci, ...

Akcioni plan predviđa

- Ulaganja u infrastrukturu
- Povećanje upisnih kvota na studijama informatike
- Prekvalifikaciju stručnjaka
- Unapređenje poslovnog okruženja
- **Uvođenje informatike i programiranja u osnovnu školu**

Kome je to važno?

Važno je ...

- 1 ... za ekonomiju Srbije
- 2 ... našim đacima i svoj našoj deci
- 3 ... vama i nama lično

1. Zašto je to važno za ekonomiju Srbije?

IT sektor i ekonomija

IT je jedna od retkih mogućnosti za brz ekonomski napredak Srbije.

IT sektor je već postao jedan od ključnih nosilaca razvoja Srbije

- Izvoz softvera: od 200 mil € u 2011, do 740 mil € u 2016.
Izvoz softvera premašuje izvoz žitarica, malina kao i pogonskih mašina i uređaja
- Trgovinski suficit u IT uslugama 407 mil € u 2016
- Prosečna godišnja stopa rasta: 25%
- Udeo u BDP-u: 0,4% u 2010 na 1,7% BDP-a u 2016

1. Zašto je to važno za ekonomiju Srbije?

Prednosti IT sektora

- Potrebna ulaganja nisu velika
- Radi se o „čistoj“ i visoko-tehnološkoj industriji
- Obrt je brz
- Donosi napredak svim drugim granama

Razvoj IT sektora u Srbiji

- Trenutno postoji potreba za hiljadama novih programera
- Plate u IT sektoru višestruko su veće nego prosečne
- Sve veće interesovanje za studije informatike (na MatF-u: više od četiri kandidata za jedno mesto)

2. Zašto je to važno našim đacima i svoj našoj deci?

Našim đacima i svoj našoj deci programiranje je važno ...

- ... jer će u mnogim sredinama, i gradskim i seoskim, biti otkriveno puno dece koja su talentovana za programiranje
- ... jer će nekima pomoći da nađu svoj put ka informatičkim poslovima
- ... jer će svoj deci pomoći u razvoju rasuđivanja:
programiranje nas uči da razmišljamo precizno i analitički

2. Zašto je to važno našim đacima i svoj našoj deci?

Programiranje nas uči da razmišljamo precizno i analitički

- Već hiljadama godina se zarad unapređivanja sposobnosti rasuđivanja i preciznog izražavanja izučava matematika
- Programiranje može da bude još bolje za tu svrhu!
- Dete odmah može da sazna da li je tačno rešilo zadatak

Programiranje može i treba da se razmatra kao spoj:

- nauke
- tehnologije
- umetnosti
- igre

3. Zašto je sve ovo važno vama i nama lično?

Važno je zato što ...

- ... time možemo da doprinesemo ekonomiji Srbije
- ... ovim pomažemo i sebi u budućnosti (npr. kada IT donese više u budžet, biće veće i naše plate)
- ... možemo nekoj deci da pomognemo da izaberu ovaj put i rade kvalitetne i dobro plaćene poslove
- ... možemo svojoj deci da pomognemo da rasuđuju bolje

Dobra vest je ...

Nije mnogo teško

Postoji pripremljena podrška:

- Materijali koje ćemo koristiti brižljivo su pripremani
- Materijali nisu preambiciozni
- Umesto memorisanja činjenica akcenat je na rasuđivanju
- Postoji mali broj ideja koje deca treba da usvoje

Može da bude (veoma) zabavno

Deca vole računare, a programiranje je blisko igri

Ne očekujemo previše

Na samom početku (u 5. i 6. razredu)...

- Fond časova je mali (oko 16 časova posvećenih programiranju)
- Cilj je učenike **zainteresovati** za programiranje
- Sa nadarenim učenicima možemo postići više, ali ni ostalim učenicima ovi časovi ne bi trebalo da budu naporni

Zašto ovo nije nemoguće?

Nije nemoguće zato što ...

- ... postoji opšta svest da ova prilika ne sme da se propusti
- ... postoji opšta društvena saglasnost o potrebi za osnovnim informatičkim obrazovanjem
- ... ova ideja i potreba prevazilazi politiku, bilo koji univerzitet, bilo koju kompaniju

Zašto ovo nije nemoguće?

Nije nemoguće jer na koracima kao što je ovaj rade ljudi ...

- ... koji su vrhunski poznavaoici informatike
- ... koji su iskusni pedagozi
- ... koji su i roditelji i znaju šta deci može biti teško, a šta nezanimljivo
- ... koji žele bolju budućnost u Srbiji i za svoju decu

O autorskom timu

U izradi nastavnih materijala učestvovali su:

- Katarina Aleksić, profesor informatike u OŠ Branislav Nušić, Beograd
- Nevenka Spalević, profesorka u penziji Matematičke gimnazije u Beogradu
- Ana Kaplarević Mališić, docent PMF Kragujevac
- Filip Marić, profesor MatF Beograd
- Jovan Popović, Microsoft Development Center Serbia
- Milan Vugdelija, Microsoft Development Center Serbia
- Teo Šarkić, Microsoft Development Center Serbia
- Nebojša Vasiljević, Fondacija Petlja

O autorskom timu

Razvoj infrastrukture su vodili

- Srđan Božović, Microsoft Development Center Serbia
- Marko Baković, Računarski fakultet, Beograd

Autori obuke

Profesori Matematičkog fakulteta, Univerziteta u Beogradu

- Milena Vujošević Janičić
- Filip Marić
- Predrag Janičić

O trenerima

Obuka trenera

Miroslav Marić, Matematički fakultet, Beograd

Treneri, Matematički fakultet, Beograd

- Aleksandra Miladinović
- Maja Radulović
- Lazar Vojičić
- Duško Vešić
- Dušan Kovačević
- Katarina Andrejević
- Marijana Nisić
- Nemanja Jurić
- Vladimir Kuzmanović
- Dušan Džamić
- Sanja Pribanović
- Zorana Tošić
- Tanja Ivanović
- Olivera Tomić
- Dušan Šuljagić

Nastavnici — najvažniji deo tima

Vaša uloga je ključna za uspeh!

- Nastavnici su srce projekta
- Važno je da komuniciramo kako bi zajedno došli do najboljeg rešenja za sve

Zajednički cilj

**Kroz informatiku unaprediti rasuđivanje
učenika i zainteresovati ih za programiranje**

I mali pomak biće značajan za celu zemlju!

Kratak pregled obuke

Komunikacija tokom obuke

- Interakcija tokom časova veoma je važna
- Postavljajte pitanja!
- Na sva pitanja na koja ne budemo imali odgovore sada, dobićete odgovore tokom drugog dela obuke (kroz dve nedelje).

Kratak pregled obuke

Dva bloka od po 6 časova

- Prvih 6 časova danas
- Drugih 6 časova za dve nedelje

VAŽNO!!!

Tokom dve nedelje između obuka Vaš zadatak je da detaljno proučite i isprobate sve pripremljene nastavne materijale!

Kratak pregled obuke

Prvi deo obuke, 6 časova

Sadrži upoznavanje:

- osnovnih ciljeva i očekivanih rezultata uvođenja programiranja u osnovnu školu
- kratak pregled nastavnih materijala koji su učenicima i nastavnicima na raspolaganju
- tehničkih detalja koji su potrebni za korišćenje interaktivnih materijala
- osnovnih pojmova koje đaci treba da usvoje tokom nastave programiranja

Kratak pregled obuke

Drugi deo, 6 časova

- Vremenski je dve nedelje nakon prvog dela
- Ima za cilj da nakon odslušanog prvog dela i nakon samostalnog rada u pripremi nastave, da odgovori na sva pitanja koja su ostala otvorena
- Vaši komentari i utisci mogu da doprinesu doradi i unapređenju postojećih nastavnih materijala
- U meri u kojoj su nastavnici zainteresovani, biće prikazani i napredniji zadaci pogodni za sekcije i pripremu đaka za takmičenja

Kontakt informacije

Ovo je tek prvi korak

Podrška je organizovana i nakon obuke. **Niste sami!**

Problemi, pitanja, komentari, predlozi...




- <http://algora.petlja.org/c/zbornica>
- Forum „Algora”, kategorija „Zbornica”
- Međusobna komunikacija nastavnika, vidljiva samo nastavnicima i trenerima
- Aktivno učešće trenera, autora nastavnih materijala i tima koji radi na razvoju infrastrukture

Forum Algora

Algora Petlja BubbleBee Takprog

Algora za cilj ima da na jednom mestu okupi osnovce, srednjoškolce, takmičare, nastavnike i profesore, studente, početnike i iskusne programere, sve one kojima je potrebna pomoć u oblastima kojima se Petlja portal bavi i one koji tu pomoć mogu da pruže. Želja nam je da sve korisne informacije, materijale, jednom reči "znanje" koje je trenutno razbacano skupimo na jednom mestu, kako bi se što brže širilo. Algora je deo Petlje i deli iste vrednosti i ciljeve Petlje portala.

■ Zbornica ▶ Latest Top

Topic	Users	Replies	Activity
<p>🔍 O kategoriji "Zbornica"</p> <p>Ovo je kutak za nastavnike i profesore, za sve vaše diskusije, probleme, iskustva... Petlja ne može da ostvari svoj cilj bez vaše pomoći!</p>		0	May 5
Programiranje u osnovnoj školi		13	14d
Modernizacija uloge računarskog programiranja u obrazovanju		0	May 5

Uvođenje informatike i programiranja u osnovnu školu

Nastavni plan i program informatike za OŠ - 2017/18.

- 5. razred – obavezni predmet, planovi usvojeni 2017.
- 6. razred – izborni predmet, planovi usvojeni 2016.
- 7. i 8. razred – izborni predmeti, planovi usvojeni 2009. i 2010.

Oblasti koje se izučavaju

- Informaciono-komunikacione tehnologije
- Digitalna pismenost
- Računarstvo

Programiranje u OŠ - 2017/18.

U ovoj obuci fokusiramo se samo na oblast programiranja

Po trenutnim nastavnim planovima i programima, programiranje se može predavati u svim razredima

5. razred

- Deo redovnog programa obaveznog predmeta (oblast Računarstvo)
- Vizuelno-blokovsko programiranje (code.org, MIT Scratch, Alice, ...)

Programiranje u OŠ - 2017/18.

6. razred

- Deo redovnog programa izbornog predmeta (oblast Računarstvo)
- Tekstualno programiranje (Python, JavaScript, C#, C++, ...)

7. i 8. razred

- Izborni modul „Odabrana poglavlja programiranja” u sklopu izbornog predmeta
- Tekstualno programiranje (Python, Javascript, C#, C++, ...)

Programiranje u OŠ - perspektiva za naredne godine

Obavezni predmet u sva četiri razreda.

- 5. razred – vizuelno programiranje
- 6. razred – osnovne tehnike programiranja u tekstualnom jeziku
- 7. razred – primena naučenih osnovnih tehnika u nekom konkretnom domenu (grafika, igre, ...)
- 8. razred – naučno izračunavanje (korelacija sa matematikom, fizikom, ...)

Ciljevi i rezultati

Osnovni ciljevi

- Unaprediti kod učenika algoritamsko rasuđivanje koje im je važno u svakom poslu i u svakodnevnom životu
- Naučiti učenike osnovne pojmove programiranja
- Otvoriti im put da saznaju i nešto više i da se u budućnosti možda bave informatikom

Učenici imaju iskustvo u korišćenju tehnologije

Sinergija korisničkog i programerskog – sa malo elementarnog programiranja postajemo neuporedivo produktivniji u korišćenju računara za rešavanje problema iz ličnog i profesionalnog života

Ciljevi i rezultati

Osnovni ciljevi

- Izgrađivanje pozitivnih osobina ličnosti, kao što su sistematičnost, upornost, tačnost, preciznost, urednost, smisao za samostalni i grupni rad
- Razvijanje sposobnosti logičkog, kritičkog, analitičkog i apstraktnog razmišljanja
- Sticanje sposobnosti preciznog izražavanja u pismenom i usmenom obliku

Prolazne tačke

Na odličnom ste putu ako ...

- ... su časovi zabavni i većini učenika i Vama
- ... đaci rade postavljene zadatke i postavljaju pitanja
- ... imate 5-10% đaka koji su zainteresovani za takmičenje iz programiranja

Pregled

- 1 Uvod
- 2 Nastava zasnovana na ishodima**
- 3 Osnovni pojmovi i ideje
- 4 Raspoloživi nastavni materijali
- 5 Algoritmi i primeri
- 6 Između dva dela obuke
- 7 Zaključci

Funkcija osnovnog obrazovanja

Funkcija osnovnog obrazovanja

Funkcija osnovnog obrazovanja i vaspitanja jeste da: bazično opismeni učenike iz svih oblasti značajnih za život u savremenom svetu, da razvija funkcionalna znanja, umenja, motivaciju za učenje, stavove i vrednosti neophodne za formiranje nacionalnog i kulturnog identiteta, te bazične kulturne potrebe i navike, što osposobljava za dalje školovanje, celoživotno učenje i aktivan i konstruktivan život u savremenom društvu.

Funkcionalna znanja

Učenici treba da konstruišu funkcionalna znanja!

Nastavni proces treba da bude orijentisan na:

- **proces** — uvažavanje karakteristika i iskustava učenika sa kojima se radi, usaglašavanje metoda, oblika, sredstava. . .
- **rezultat** — usmerenost na učenje učenika, a ne samo na obradu sadržaja.

Učenje za...

Školu/ocenu

- Znanja se stiču u kratkom vremenskom periodu
- Kratkoročna su
- Sama su sebi svrha

Funkcionalno znanje

- Znanja se razvijaju kontinuirano
- Duže traju
- Vidljiva je njihova primena u svakodnevnom životu

Ishodi

Ishodi

- Ishodi su očekivani rezultati procesa učenja.
- Predstavljaju jasne i precizni iskaze o tome šta učenik treba da zna, da izvede, i vrednosno proceni po završetku procesa učenja.
- Definisani su na nivou primene znanja.

Ciljevi / Ishodi

Ciljevi

Cilj je formulisana svrha učenja i podučavanja. Njima se opisuje ono što se učenjem i podučavanjem namerava postići.

Ishodi

Ishodi su očekivani rezultati procesa učenja. Definisani su kao jasni i precizni iskazi o tome šta učenik treba da zna, da izvede, i vrednosno proceni po završetku procesa učenja.

Planiranje nastave

Planiranje nastave

- Svaka nastavna oblast/tema ima definisane ishode.
- Iz njih se izdvajaju ishodi za mesečni plan, dnevne pripreme.
- U pripremi za čas, nastavnik operacionalizuje ishode na tri nivoa (diferencira) i planira aktivnosti u odnosu na njih.

Ostvarivanje nastave

Ostvarivanje nastave

- Pažnja je usmerena ka učeniku – učenik je u centru.
- U odnosu na definisane ishode, nastavnik osmišljava i organizuje sopstvene i aktivnosti učenika.
- U odnosu na karakteristike učenika, nastavnik bira i prilagođava metode, sredstva i oblike rada kojima će ostvariti željeni ishod.

Fokus nije na realizaciji sadržaja, već na promišljanju sopstvene prakse.

Praćenje nastave

Praćenje nastave

Ishodi olakšavaju nastavniku da kontinuirano prati i vrednuje:

- učenička postignuća (jasni iskazi o tome šta učenici treba da postignu su sredstvo kojim se nastavnik rukovodi u praćenju i vrednovanju učeničkih postignuća);
- procese nastave (nivo ostvarenosti ishoda utiče na dalje planiranje nastave);
- sebe i svoj rad (rezultate svojih praćenja i promišljanja vraća u proces nastave i učenja, i tako stalno unapređuje učenje, nastavu i sebe, lično i profesionalno).

Učeniku daju jasnu sliku šta se od njega očekuje, omogućavaju samovrednovanje napredovanja u učenju i razvijaju samoregulaciju.

Pomeranje težišta

Pomeranje težišta

- Sa obrade sadržaja na planiranje aktivnosti kojima se obezbeđuje da na kraju časa, meseca, školske godine bude vidljivo i jednostavno proveriti šta učenik zna i može da primeni.
- Sa sticanja znanja za ocenu, na razvijanje kompetencija.

Rezime

Rezime

- **ishodi** – očekivani rezultati učenja
- **kompetencije** – aktivna primena naučenog
- **standardi** – mera ostvarenosti ishoda, razvijenosti kompetencija

Pregled

- 1 Uvod
- 2 Nastava zasnovana na ishodima
- 3 Osnovni pojmovi i ideje**
 - Osnovni principi učenja programiranja
 - Algoritamski način razmišljanja
- 4 Raspoloživi nastavni materijali
- 5 Algoritmi i primeri
- 6 Između dva dela obuke
- 7 Zaključci

Osnovni principi učenja programiranja

Programiranje predstaviti kroz primere iz svakodnevnog života

- Metodološki, **koncepte treba uvoditi na bazi asocijacija iz doživljenog iskustva**, a ne na bazi prethodnog teorijskog objašnjenja
- Kao što ne moramo na početku učenja govornog jezika znati šta su imenice i glagoli, tako i na početku učenja programiranja ne moramo znati šta su identifikatori i izrazi
- Izbegavati stručnu terminologiju koliko god je to moguće
- **Poželjno je da što više zadataka daje neku naznaku mogućnosti realne primene**

Primeri iz svakodnevnog života

Apstraktna formulacija (dosadna, nejasna motivacija)

Dat je niz od 5 brojeva. Odredi najveći element u tom nizu.

Konkretna formulacija (interesantnija, jasnija, bliža)

- Date su daljine skokova 5 takmičarki. Koja daljina je donela zlatnu medalju?



Primeri iz svakodnevnog života

Apstraktna formulacija (dosadna, nejasna motivacija)

Razmeniti vrednost dve promenljive.

Konkretna formulacija (interesantnija, jasnija, bliža)

- U dve čaše su dve vrste soka. Kako se može razmeniti njihov sadržaj?



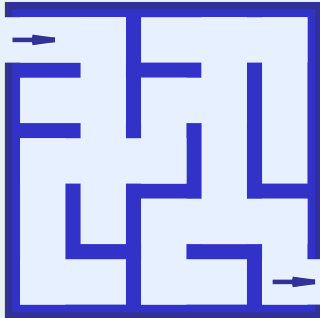
Osnovni principi učenja programiranja

Programiranje kroz zamisao, pokušaj i grešku

- Programiranje se uči kroz pokušaj i grešku — bez toga nema programiranja
- Ali ne sme se upustiti u pravljenje pokušaja pre razumevanja problema i, makar grubog, koncipiranja rešenja!

Programiranje kroz zamisao, pokušaj i grešku — primer

Lavirint



- Prvi (naivan) pokušaj: idemo samo pravo
- Drugi pokušaj: idemo samo pravo, a ako ne može dalje, onda desno
- Treći pokušaj: držimo se desnog zida

Osnovni principi učenja programiranja

Učenje na greškama

- Ako smo napravili grešku — razumimo je dobro, pre nego što nastavimo dalje
- Učenik mora da razume program i bez njegovog izvršavanja (pažljivom analizom napisanog koda)
- Svaki programer mnogo više vremena utroši na razumevanje i ispravljanje svojih grešaka nego na pisanje novog koda!

Algoritmi (procedure)

Šta je to algoritam (procedura)?

- Neformalno, algoritam opisuje korake čijim se sprovođenjem rešava neki problem
- Na primer, "Kakva je procedura za izdavanje lične karte?"

Algoritamski način razmišljanja

- Korisan je ne samo u programiranju, nego u svakom poslu i u svakodnevnom životu
- Unapređuje analitičnost, razlaganje problema na potprobleme, apstrahovanje (zanemarivanje nebitnih informacija), itd.
- **Ključan za uspešnost u bilo kom poslu**

Automatizacija i algoritmi

Sve više poslova će biti podložno automatizaciji

- Na primer, preduzetnik otvara onlajn prodavnicu
 - nema živog kontakta sa kupcima
 - potrebno unapred definisati poslovna pravila (kada treba ponuditi popust, kada treba prikazati reklamu, ...)
 - iako neće sam programirati onlajn sistem, preduzetnik mora da razume algoritamski aspekt i način funkcionisanja automatizovanih procesa

Automatizacija i algoritmi

Kada god postavljamo pravila po kojima će se neki proces automatski odvijati, mi smo uključeni u oblikovanje algoritma

Algoritamski način razmišljanja

Kako učiti ...

- Fizičke sposobnosti vežbaju se kroz sport, bez obzira na to što većina đaka neće postati profesionalni sportisti
- Razumevanje zakona fizike se produbljuje kroz izradu računskih zadataka, bez obzira na to što većina đaka takve proračune neće izvoditi u svojoj profesiji

Kako učiti algoritamski način razmišljanja?

Algoritamski način razmišljanja se najbolje može uvežbati kroz programiranje, bez obzira na to što većina đaka neće biti profesionalni programeri

Zašto su algoritmi važni?

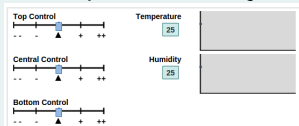
Naši đaci ...

- ... često znaju gomile činjenica i formula, ali nisu u stanju da ih iskoriste u rešavanju jednostavnih problema iz svakodnevnog života
- ... često znaju procentni račun, ali ne znaju da ga primene za rešavanje elementarnog pitanja u vezi sa štednjom ili kreditom
- ... obično vrlo loše urade PISA testove

Primer

PISA zadatak

- Imaš klima uređaj bez uputstva za upotrebu. Na daljinskom upravljaču postoje tri kontrole sa označenim početnim stanjima. U simulatoru u veb aplikaciji, moguće im je menjati vrednosti i pratiti vrednosti temperature i vlažnosti. Odredi koja kontrola utiče na temperaturu a koja na vlažnost i kako.



- Navedeni zadatak rešilo je oko 80% dece u Južnoj Koreji, Japanu i Singapuru, a samo 45% u Srbiji!

Primer

Analitičnost u rasuđivanju

Nedavna anketa na jednom našem velikom portalu:

Novak Đoković će na ovom turniru:

- 1 izgubiti u prvoj nedelji
- 2 izgubiti u polufinalu od Federera
- 3 izgubiti u finalu
- 4 osvojiti titulu



Šta da izaberu oni koji misle da će Novak Đoković izgubiti u polufinalu od nekog drugog?

Primer

Analitičnost u rasuđivanju

Druga anketa:

Koliko ste platili Vaš televizor?

- 1 Do 20.000 din
- 2 Od 20.000 do 30.000 din
- 3 Od 30.000 do 40.000 din
- 4 Od 40.000 do 50.000 din
- 5 Više od 60.000 din

Šta da izaberu oni koji su televizor platili 52.000 din?

Algoritmi u svakodnevnom životu

Algoritmi (procedure) su svuda oko nas

Deci je važno navesti primere algoritama u svakodnevnom životu:

- Kuhinjski recepti
- Procedura registrovanja na gmail-u
- Procedura koju putnik treba da sledi pre ukrcavanja na avion (rezervacija karte, kupovina, čekiranje, predaja prtljaga, ...)
- Protokol praćenja trudnoće
- Procedura legalizacije objekata
- Procedura za vađenje pasoša
- Procedura za upis u srednju školu

Algoritmi u matematici

Deca su se srela i sa mnogim matematičkim algoritmima

- Algoritam sabiranja brojeva potpisivanjem
- Algoritam konstrukcije simetrane duži
- Algoritam za rešavanje linearne jednačine
- Algoritam sabiranja razlomaka
- Algoritam za računanje najvećeg zajedničkog delioca
- Algoritam za računanje najmanjeg zajedničkog sadržaoća

Pregled

- 1 Uvod
- 2 Nastava zasnovana na ishodima
- 3 Osnovni pojmovi i ideje
- 4 Raspoloživi nastavni materijali**
 - Programski jezici i okruženja
 - O materijalima
 - Sistem za praćenje rada učenika
- 5 Algoritmi i primeri
- 6 Između dva dela obuke
- 7 Zaključci

Mogući izbor za 5. razred

Naš izbor je Scratch

- code.org
- **Scratch**
- Alice
- ...

Mogući izbor za 5. razred - code.org

code.org

- ... početni zadaci u obliku igrica (npr. sat programiranja)
- ... unapred pripremljeni zadaci daju premalu slobodu u izražavanju i premalu kreativnost
- ... moguće je na samom početku sa decom preći neki uvodni kurs (na primer, Sat programiranja)

Mogući izbor za 5. razred - code.org

Scratch ...

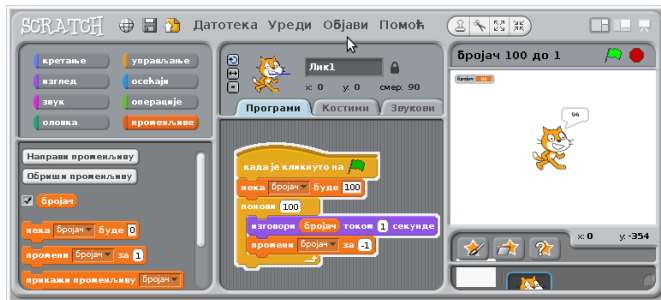
- ... je moderan vizuelni pristup programiranju, posebno pogodan za usvajanje početnih i osnovnih koncepata kod dece.
- ... smatra se najprikladnijim za prvi susret sa programiranjem.

Mogući izbor za 5. razred - Alice

Alice ...

- ... sličan koncept Scratch-u
- ... 3d grafika
- ... ozbiljan izlet u objektno-orijentisanu paradigmu

Scratch



Scratch može da se koristi na dva načina

Instaliran na računaru

- Instalacija se skida sa adrese:

`https://scratch.mit.edu/scratch2download/`

- Preduslov je da je na računaru instaliran **Adobe Air**, koji se može preuzeti sa iste adrese

Onlajn platforma

Na adresi:

`https://scratch.mit.edu/`

Uvod
Nastava zasnovana na ishodima
Osnovni pojmovi i ideje
Raspoloživi nastavni materijali
Algoritmi i primeri
Između dva dela obuke
Zaključci

Programski jezici i okruženja
O materijalima
Sistem za praćenje rada učenika

Scratch

Detaljno uputstvo

<http://petlja.org/s/p5okr>

Petlja BubbleBee Takprog Algora PSEML BubbleCup X Osnovna škola (demo) Prijava se

BubbleBee / Interaktivni priručnik za Scratch za peti razred / SCRATCH RADNO OKRUŽEŃE

SCRATCH RADNO OKRUŽEŃE

U prethodnoj lekciji razgovarali smo o algoritimima i računarskim programima. Predstavili smo ti vizuelni programski jezik Scratch. To je jezik kojim tvoj računaru saopštavati ono što želiš da uradi.

Da bi tvoja programerska avantura mogla da počne, neophodno je da se upoznaš sa radnim okruženjem Scratch-a.

Scratch možeš da koristiš na dva načina:

1. možeš da ga instaliraš na svoj računar (potrebno je da, sa internet adrese <https://scratch.mit.edu/scratch2download/>, preuzmeš instalacionu datoteku (fajl/ove) za instalaciju programa Adobe AIR i programa Scratch Offline Editor i instaliraš ih).

Scratch 2.0 Offline Editor

You can install the Scratch 2.0 editor to work on projects without an internet connection. This version will work on Mac, Windows, and some versions of Linux (64 bit).

Note for Mac Users: the latest version of Scratch 2.0 Offline requires Adobe AIR 20. To upgrade to Adobe AIR 20 manually, go [here](#).

Adobe AIR Scratch Offline Editor Support Materials

Mogući izbor za 6. razred

Naš izbor je Python

- Python
- JavaScript
- C#, C++, Java
- Pascal, C

Mogući izbor za 6. razred

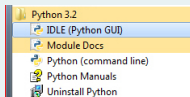
Python ...

- ... je moderan i živ programski jezik, koristi se i u realnim primenama.
- ... dinamički tipiziran i time znatno jednostavniji za korišćenje od jezika poput C#, C++, Java, Pascal, C, ...
- ... smatra se najprikladnijim za prvi susret sa programskim jezicima.
- ... statistika iz 2014 pokazuje da se Python koristio u prvim kursevima na 8 od 10 najboljih američkih univerziteta
- ... veliki broj biblioteka za razne interesantne primene (npr. ugrađena kornjača grafika, SciPy, NumPy, ...)

Python može da se koristi na nekoliko načina

Instaliran na računaru

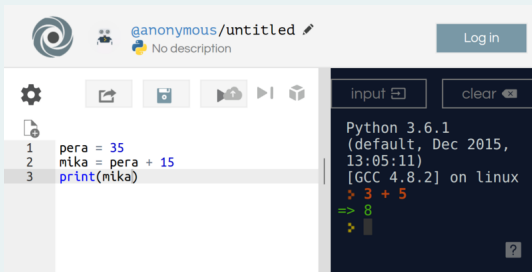
- instalacija se preuzima sa adrese:
`http://www.python.org/downloads/`
- **VAŽNO: Instalirati verziju 3.x**, a ne verziju broj 2.x
- preporučujemo razvojno okruženje IDLE (već uključeno u osnovnu instalaciju)
- Pokretanje na sistemu Windows: Start Menu -> idle



Python može da se koristi na nekoliko načina

Onlajn okruženja

- potrebna stalna veza sa internetom
- <https://repl.it> - podržava razne jezike (Python 3, Python 3 with turtle, ali i C#, C++, ...)



The screenshot shows a web-based Python REPL interface. At the top, there is a logo, a user profile for '@anonymous/untitled', and a 'Log in' button. Below the header is a toolbar with icons for settings, file operations, and execution. The main area is split into two panes: a code editor on the left and a terminal on the right. The code editor contains three lines of Python code: `1 pera = 35`, `2 mika = pera + 15`, and `3 print(mika)`. The terminal on the right shows the output of the code: `Python 3.6.1`, `(default, Dec 2015,`, `13:05:11)`, `[GCC 4.8.2] on linux`, `> 3 + 5`, `=> 8`, and a cursor on the next line.

Python može da se koristi na nekoliko načina

Interaktivni priručnik na `petlja.org`

- Na adresi:

`http://petlja.org/s/p6`

- Programi se pišu direktno u sklopu veb-strane priručnika
- Idealno za početne korake u učenju
- Nedovoljno za kasniji ozbiljniji rad

Izgled primera sa online platforme

The screenshot shows a web interface for an online learning platform. At the top, there is a dark navigation bar with the user's name 'Petlja' and a list of course topics: 'Bubblevee', 'Takprog', 'Algora', 'PSI-ML', 'BubbleCup X', 'Osnovna škola (demo)', and a 'Prijavi se' button. On the left side, there is a vertical menu with various topics related to strings and algorithms, such as 'Ниске (stringovi)', 'Надовезивање ниски', 'Учитавање ниски', 'Дужина ниске, издавање делова ниске', 'Претрага ниске', 'А шта је са цифрама у ниски?', 'Задачи са нискима', 'Структуре података', 'Гранање', 'Понављање', 'Корњачка графика', 'Карел робот', 'Имена вредности променљивих', and 'Основни алгоритми'. The main content area is titled 'Надовезивање ниски' (Concatenating strings). It contains a text block explaining that certain characters in a string need to be escaped with backslashes, using the example 'Rekao je: \Zdravo, svima!'. Below this is a code editor with a single line of code: `print("Rekao je: \Zdravo, svima!.")`. There are two buttons: 'Покрени програм' (Run program) and 'Корак по корак' (Step by step). Below the code editor, there is a paragraph explaining string concatenation and multiplication, with examples like 'abraka' + 'dabra' resulting in 'abradabra' and 'ba' * 2 resulting in 'baba'. At the bottom, there is a question 'Q-7: Који од наредних израза крију у себи исправно записан назив воћа?' (Which of the following expressions contain a correctly written fruit name?) with multiple-choice options (A) 'ba' + 'na' * 2, (B) 'a' + 'na' * 2 + 's', and (C) 'a' + 'na' * 2 + 's'.

Izgled primera sa online platforme

Petlja BubbleBee Takprog Algora Osnovna škola PSI:ML BubbleCup X - Filipmaric -

Уроцине функције

- Дефинисање функција
- Ниске (stringови)
- Структуре података
- Гранање
- Измена вредности променљивих
- Понављање
- Корњача графика
- Карел робот ▾
 - Линијски програми
 - Понављање
 - Гранање
 - Разни лавиринти
 - Основни алгоритми

Прикажимо употребу ових наредби на неколико једноставних програма.

Иди на поље (3, 3)

Напиши програм који ће робота довести на поље (3, 3).

Покрени програм Врати на почетак

```
1 from karel import *
2 napred() # idi napred
3 napred() # idi napred
4 levo() # okreni se nalevo
5 napred() # idi napred
6 napred() # idi napred
```

Karel 0 (karel_0.png)

Dve grupe povezanih nastavnih materijala

Materijali koje je izradila fondacija „Petlja”

- <http://petlja.org/osnovnaskola>
- Interaktivni priručnici, praktikumi, zbirke zadataka

Materijali nastali u okviru projekta „Programiraj” MTT

- <http://petlja.org/programiraj>
- Video materijali, pripreme za čas

Raspoloživost

- Svi materijali su javno dostupni i besplatni!

Prikaz raspoloživih materijala

Navedeni materijali mogu se istovremeno koristiti u nastavi

- Materijali se međusobno dopunjuju, ...
- ... ali postoji i značajno preklapanje između njih (npr. u priručnike su uključene video-lekcije, dok se u video lekcijama prikazuju neke lekcije iz priručnika)
- Upoznaćemo se sa svim materijalima
- Svi Vaši komentari na temu materijala su dobrodošli

Raspoloživi materijali — fondacija Petlja

<http://petlja.org/osnovnaskola>



Програмирање у Скречу,
приручник за пети разред

ОТВОРИ



Програмирање у Скречу,
практикум за пети разред
(бета)

ОТВОРИ



Програмирање у Пајтону,
приручник за шести
разред

ОТВОРИ



Програмирање у Пајтону,
збирка алгоритамских
задача за шести разред

ОТВОРИ



Програмирање у Пајтону,
збирка кратких питања за
шести разред (бета)

ОТВОРИ

Raspoloživi materijali — fondacija Petlja

5. razred (Scratch)

- Programiranje u Skreču, priručnik za peti razred
- Programiranje u Skreču, praktikum za peti razred

6. razred (Python)

- Programiranje u Pajtonu, priručnik za šesti razred
- Programiranje u Pajtonu, zbirka algoritamskih zadataka za šesti razred
- Programiranje u Pajtonu, zbirka kratkih pitanja za šesti razred


Programiranje u Skreču, priručnik za peti razred

Priručnik

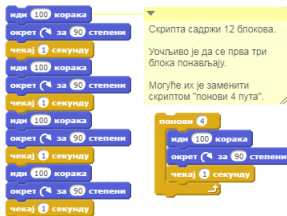
- Iznosi osnovne pojmove na veoma sažet način
- Pokriva samo osnovni nivo — ono što sva deca mogu da savladaju
- Integrisane video-lekcije nastale u sklopu projekta „Programiraj”

Programiranje u Skreću, priručnik za peti razred

Posmatrajući dužu skriptu, nije teško zaključiti da nizaње идентичних група блокова чини програм дужим, тежим за разумевање и надограђивање. Ако би требало променити угао за који лик скреће у десну страну, промена би морала да буде урађена на 4 места.

Блок  из категорије **Управљање** очигледно помаже да се овакви проблеми превазиђу.

Програмери (људи који пишу програме) сматрају да је корисно коментарисати скрипте и објаснити шта одређени блокови раде. Коментарисањем олакшаваш другим програмерима да разумеју програме које ствараш, као и да их надограђују.



Programiranje u Skreču, praktikum za peti razred

Praktikum

- Veoma bogat i detaljno razrađen materijal
- Pokriva sve nivoe znanja (osnovni, srednji, napredni)
- Sadrži kolekciju kratkih pitalica za proveru savladanog
- Sadrži kolekciju zadataka za vežbu
- Sadrži ideje projekata za samostalni rad

Programiranje u Skreču, praktikum za peti razred

Kratke pitalice za proveru savladanog

Šta se postiže izvršavanjem sledećeg beskonačnog ciklusa, ako su prethodno izvršene naredbe **иди на x: 0 y: 0** ; **усмери се ка 90** и **нека начин окретања буде лево-десно** ?



- (A) лик се okreће лево-десно
- (B) лик иде напред-назад
- (C) лик иде горе-доле
- (D) лик се врти у круг

Провери

Упореди

Programiranje u Skreču, praktikum za peti razred

Zadaci za vežbu

Отворена огрлица. Креирај пројекат који приказује низ кругова различите боје и величине почев од левог краја позорнице.



Прикажи упутство

Концентрични кругови. Креирај пројекат који приказује низ концентричних кругова различите боје у центру позорнице.



Прикажи упутство

Programiranje u Skreču, praktikum za peti razred

Projektni zadatak

Petlja BubbleBee Takprog Algora BubbleCup X Osnovna škola (demo)

Petlja / Osnovna škola / Interaktivni praktikum za Scratch za peti razred

Пример 3 - Пројекат Прати_ме по трагу 1

У претходним поглављима смо научили како можемо управљати кретањем лика помоћу дирки са тастатуре. Сада ћемо тим скриптама за кретање додати могућност да лик оставља траг за собом.

Лик који ће остављати траг биће бубамара *Lađubid 1* из библиотеке ликова. Бубамари ћемо придружити скрипте за вођење стрелицама као у вежби 3 поглавља *Координате позорнице*. Уз скрипте које описују понашање бубамаре када се притисне нека од дирки са стрелицама, направити ћемо и скрипту која се активира кликом на зелену заставицу, а која поставља боју и дебелину оловке и спушта оловку.

На самом почетку обрисаћемо све трагове који су евентуално остали од претходних извршавања пројекта.

Ова скрипта и изглед екрана ако се бубамара после покретања пројекта води прво десно, па доле, лево и горе приказани су на следећој слици.



Programiranje u Pajtonu, priručnik za šesti razred

Priručnik

- Postupno uvodi sve koncepte programskog jezika
- Sadrži mali broj kratkih pitanja kroz koje se proverava osnovno razumevanje tih koncepata
- Sadrži zadatke koji ilustruju primenu tih koncepata na rešavanje jednostavnih problema iz realnih situacija
- Učenici dopunjuju i ispravljaju već zadati programski kôd
- Izvršavanje programa unutar veb-stranice
- Korak-po-korak izvršavanje
- Uključena i grafika (kornjača, robot Karel)

Programiranje u Pajtonu, priručnik za šesti razred

Boje u priručniku

- Zelena boja — osnovni nivo (za svu decu na časovima, dovoljan za peticu)
- Žuta boja — teži nivo (pogodan za sekcije ili dodatnu nastavu)
- Crvena boja — napredni nivo (pogodan za pripremu za takmičenje)

Programiranje u Pajtonu, priručnik za šesti razred

Uvođenje konceptata

Сабирање, одузимање, множење

О аритметичким операцијама и њиховој примени си учио/учила још у нижим разредима.

- Основна аритметичка операција је **сабирање**. Збир бројева 3 и 5 се у математици представља као $3 + 5$. У програмском језику Python користи се готово идентичан запис $3 + 5$.
- Поред сабирања можемо разматрати **одузимање**. Разлика бројева 8 и 2 се у математици представља као $8 - 2$. У програмском језику Python користи се готово идентичан запис $8 - 2$.
- Још једна од основних операција је и **множење**. Производ бројева 4 и 6 се у математици представља као $4 \cdot 6$. У програмском језику Python множење се означава помоћу оператора `*` и производ бројева 4 и 6 се записује као $4 * 6$.

Програмски језик Python, наравно, уме и да дели, да израчунава остатак при дељењу и цео део количника и много штошта друго. О овим операцијама ћемо говорити на неком од наредних часова.

Programiranje u Pajtonu, priručnik za šesti razred

Kratke pitalice

Koja je vrednost izraza $(5 + 5) * 5$?

- (A) 15
- (B) 30
- (C) 50
- (D) 125

Провери

Упореди

Koji od narednih programa izračunava i ispisuje proizvod razlike brojeva 184 i 72 i razlike brojeva 273 i 194.

- (A) `print((184 + 72) * (273 - 194))`
- (B) `print((184 - 72) * (273 - 194))`
- (C) `(184 - 72) * (273 - 194)`
- (D) `print(184 - 72 * 273 - 194)`

Провери

Упореди

Programiranje u Pajtonu, priručnik za šesti razred

Aktivni kod

Покрени програм

Корак по корак

```
1 jmbg = "1702992850011"  
2 dan = jmbg[0:2] # izdvajamo tekst sa pozicija 0 i 1  
3 mesec = "" # izdvajamo tekst sa pozicija 2 i 3 - popravi ovaj red  
4 print("Rodjen si " + dan + "." + mesec + ".")  
5
```

ActiveCode: 1 (jmbg)

Programiranje u Pajtonu, priručnik za šesti razred

Izvršavanje „korak-po-korak”

Python 3.6

```

1 broj = 5 # neka j
2 while broj <= 100: # dok je
3     print(broj) # isp
4     broj = broj + 5 # uve

```

Print output (drag lower right corner to resize)

```

5
10

```

Frames Objects

Global frame

broj 15

← line that has just executed
→ next line to execute

< Back Step 9 of 62 Forward >

Visualized using [Python Tutor](#) by [Phillip Guo \(@pgbovine\)](#)

Programiranje u Pajtonu, zbirka algoritamskih zadataka za šesti razred

Zbirka algoritamskih zadataka

- Uvod ka ozbiljnijem programiranju
- Zadaci takmičarskog tipa
 - konzolne aplikacije
 - automatsko testiranje (pomoću test-primera)
- Ipak, težina zadataka je takva da veliki broj učenika može uspešno da ih savlada (ne moraju svi učenici)
- Nastavnicima je dozvoljen pristup rešenjima

Programiranje u Pajtonu, zbirka algoritamskih zadataka za šesti razred

Proja

vreme	memorija	ulaz	izlaz
1 s	64 Mb	standardni ulaz	standardni izlaz

Kukuruzni hleb, proja, se pravi mešanjem kukuruznog i pšeničnog brašna i vode (naravno, po želji možete dodati jaja, so, sir, kajmak i slično, ali mi ćemo razmatrati samo neophodne sastojke). Neophodno je da upotrebljena količina kukuruznog brašna bude dvostruko veća od upotrebljene količine pšeničnog brašna. Količina vode koja se dodaje u mešavinu brašna je dvostruko manja od ukupne količine brašna. Ako je data količina kukuruznog i količina pšeničnog brašna, kojom raspoložemo odrediti koju količinu vode treba dodati ako želimo da umesimo najveću moguću proju od namirnica kojima raspoložemo.

Ulaz

Info

Pokušalo: 37
Rešilo: 64%
[Rešenje](#)

Poslata rešenja

Nema poslanih rešenja

Problem rešili

anastasijapuskas,
Dusan53, dejana,
vladacar, boris.z, sr.zivk,
Teodor, Zlatko
Golubovic,
MarkovBrat123, Dusan
Zivanic07

Programiranje u Pajtonu, zbirka pitanja za šesti razred

Zbirka kratkih pitanja

- Bogata kolekcija kratkih pitanja za proveru razumevanja osnovnih koncepata
- Pitanja pogodna za kontrolne zadatke
- Moguće automatski generisati PDF sa odabranim pitanjima

Programiranje u Pajtonu, zbirka pitanja za šesti razred

Питање 9.*

Koje ће од наведених команди исписати тачан одговор на дато питање?

Ako je Marko platio 280 dinara 400 grama šunke, koliko košta kilogram šunke?

Изабери тачан одговор:

- (A) `print(280 * 0.4)`
- (B) `print(280 / 0.4)`
- (C) `print(280 / 400)`
- (D) `print(280 / 400 * 1000)`

Провери

Упореди

Uvod
Nastava zasnovana na ishodima
Osnovni pojmovi i ideje
Raspoloživi nastavni materijali
Algoritmi i primeri
Između dva dela obuke
Zaključci

Programski jezici i okruženja
O materijalima
Sistem za praćenje rada učenika

Raspoloživi materijali — projekat „Programiraj”

<http://petlja.org/programiraj>

Petlja BubbleBee Takprog Algora PSI.ML BubbleCup X Osnovna škola (demo) Prijavi se

Na ovoj strani možete preuzeti tekstualne i video materijale koji su rezultat projekata „Programiranje za osnovec” realizovanog od strane *Društva za informatiku Srbije* i projekta „Vizuелно programiranje за основец” који је реализовала *Мудла мрежа Србије*. Оба пројекта су финансирана средствима из Буџета Републике Србије преко *Министарства трговине, туризма и телекомуникација*, а уз подршку *Министарства просвете, науке и технолошког развоја*.

Ауторски тимови два наведена пројекта су блиско сарађивали са ауторским тимом Петље и узајамно користили и реферисали материјале, при чему је израда Интерактивног приручника за VI разред од стране ауторског тима Петље претходила изради материјала у пројекту „Programiranje за osnovec”, док су за V разред прво израђени материјали у оквиру пројекта „Vizuелно programiranje за osnovec”.

Визуелно програмирање за основец Програмирање за основец



V разред

- лекција - Алгоритамски начин размишљања
 - Видео
 - Приручник
 - Припрема
- лекција - Scratch радно окружење
 - Видео
 - Приручник

VI разред

- лекција-Python - инсталација
 - Видео
- лекција-Python - аритметичке операције
 - Видео
 - Приручник
 - Припрема
- лекција-Python - променљиве (именовање резултата)

Raspoloživi materijali — projekat „Programiraj”

<http://petlja.org/programiraj>

- Video materijal
- Priručnik (umesto knjige)
- Priprema (plan časa)

The screenshot shows the website interface for 'Petlja'. The navigation bar includes links for 'BubbleBee', 'Takprog', 'Algora', 'PSI:ML', 'BubbleCup X', 'Osnovna škola (demo)', and a 'Prijava se' button. The main content is divided into two columns: 'V razred' and 'VI razred'. Under 'V razred', there are two lessons: '1. lekcija - Algoritamski način razmišljanja' and '2. lekcija - Scratch radno okruženje'. Each lesson has a list of resources: 'Video', 'Priručnik', and 'Priprema'. Under 'VI razred', there are three lessons: '1. lekcija-Python - instalacija', '2. lekcija-Python - aritmetičke operacije', and '3. lekcija-Python - promenljive (imenovanje rezultata)'. Each lesson also has a list of resources: 'Video', 'Priručnik', and 'Priprema'.

Projekat „Programiraj” — priručnik

Програмирај!



АЛГОРИТАМСКИ НАЧИН РАЗМИШЉАЊА

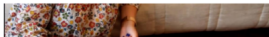
У свету у коме живимо програмирање треба посматрати као продужетак писана. Програмирајући, можемо да ствармо интерактивне приче, симулације и анимације. Алгоритам је низ корака којих се морамо придржавати како би доšli до решења неког задатка.

Постоје три типа алгоритама:

- 1) **Линијски алгоритам** - извршава се корак по корак, од првог до последњег, и сваки од корака се извршава тачно једанпут;
- 2) **Циклични алгоритми** - у њима примењујемо понављања;
- 3) **Условни (разгранати) алгоритми** - то су решења задатака који зависе од неког услова. У условним алгоритмима неке наредбе ће се извршавати, а неке неће, у зависности од испуњености услова.

Погледај видео лекцију Лекција 1 –
Алгоритамски начин размишљања, на адреси
<https://youtu.be/1CTAG6uYbWc?list=PLvXYSFHMAsA-KcZDmMTh1LKWg2nc0G0>

Шта мислиш, који тип алгоритама користи Кеџија да би избегла шетане Болета? Покушај да објасниш свој став.



Projekat „Programiraj” — plan časa

Програмирај!



ПЛАН ЧАСА	
Рачунарство и информатика за пети разред	
Наставна тема:	Рачунарство
Редни број часа:	1.
Наставна јединица	Алгоритамски начин размишљања
Тип часа:	обрада
Циљ часа:	Упознавање ученика са појмом алгоритма помоћу једноставних примера из свакодневног живота.
Исходи часа:	<ul style="list-style-type: none">Ученик примењује концепт разлагања проблема на мање делове, које затим обједињује у целовито решењеУченик објашњава појам алгоритмаУченик препознаје типове алгоритма
Област рада:	фронтални, индивидуални, рад у пару
Наставне методе:	илустративно-демонстративна, метода разговора, метода усменог излагања
Место реализације:	учионица, рачунарски кабинет (дигитална учионица)
Кратак опис часа	
Кроз навођење једноставних примера из свакодневног живота ученици ће се упознати са појмом алгоритма. Након тога, уз помоћ видео – лекције <i>Алгоритамски начин размишљања</i> , практичног рада на „задацима” (примерима из свакодневног живота које им је друг поставио) и презентовања решења, ученици ће уочити примену алгоритма у свакодневном животу и схватити значај примене принципа разлагања проблема на мање делове.	

Sistem za praćenje rada učenika

Nastavnik može da prati napredak svoji đaka

- Svi nastavni materijali su interaktivni
- Učenik dok prolazi kroz materijal ima aktivnu ulogu (odgovara na pitanja, ispravlja greške, dopunjava programe, ...)
- **Sistem za praćenje rada učenika** pomaže u prikupljanju podataka o učeničkoj aktivnosti i napretku

Nalozi na petlji

Nalozi

- Jedinstven nalog za prijavljivanje na sve servise na `petlja.org` (osnovnaskola, bubblebee, takprog, algora)
- Nalog pripada jednoj osobi (nemojte praviti školske naloge)
- Registracioni mejl (može da završi u neželjenoj pošti)
- Probleme sa registracijom prijavljivati na `http://algora.petlja.org/t/razumevanje-petlja-nalog`

Profesorski nalozi

Profesorski nalozi

- Profesori imaju posebna prava na sajtu
- Imaju pristup svim rešenjima zadataka u zbirci
- Mogu da kreiraju grupe učenika i prate njihov rad
- <http://petlja.org/cpanel/Groups>
- U delu “Moje grupe” treba da se nalazi grupa “Profesori”
- Trener može da vas doda u tu grupu

Učenički nalozi

Otvaranje naloga i kreiranje grupa

- Svaki korisnik petlje može da pravi ad-hoc grupe drugih korisnika čiji rad želi da prati (neophodno je da su oni prethodno registrovani na petlji i da pristaju da budu članovi grupe)
- Profesor može jednostavno da kreira posebne učeničke grupe (nije potrebno da učenici budu registrovani, niti da svako od njih da posebnu saglasnost)

Učenički nalozi

Učenički nalozi

- Preporučujemo da učenici u nastavi koriste naloge koje ste kreirali Vi, a ne njihove lične naloge (ako već poseduju nalog na Petlji).
- Učenički nalozi imaju korisničko ime oblika OS00001 i Petlja portal ih ni na koji način ne povezuje sa stvarnim korisnicima.
- Na Vama je da vodite računa koji učenik koristi koji nalog kako biste ispravno pratili njihov rad.
- Sa učeničkim nalogima se može pristupiti svim servisima Petlje, osim Algori.

Uvod
Nastava zasnovana na ishodima
Osnovni pojmovi i ideje
Raspoloživi nastavni materijali
Algoritmi i primeri
Između dva dela obuke
Zaključci

Programski jezici i okruženja
O materijalima
Sistem za praćenje rada učenika

Otvaranje naloga i kreiranje grupa

Kreiranje učeničkih grupa

● <http://petlja.org/cpanel/Accounts>

The screenshot shows the 'Accounts' management interface. On the left is a sidebar with navigation links: 'Kontrolna tabla', 'Profil', 'Grupe', 'Naloz', and 'Statistika'. The main content area is titled 'Naloz' and contains a form for creating a new account. The form has three fields: 'Broj naloga' (set to 25), 'Grupa' (set to 'Nova grupa'), and 'Naziv'. Below the form are 'Poniši' and 'Kreiraj' buttons. Underneath the form is a section 'Moj nalozi' with a filter dropdown set to 'Sve grupe'. Below this is a table of existing accounts.

Korisničko ime	Šifra	Grupa
OS_94981	[Masked]	Primer grupe
OS_94982	[Masked]	Primer grupe
OS_94983	[Masked]	Primer grupe

Otvaranje naloga i kreiranje grupa

Kreiranje učeničkih grupa

- Nalozi se kreiraju odabirom broja naloga, selekcijom “Nova grupa” u polju “Grupa” i upisivanjem naziva grupe (na primer naziv odeljenja).
- Ne morate uvek kreirati novu grupu, već naloge možete dodati i u neku postojeću odabirom te grupe u polju “Grupa” (polje “Naziv” ostavite prazno).
- Na dnu stranice u delu “Moji nalozi” možete videti korisničko ime i šifru koje je potrebno podeliti učenicima.
- Učenik se prijavljuje na Petlju sa ovim nalogom isto kao sa standardnim nalogom, upisivanje korisničkog imena i šifre.

Upravljanje nalogima

Upravljanje nalogima

- Za svaki nalog možete generisati novu šifru
- Promeniti grupu kojoj nalog pripada
- Obrisati nalog

Moguće situacije

Moguće situacije

- Novi učenik – samo napravite još naloga i dodajte ih u postojeću grupu
- Učenik je promenio odeljenje – ako ste mu Vi i dalje nastavnik, samo nalog možete prebaciti u drugu grupu, a ako niste, onda je komplikovanje, ali je moguće (opisano detaljnije u uputstvu)
- Promena nastavnika – stari nastavnik može prebaciti vlasništvo nad grupu (klikom na “Detaljnije” za odgovarajuću grupu)

Statistike rada učenika

Nastavnik ima uvid u statistike rada učenika

The screenshot displays a web interface for a teacher's dashboard. On the left is a sidebar menu with options: Profil, Grupe, Naloz, and Statistika (highlighted). The main area is titled 'Statistika' and shows a list of student groups with their performance metrics. The first group, 'zmajek', has a score of 12/30 on quizzes and 0/0 on tasks and tests. The other groups listed are 'Delta003' and four groups with IDs starting with 'OS_94981' through 'OS_94984', all showing 0/30 on quizzes and 0/0 on tasks and tests. The interface includes filters for 'Grupa' (set to 'Sve grupe'), 'Kurs' (set to 'Programiranje u Pajtonu, pripremnici za šest:'), and 'Lekcija' (set to 'Osnove aritmetičke operacije i primena').

Grupa	pitalice	zadaci	test
zmajek	12 / 30	0 / 0	0 / 0
Delta003	1 / 30	0 / 0	0 / 0
OS_94981	0 / 30	0 / 0	0 / 0
OS_94982	0 / 30	0 / 0	0 / 0
OS_94983	0 / 30	0 / 0	0 / 0
OS_94984	0 / 30	0 / 0	0 / 0

Pregled

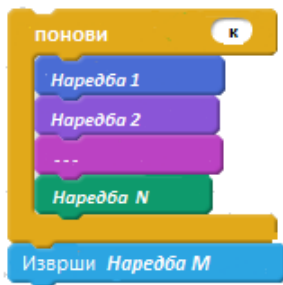
- 1 Uvod
- 2 Nastava zasnovana na ishodima
- 3 Osnovni pojmovi i ideje
- 4 Raspoloživi nastavni materijali
- 5 Algoritmi i primeri**
 - Opisivanje algoritama
 - Kontrola toka programa
 - Izračunavanje
 - Predstavljanje podataka
 - Očekivani problemi i teme
- 6 Između dva dela obuke

Načini opisivanja algoritma

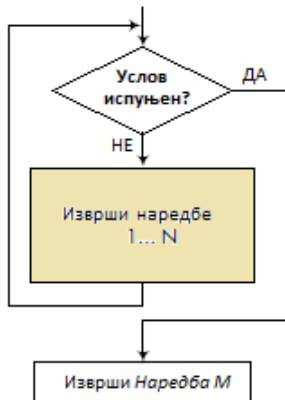
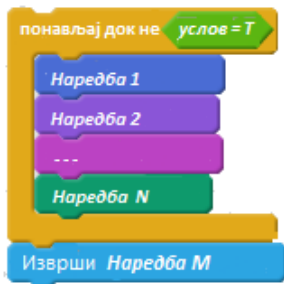
Postoji više načina da se opiše algoritam

- Prirodno-jezički opis
- Pseudokod
- Blok-dijagrami
- Blockly/Scratch dijagrami
- Programski kôd

Primer: Scratch i blok dijagrami



Primer: Scratch i blok dijagrami



Načini opisivanja algoritma

Blok dijagrami

- Dugo korišćeni, bili su skoro sinonim za algoritam iako su oni samo jedna od tehnika za opis algoritama
- **Blok dijagrami su prevaziđen način opisa algoritama**
- Blok dijagrami u klasičnom obliku nemaju mogućnost automatske provere i izvršavanja
- Danas postoje neka okruženja koja mogu da izvršavaju blok dijagrame, npr. <http://www.flowgorithm.org>, ali kako postoje i značajno bolji pristupi, **blok dijagrami se u svetu više ne koriste u nastavi za decu**

Načini opisivanja algoritma

Blockly/Scratch/Code.org dijagrami

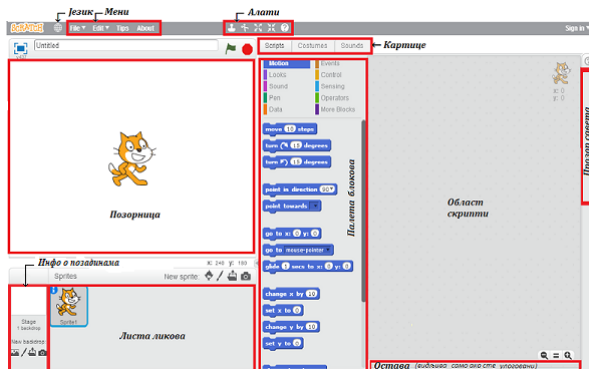
- Ovi dijagrami takođe sadrže formalnu grafičku notaciju za opis algoritama
- Smatra se da je sama notacija intuitivnija i bliža deci od notacije blok dijagrama
- **Važno: donose bogato, savremeno programsko okruženje**
- Nema potrebe opterećivati decu sa dve-tri različite, suštinski ekvivalente formalne grafičke notacije

Uvod

Nastava zasnovana na ishodima
Osnovni pojmovi i ideje
Raspoloživi nastavni materijali
Algoritmi i primeri
Između dva dela obuke
Zaključci

Opisivanje algoritama
Kontrola toka programa
Izračunavanje
Predstavljanje podataka
Očekivani problemi i teme

Scratch donosi savremeno programsko okruženje



Načini opisivanja algoritma

Programski kôd za opis algoritama

- Veoma je važno pisati i jednostavne konzolne programe
- Pisanje konzolnih programa veoma je važno za usvajanje koncepta algoritma, ali to je već napredniji nivo (6. razred)
- Na primer, za vežbanje petlji, nezaobilazni su programi poput
 - ispisa nekog teksta n puta,
 - ispisa brojeva od 1 do n ,
 - ispisa svih parnih brojeva od 1 do n ,
 - učitavanja dok se ne unese neka posebna oznaka kraja (prazan red, nula).

Algoritmi

Algoritmi su izgrađeni nad narednim pojmovima

- Kontrola toka
- Izračunavanje
- Predstavljanje podataka i rad sa podacima

Šta je sa “pravim”, velikim programima?

To važi i za “prave”, velike programe!

Algoritmi

Kontrola toka

- Nizanje naredbi
- Grananje
- Ponavljanje

Šta je sa “pravim”, velikim programima?

Ništa drugo nije potrebno ni za “prave”, velike programe!

Nizanje naredbi — sekvencijalni algoritmi

Primer sekvencijalnog algoritma

- Kada idemo na spavanje treba da:
 - 1 obučemo pižamu
 - 2 operemo zube i umijemo se
 - 3 legnemo u krevet

Redosled koraka u sekvencijalnom algoritmu

- Sinonimi: nizanje naredbi, izvršavanje redom, sekvencijalni algoritmi, linearni algoritmi
- Nekada se redosled koraka može promeniti, ali nekada ne može
- Na primer, možemo prvo da operemo zube i umijemo se, a zatim da obučemo pižamu, ali ne možemo da legnemo u krevet pa da peremo zube

Grananje u algoritmima

Primer grananja

Kada izlazimo, ako pada kiša: uzimamo kišobran;
inače, uzimamo kačket.

Grananje

- Izvršavanje koraka može zavisiti od uslova
- **Potpuno grananje:** ukoliko izvršavamo neke naredbe u slučaju da je uslov ispunjen, a u slučaju da uslov nije ispunjen, izvršavamo neke druge naredbe (kao u prethodnom primeru)
- **Nepotpuno grananje:** ukoliko izvršavamo neke naredbe samo u slučaju da je uslov ispunjen (na primer, ako pada kiša, uzimamo kišobran, a inače ne uzimamo ništa)

Ponavljanje u algoritmima: primer

Primer ponavljanja

- Do kada se igra taj-brejk u tenisu?
- Sve dok jedan igrač nije osvojio barem 7 poena i barem dva poena više od protivnika



Ponavljanje u algoritmima

Broj ponavljanja

- Koraci u algoritmima se mogu ponavljati
- Broj ponavljanja može biti fiksiran i unapred poznat (na primer, 10, 100, 1000 puta)
- Broj ponavljanja može biti nepoznat, kao u prethodnom primeru
- **Ponavljanja su đacima konceptualno najteži pojam, posebno ukoliko broj ponavljanja nije fiksiran**

Digresija: skokovi u algoritmima

Idi na korak N

- Koraci u algoritmu mogu biti numerisani i može se koristiti konstrukcija *Idi na korak N*
- To je često pogodno i jasno u životnim primerima
- Skokovi su se intenzivno koristili u ranom programiranju, a danas se više ne koriste, jer u velikim programima narušavaju razumljivost koda

Digresija: skokovi u algoritmima

Primer za *idi na korak N*: kako radi ATM mašina

- 1 Ubaci karticu u ATM mašinu
- 2 Unesi PIN kod
- 3 Ako mašina prijavi da je uneti kod pogrešan, idi na korak 2
- 4 Unesi iznos za podizanje
- 5 Ako mašina prijavi da uneti iznos ne može biti isplaćen, idi na korak 4
- 6 ...

Kako predavati/učiti/vežbati kontrolu toka?

Početno razvojno okruženje

- Učenicima je po pravilu zanimljivije da programiranje bude u grafičkom nego u tekstualnom okruženju
- Još je zanimljivije ako su zadaci unapred pripremljeni u obliku malih izazova i ako učenik dok rešava programerski zadatak ima utisak da igra igricu

Zadaci za 5. i 6. razred

Suštinski isti koncepti - ali sa bitnom razlikom

- U petom razredu slažu se blokovi i ne postoji mogućnost pravljenja sintaksne greške.
- U šestom razredu, đaci moraju da nauče osnovnu sintaksu programskog jezika Python, kao i da se suoče sa situacijom kada naprave sintaksnu grešku.
- U oba slučaja, moguće je praviti semantičke greške (tj imati pogrešne ideje koje vode izračunavanju pogrešnog rezultata)

Zadaci za 5. i 6. razred: primer

Da li je broj unet sa tastature paran ili neparan?

Rešenje za peti razred

Scratch code for checking if a number is even or odd:

- kada je kliknuto na [zastavu]
- питај Молим те, унеси број, и чекај
- ако је `одговор мод 2 = 0` онда
 - изговори Унети број је паран! током 2 секунде
- у супротном
 - изговори Унети број је непаран! током 2 секунде

Rešenje za šesti razred

```

broj = int(input("Unesi broj"))
if (broj % 2 == 0):
    print("Broj je paran")
else:
    print("Broj je neparan")
  
```


Kako predavati/učiti/vežbati kontrolu toka?

Grafičko okruženje

- Postoje dva klasična okruženja koja se još od 1960-ih pa do danas koriste u početnoj nastavi programiranja.
 - Kornjačina grafika
 - Karel robot
- Materijali na veoma popularnom sajtu <http://code.org> zasnovani su upravo na ova dva pristupa.

Kornjačina grafika

Šta je kornjačina grafika?

- Uvedena još u programskom jeziku Logo
- Koristi se na sajtu <http://code.org> - npr. *Frozen tutorijal*
- Direktno podržana i u sistemu MIT Scratch i u jeziku Python
- Programer zadaje naredbe za kretanje kursora (kornjače) po ekranu
- Kornjača tokom kretanja ostavlja trag
- Zahteva samo poznavanje jedinica mere dužine i uglova (u stepenima)

Osnovne naredbe kornjače

naredba

Scratch

Python

idi napred n koraka

иди 10 корака

`turtle.forward(n)`

okreni se ulevo n stepeni

усмери се ка 90

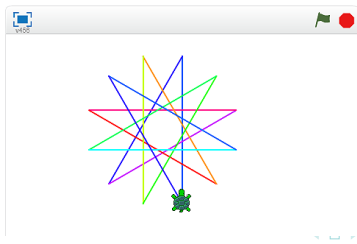
`turtle.left(n)`

okreni se udesno n stepeni

усмери се ка -90

`turtle.right(n)`

...



Kornjačina grafika: primeri

Primer za peti razred

- Pred tobom je program za crtanje kvadrata. Šta treba da promeniš, pa da lik iscrtava pravougaonik?
- <http://petlja.org/BubbleBee/r/lekcije/scratch-support/kornjaca>

Primer za šesti razred

- Napiši program u kojem kornjača crta kvadrat čija je dužina stranica 100 koraka.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/kornjaca>

Detaljnije o kornjača grafici

Priručnik za peti razred

<http://bubblebee.petlja.org/Media/Default/Programiraj/V/lekcija5/Scratch%20Kornjaca%20grafika.pdf>

Priprema časa za peti razred

<http://bubblebee.petlja.org/Media/Default/Programiraj/V/lekcija5/Priprema%205%205%20razred.pdf>

Detaljnije o kornjača grafici

Priručnik za šesti razred

<http://bubblebee.petlja.org/Media/Default/Programiraj/VI/lekcija7/prirucnik7.pdf>

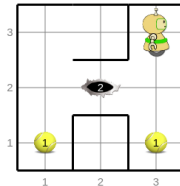
Priprema časa za šesti razred

<http://bubblebee.petlja.org/Media/Default/Programiraj/VI/lekcija7/priprema7.pdf>

Karel robot

Šta je Karel robot?

- Uveo ga je Ričard Patis u knjizi "Karel The Robot: A Gentle Introduction to the Art of Programming"
- Robot se kreće kroz lavirint sakupljajući i premeštajući loptice
- Za razliku od kornjače, može da vrši interakciju sa okruženjem (da ispituje) – pogodno za vežbanje grananja i petlji



Karel robot

Šta je Karel robot?

- Ranije poseban programski jezik, a kasnije biblioteka za razne programske jezike (Java, JavaScript, Pascal, Python, ...)
- Postoje različite implementacije u Python-u, ali nije deo standardne biblioteke
- **Implementirali smo novu biblioteku u Python-u (naredbe robotu na srpskom jeziku)**

Karel robot

Osnovne komande

- `napred()` - pomeri se jedno polje napred
- `levo()` - okreni se za 90 stepeni nalevo (u smeru suprotnom kazaljki na satu)
- `uzmi()` - pokupi lopticu sa polja na kojem se nalaziš
- `broj_loptica_na_polju()` - vraća broj loptica na polju na kojem se robot nalazi
- ...

Karel robot: primeri

Primer za šesti razred

- Naredni lavirint je začaran i ne zna se koliko tačno polja postoji između robota i loptice. Korišćenjem petlje while učini da robot dođe do loptice i da je zatim pokupi.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/karel>

Primer za šesti razred

- Pomozi robotu da pokupi sve loptice.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/karel>

Ponavljjanje: primeri

Primer za peti razred

- Pomozite mami da nacrtava četiri stepenika dužine i visine po 50 koraka.
- <http://petlja.org/BubbleBee/r/lekcije/scratch-support/ponavljanje>
- Pogledajte i video (1min 43s)

Primer za šesti razred

- U igri žmurke deca broje 5, 10, 15, itd., do 100. Napiši program koji ispisuje baš te brojeve.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/ponavljanje>

Grananje: primer

Primer za peti razred

- Kako napraviti kviz?
- Priručnik

`http://bubblebee.petlja.org/Media/Default/
Programiraj/V/lekcija9/Priru%C6%92nik%20za%20u%C6%
92enike%209-Kviz-V%20razred.pdf`

- Priprema časa

`http:
//bubblebee.petlja.org/Media/Default/Programiraj/
V/lekcija9/Priprema%209%20-%20Kviz-V%20razred.pdf`

Grananje: primer

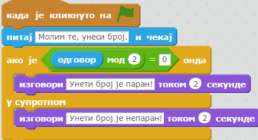
Primer za šesti razred

- Napišimo program koji korisnicima mlađim od 15 godina šalje posebnu pohvalu jer su krenuli da programiraju veoma rano.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/grananje>

Grananje: primer

Da li je broj unet sa tastature paran ili neparan?

Rešenje za peti razred



Rešenje za šesti razred

```
broj = int(input("Unesi broj"))
if (broj % 2 == 0):
    print("Broj je paran")
else:
    print("Broj je neparan")
```

Objašnjenje

Pogledajte video sa objašnjenjem ovog zadatka

<http://petlja.org/BubbleBee/r/lekcije/scratch-support/odlucivanje-granje>

Izračunavanje

Izračunavanje i veza sa matematikom

- Ne treba insistirati na teškim matematičkim zadacima — **matematički zadaci za 3. i 4. razred OŠ su sasvim odgovarajući**
- ... ali, ne treba ni bežati po svaku cenu od matematike (naročito u radu sa boljim učenicima)
- Program u jeziku kakav je Python se ne razlikuje puno od precizno zapisanog matematičkog rešenja

Izračunavanje: primer

Primer

Pera ima 1234 dinara, a Ana ima tačno dva puta više od njega.
Koliko dinara imaju zajedno?

```
pera = 1234  
ana = 2 * pera  
zajedno = pera + ana  
print(zajedno)
```


Izračunavanje

Konkretni i opšti brojevi

- Učenici su u matematici naviknuti da rešavaju zadatke zadate nad konkretnim brojevima
- Račun sa opštim brojevima se u nekoj meri sreće u šestom razredu
- Prelazak sa jednog na drugi nivo često nije trivijalan za učenike
- Inicijalizacijom promenljivih na konkretne vrednosti na početku programa može da se izbegne priča o učitavanju
- Loša strana je to da učenici sami kreću da računaju umesto da račun prepuste računaru

Izračunavanje: primer

Primer - konkretni brojevi

Na priredbi dobrodošlice đacima prvacima treba raspodeliti raspoložive bananice deci — tako da svako dete dobije isti broj bananica. Škola je nabavila pet kutija od po 40 bananica, a na priredbu će doći 85 đaka. Koliko će svako dete dobiti bananica, a koliko će bananica ostati neraspodeljeno?

```
ukupno_bananica = 5 * 40
```

```
ukupno_dece = 85
```

```
bananica_po_detetu = ukupno_bananica / ukupno_dece
```

```
neraspodeljeno = ukupno_bananica % ukupno_dece
```

```
print("Svako će dete dobiti", bananica_po_detetu, "bananica")
```

```
print("Ostaće", neraspodeljeno, "bananica")
```

Izračunavanje: primer

Primer - opšti brojevi

Na priredbi dobrodošlice đacima prvacima treba raspodeliti raspoložive bananice deci — tako da svako dete dobije isti broj bananica. Ako se zna koliko će dece doći na priredbu, koliko će svako dete dobiti bananica, a koliko će bananica ostati neraspodeljeno?

```
ukupno_bananica = int(input("Koliko ukupno ima bananica?"))  
broj_dece = int(input("Koliko će dece doći na priredbu?"))  
bananica_po_detetu = ukupno_bananica / broj_dece  
neraspodeljeno = ukupno_bananica % broj_dece}  
print("Svako će dete dobiti", bananica_po_detetu, "bananica")}  
print("Ostaće", neraspodeljeno, "bananica")}
```

Izračunavanje

Izmena vrednosti promenljive

- Koncept promene vrednosti promenljive (mutabilnost) stran je i komplikovan učenicima (jer se u matematici vrednost promenljive ne menja tokom vremena)
- Izbegavati njegovo uvođenje na početku kursa (npr. ako je cena proizvoda promenjena, umesto promenljive cena kojoj se menja vrednost koristiti promenljive `cena_pre` i `cena_posle`)
- Promeni vrednosti promenljivih se možete posebno posvetiti i detaljno je objasniti u drugoj polovini kursa

Izračunavanje: primer

Primer - izmene vrednosti promenljive

[http://petlja.org/BubbleBee/r/lekcije/
interactive-book-py6/deljenje](http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/deljenje)

Cena hleba je bila 35 dinara, zatim je poskupela 10%. Mleko je koštalo 100 dinara, ali je sada na sniženju i prodaje se po 20% nižoj ceni. Koliko koštaju tri hleba i dva mleka?

```
hleb_pre = 35  
hleb_posle = hleb_pre + (10 / 100) * hleb_pre  
mleko_pre = 100  
mleko_posle = mleko_pre - (20 / 100) * mleko_pre  
racun = 3 * hleb_posle + 2 * mleko_posle  
print(racun)
```

Izračunavanje: primeri

Primer za peti razred

- Koji od dva zadata izraza ima negativnu vrednost?
- <http://petlja.org/BubbleBee/r/lekcije/scratch-zbirka/aritmeticki-izrazi-i-promenljive>

Primer za šesti razred

- Na prvoj polici ima 150 knjiga. Na drugoj ima duplo manje nego na prvoj, a na trećoj tri puta manje nego na drugoj. Koliko je ukupno knjiga na policama.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/deljenje>

Funkcije u izračunavanju

Korišćenje bibliotečkih funkcija

- Mnogi programski jezici pružaju funkcije poput `abs`, `min`, `max`, `sum`, ...
- Ako jezik nudi neku funkcionalnost gotovu, ne treba je skrivati od dece i tražiti da je oni sami programiraju
- Ipak, implementacija ovakvih stvari može biti dobra vežba i ima je smisla pokazati, ali tek nakon prvobitne primene gotovih funkcija

Funkcije u izračunavanju: primer

Primer - veća ocena

<http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/ugradjenefunkcije>

Ljubica je jednog dana dobila ocenu iz biologije i geografije. Koja je veća ocena koju je Ljubica dobila tog dana?

```
biologija = int(input("biologija: "))  
geografija = int(input("geografija: "))  
print("veća ocena: ", max(biologija,geografija))
```


Izračunavanje - samostalno definisanje funkcija

Definisanje funkcija

- Razbijanje problema na potprobleme izuzetno je važna veština koju kroz ovaj kurs treba podsticati
- Njemu u programiranju odgovara podela programa na funkcije
- Svi argumenti funkcija mogu da budu isključivo ulazni, a da funkcija vraća vrednost naredbom `return`
- Funkcije treba da vrše jednostavna izračunavanja (obično primenom neke složenije matematičke formule)

Funkcije u izračunavanju: primer

Primer - obim jednakostraničnog trougla

[http://petlja.org/BubbleBee/r/lekcije/
interactive-book-py6/definisanjefunkcija](http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/definisanjefunkcija)

Funkcija koja izračunava obim jednakostraničnog trougla se može definisati i onda upotrebiti na sledeći način.

```
def obim_jednakostranicnog_trougla(a):
```

```
    return 3 * a
```

```
print(obim_jednakostranicnog_trougla(5))
```

```
print(obim_jednakostranicnog_trougla(7.5))
```

```
print(obim_jednakostranicnog_trougla(2))
```

Funkcije u izračunavanju: primer

Primer - površina kvadrata

- Definiši funkciju koja na osnovu dužine stranice izračunava površinu kvadrata.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/definisanjefunkcija>

Predstavljanje podataka

Liste

- Prethodni programi su sadržali promenljive - promenljive mogu da sačuvaju samo jedan podatak (broj, reč, vrednost tačno ili netačno...) To nije dovoljno kada želimo da sačuvamo veći broj podataka, poput telefonskih brojeva, naziva proizvoda i slično
- Liste se mogu posmatrati kao police na kojima se nalazi veliki broj kutija, numerisanih brojevima od 1 pa na dalje, i u svakoj od kutija se čuva po jedan podatak.
- <http://petlja.org/BubbleBee/r/lekcije/scratch-support/liste>

Predstavljanje podataka: primer

Liste

[http://petlja.org/BubbleBee/r/lekcije/
interactive-book-py6/strukturepodataka](http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/strukturepodataka)

Zapitajmo se kako bismo u programu mogli predstaviti spisak imena košarkaša u jednom timu? Neka je to naša košarkaška reprezentacija koja je 2016. igrala na Olimpijadi. Pretpostavićemo da svaki igrač ima svoj redni broj i to od 1 do 12.

```
igrac1 = "Милош Теодосић"  
igrac2 = "Мирослав Радуљица"  
igrac3 = "Богдан Богдановић"  
igrac4 = "Владимир Штимац"  
...
```

Predstavljanje podataka: primer

Liste

<http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/strukturepodataka>

Bolji način da predstavimo spisak igrača je da upotrebimo listu.

```
igraci = ["Stefan Joviћ", "Marko Simonoviћ", "Bogdan Bogdanoviћ", \  
          "Nikola Kaliniћ", "Milan Mачван", "Stefan Markoviћ", \  
          "Nemađa Nedoviћ", "Miroslav Raduљица", "Miloш Teodosiћ", \  
          "Nikola Joкић", "Vladimir Шtimaц", "Stefan Bирчевић"]
```

Pojedinačni članovi liste nazivaju se i elementi.

Predstavljanje podataka: primer

Primer za šesti razred

- Poznat je spisak igrača u timu. Oni nose dresove sa brojevima od 1 pa na dalje. Napiši program koji za dati broj dresa određuje igrača koji igra pod tim rednim brojem.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/strukturedpodataka>

Predstavljanje podataka: primer

Primer za šesti razred

- Poznate su visine devojčica i visine dečaka u jednom odeljenju. Napravi listu svih visina i izračunaj zatim broj i prosečnu visinu svih đaka.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/strukturedpodataka>

Predstavljanje podataka: primer

Primer za peti razred

- Program koji svira muziku samo za nas
- <http://bubblebee.petlja.org/Media/Default/Programiraj/V/lekcija10/Scratch%20-%20Liste.pdf>

Osnovni algoritmi: primeri

Primer za peti razred

- Napiši program koji unosi imena učenika sve dok se ne unese prazna niska i na kraju prijavljuje koliko imena je uneto.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/osnovnialgoritmi>

Primer za šesti razred

- Napiši program koji izračunava prosečnu temperaturu ako je data lista koja sadrži temperature svakog dana tokom nekog perioda.
- <http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/osnovnialgoritmi>

Uvod
Nastava zasnovana na ishodima
Osnovni pojmovi i ideje
Raspoloživi nastavni materijali
Algoritmi i primeri
Između dva dela obuke
Zaključci

Opisivanje algoritama
Kontrola toka programa
Izračunavanje
Predstavljanje podataka
Očekivani problemi i teme

Dodatno

Osnovni algoritmi

[http://petlja.org/BubbleBee/r/lekcije/
interactive-book-py6/osnovnialgoritmi](http://petlja.org/BubbleBee/r/lekcije/interactive-book-py6/osnovnialgoritmi)

Dodatni algoritmi

<http://petlja.org/BubbleBee/r/kursevi/ZbirkaVIRazred>

Očekivani problemi i teme

Eksperimentisanje i destruktivnost

- Učenicima dati prostor da sami probaju nešto i eksperimentišu: osnovni cilj i jeste da se oni zainteresuju za programiranje, a to je najbolji način
- Deca umeju da istražuju kroz destruktivnost: “kako da nateram brauzer da poludi i sve se zaglavi?” (na primer, ako petlja ide do 1000000?) — ne kažnjavati đake zbog takvih postupaka, istraživanje je važno, a nakon destruktivnosti uslediće napredak u znanju

Očekivani problemi i teme

Stil programiranja

- Da bi program bio kvalitetno napisan, treba da bude:
 - **uredan** — lepo nazubljen,
 - **čitak** — deskriptivna imena promenljivih, pravilna upotreba funkcija
 - **ne sme da sadrži ponavljanja** — isti kod ne sme da se pojavljuje više puta već na takvim mestima treba upotrebljavati petlje ili pozive definisanih funkcija

Očekivani problemi i teme

Dobri i loši programi

- Deca često umesto da nauče nešto novo žele da samo iskoriste nešto postojeće (“copy-paste”): na primer, da bi 10 puta uradili nešto oni će 10 puta ponoviti isti kod, umesto da upotrebe petlju — ovo je veoma loša navika i treba stalno objašnjavati i insistirati na tome da je važno da se ne koristi višestruko kopiranje već postojećeg koda
- Ukazati im na razlike između dobrog i lošeg programa (čak i ako im je rezultat rada isti) – nagovestiti značaj stila programiranja

Pregled

- 1 Uvod
- 2 Nastava zasnovana na ishodima
- 3 Osnovni pojmovi i ideje
- 4 Raspoloživi nastavni materijali
- 5 Algoritmi i primeri
- 6 Između dva dela obuke**
 - Utisci i efekti prvog dela obuke
 - Često postavljana pitanja
- 7 Zaključci

Šta nas čeka na nastavku obuke?

Naredna obuka

- **Važno je da detaljno pročete kroz sve materijale pre naredne obuke**
- Važno je da pripremite pitanja na teme koje Vam nisu jasne kako bi to mogli da razjasnimo sledeći put
- Na kraju obuke, biće održan izlazni test

Utisci sa obuke

Vaše mišljene je važno

Molimo Vas da sada unesete svoje utiske o upravo održanoj obuci - pomozite da naredna obuka bude još bolja!

<https://goo.gl/forms/8AQD3HaDrrgfRbZj1>

„Podrška nastavi informatike u OŠ”

Drugi deo obuke

Drugi deo obuke nastavnika

Projekat „Podrška nastavi informatike u OŠ”

- Ovo je drugi deo obuke nastavnika u okviru projekta **Podrška nastavi informatike u OŠ**
- Projekat organizuje **Fondacija Petlja**
- Pomoć i podršku projektu pružaju
 - **Vlada Republike Srbije**
 - IT kompanija **Microsoft**
 - Neprofitno udruženje vodećih IT kompanija **Inicijativa Digitalna Srbija**
 - **Matematički fakultet Univerziteta u Beogradu**

Drugi deo obuke nastavnika

Projekat „Podrška nastavi informatike u OŠ”

- Akcioni plan Vlade o informatici predviđa, između ostalog, **uvođenje informatike i programiranja u osnovnu školu**
- To je važno za celokupni obrazovni sistem i za ekonomiju Srbije
- Informatika u školama je važna đacima koji će raditi u IT sektoru, ali i svima drugima jer **programiranje nas uči da razmišljamo precizno i analitički**

Prvi deo obuke

Prvi deo obuke

- Izvanredan odziv: više od 900 nastavnika širom Srbije prijavilo se za obuku
- Obuka je organizovana u mnogim gradovima: Aleksinac, Bela Palanka, Beograd, Ljig, Sremska Mitrovica, Vranje, Žitkovac...
- Obuku je prošlo nekoliko stotina nastavnika
- Stvorena je široka baza, osnova za zajednicu koja će zajednički unapređivati nastavu informatike

Na prvom delu obuke ...

Upoznali smo se sa...

- ... motivima za uvođenje programiranja u osnovne škole
- ... ciljevima i očekivanim ishodima učenja
- ... osnovnim principima učenja programiranja
- ... važnošću algoritamskog načina razmišljanja
- ... programskim jezicima i okruženjima
- ... osnovnim nastavnim materijalima
- ... sistemom za praćenje rada učenika

Uradili smo ...

- ... različite primere i algoritme

Utisci trenera

Treneri su veoma pozitivno ocenili

- aktivnost nastavnika-polaznika
- zainteresovanost nastavnika-polaznika

Utisci nastavnika

Nastavnici koji su pohađali obuku ocenili su je veoma dobro (na bazi 135 anektiranih nastavnika):

- **4.59** (na skali 1-5) na pitanje “Da li smatrate da će Vam ova obuka pomoći u pripremi nastave?”
- **4.57** (na skali 1-5) na pitanje “U okviru obuke obrađene su najznačajnije teme koje mi objašnjavaju motivaciju i suštinu časova programiranja?”
- **4.85** (na skali 1-5) na pitanje “Planiram da koristim predstavljene materijale i resurse”
- ...

Utisci nastavnika

U anketi nastavnici-polaznici često su sugerisali da se poveća udeo praktičnog rada

To je upravo i plan za drugi deo obuke

Danas ...

Odgovorićemo ...

- ... na neka postavljena pitanja od prošlog puta

Uradićemo ...

- ... nove praktične primere i algoritme

Pitanja i odgovori

Da li đaci mogu da vide na forumu Algora deo za nastavnike?

Trenutno mogu. Zbornica će biti zatvorena samo za nastavnike ali zbog obima posla ne možemo da kažemo kada će to tačno biti.

Ako Zbornica bude zatvorena za obične naloge, kako nastavnici onda mogu tražiti nastavnički nalog preko iste?

Ideja je da nastavnik koji je bio na obuci može da dostavi kontakt svog kolege na loop@petlja.org (to daje garanciju da je u pitanju kolega, a ne neki roditelj).

Pitanja i odgovori

Da li, u statistici gde se prati rad učenika, može da se vidi ceo odgovor učenika (misli se na zadatke u kojima treba da se dopuni kod), a ne samo da li je odgovor tačan ili ne?

Da, biće napravljeno. Potrudimo se da bude što pre.

Da li je moguće da se u sistemu za praćenje rada učenika, u delu Statistika, ubaci tajmer koji će da beleži vreme kada je neki zadatak rešen, da bi na taj način nastavnik znao da li djak sa ueničkim nalogom zadatke rešava kod kuće ili na času?

Može. Dodaćemo vreme u detaljan pregled statistike.

Pitanja i odgovori

Staviti link na glavnu stranu ka materijalima nastalim u okviru projekta „Programiraj MTT“, ili barem u kontrol panel vidljiv nastavnicima.

Pretpostavljam da se radi o linku

<http://petlja.org/Programiraj?> Link postoji u delu <http://petlja.org/OsnovnaSkola> koji je polazna tačka za podršku nastavi.

Logout ne radi kako treba

Trebalo bi da je ovaj problem rešen. Ukoliko primetite ponovne probleme na ovu temu, molimo Vas da ih prijavite.

Pitanja i odgovori

Šifre

Šifra je neophodna kako niko ne bi bio u mogućnosti da koristi tuđe naloge.

Deljenje šifri nije nešto što treba raditi često. Šifre treba podeliti jednom, na početku. Učenik treba da zapamti svoju šifru i koristi je. Nastavnik treba da resetuje šifru u slučaju da je učenička šifra kompromitovana, recimo neko drugi pristupa njegovom nalogu.

S obzirom da je onemogućavanje pristupa van časa naišlo na veću publiku, razvićemo tu opciju na drugačiji način. Ne kroz resetovanje šifre već kroz enable/disable mehanizam koji će moći da se upotrebi na početku i na kraju časa.

Pitanja i odgovori

Da li u jednoj grupi učenički nalozi mogu imati jednu zajedničku šifru?

Ne.

Da li moram praviti učeničke šifre, ionako gubim 10 - 15 minuta svakog časa?

Da, ali učeničke šifre ne treba praviti često već samo jednom.

Učenik sa svog naloga koji mu je nastavnik napravio može da napusti grupu u koju ga je nastavnik raspodelio. Mislim da bi ovo trebalo da se onemogući.

Biće ispravljeno u najkraćem roku.

Pitanja i odgovori

Da li je moguće napraviti jedno dugme koje bi omogućilo refresh-ovanje svih šifara u jednog grupi?

Da, biće napravljeno. Potrudimo se da bude što pre.

Da li bi mogla da se doda opcija da se šifre koje su generisane, kada se napravi grupa sa nalozima, blokiraju na neki način i da se pred početak časa šifre odblokiraju, da ne bi nasatvnici morali pred svaki čas da generišu nove šifre?

Napravićemo opciju disable-ovanja naloga kroz kontrolnu tablu. Sa disable-ovanim nalogom neće biti moguće ulogovati se.

Pitanja i odgovori

Da li se učenik prijavljuje sa istim učeničkim nalogom i na BubbleBee?

Da. Ako se misli na stari `bubblebee.petlja.org`, nakon nekoliko meseci bez podrške, konačno smo ugasili tu verziju. Novi BubbleBee je na `petlja.org/BubbleBee`.

Zašto ne može nastavnik da pravi imena učeničkih naloga sam (da napravi sopstvenu kombinaciju naziva)?

Zbog jednostavnosti ali i zbog poverljivosti ličnih informacija.

Pitanja i odgovori

Da li će se na takmičenjima za peti razred raditi Skreč?

Ne.

Da li se u radu sa boljom decom može raditi C,C++,C#?

C baš nikako ne bih savetovao za rad sa malom decom. Moguće je raditi C++ i C#, ali i Pajton koji je zvanično podržan ove godine na takmičenjima.

Da li će se na takmičenjima sa šesti razred koristiti isključivo Pajton?

Ne. Lista podržanih jezika i okruženja je objavljena na sajtu DMS.

Pitanja i odgovori

Čemu služi deo „Takmičenja” u kontrolnom panelu ?

Petlja omogućava online takmičenja kroz takmičarsko okruženje. Ovaj deo nije specifičan za nastavu programiranja. Ima smisla u nekim situacijama koristiti takmičenja u nastavi ali to daleko izlazi iz okvira naše podrške. U nekom trenutku ćemo pisati na Algori o mogućnostima koje Petlja nudi i kako najbolje da se iskoriste i biće reči o takmičenjima.

Pitanja i odgovori

Da li se sve oblasti moraju preći i uopšte kolika je količina gradiva koja se mora preći?

Potrebno je ispuniti sve ishode. Način na koji će se to postići zavisi od nastavnika. Savet: poželjno je sa svima preći ono što je obeleženo zelenom bojom. Žute i crvene stvari treba raditi samo ako se proceni da deca mogu da to usvoje.

Časovi su neusklađeni sa mogućnostima dece i materijli se u nastavi prelaze prebrzo.

Petlja nudi širok dijapazon materijala. Tempo treba uskladiti sa mogućnostima dece. Nastavu treba individualizovati što više. Nigde ne piše da treba da se pređe sa svom decom sve što je ponuđeno.

Pitanja i odgovori

Kog će oblika biti provere znanja, pored onih testova koji mogu da se štampaju?

To zavisi od nastavnika. Na njemu je potpuna sloboda i odgovornost da organizuje ocenjivanje učenika. Ovi testovi su samo mala pomoć i podrška (a, naravno, ne moraju se koristiti).

Da li može biti omogućeno da nastavnici sami biraju pitanja za testove?

Može. Radićemo na tome.

Da li može biti omogućeno da se generiše test sa permutovanim pitanjima?

Ne, to nije moguće.

Pitanja i odgovori

Da li bi test mogao online da se radi?

Na Petlji ne. Petlja samo pruža pomoć, ali nikako ne želi da utiče na ocenjivanje učenika.

Primedba da su samo najlakša pitanja/zadaci urađeni u priručnicima, voleli bi da su urađena i neka teža.

Možda će još nešto malo biti dodato. U početku je skoro sve bilo urađeno, ali onda učenicu nisu praktično imali ništa da rade i to je više bio udžbenik nego interaktivni priručnik.

Pitanja i odgovori

Da li se materijal sa Petljinog sajta može prebaciti u pdf?

Ideja je da sve bude interaktivno i da se ne čita ležeći u krevetu, nego isprobavajući uz računar.

Da li će biti objavljena prezentacija sa obuke?

Nakon odrađenog kompletnog prvog kruga obuka, biće objavljena prezentacija.

Pitanja i odgovori

Da li će biti testova za Skreč?

U okviru Praktikumuma za peti razred postoji širok izbor pitanja kakva se mogu koristiti za proveru znanja, tako da primera i ideja za pitanja za testove ne nedostaje. Jedino što ne postoji mehanizam za automatizovano kreiranje testova kao za šesti razred, jer je taj mehanizam baziran na tekstualnom izvornom kodu.

Da li može biti omogućeno da na testu pored broja pitanja stoji kog je nivoa?

Može. Napravićemo tako da je to opciono pri generisanju instance testa.

Pitanja i odgovori

Može li Pajtom 3.6xxx da se instalira na Windows XP?

Moramo da skrenemo pažnju da Windows XP više nije podržan od strane proizvođača. U slučaju da koristite Windows XP, najbolje je da instalirate Pajton 3.4.

Pitanja i odgovori

- (1) Da li bi moglo da se obezbedi zasebno uputstvo koje se tiče sintakse Pajtona?
- (2) Molba da se na sajtu Petlje napravi spisak Pajton funkcija koje se često koriste sa detaljnim opisom šta i kako rade.

Hvala na predlogu. Biće napravljeno.

Pošto je prošle godine informatika u 5. razredu bila izborni predmet i ako ga učenik nije birao, pa samim tim ne zna Skreč, kako raditi sa tim učenikom Pajton, ako u 6. razredu izabere informatiku?

Skreč nije neophodan preduslov za učenje Pajtona. Pomaže možda malo, ali može se i bez njega.

Pitanja i odgovori

Da li nastavnici mogu da ubacuju svoje zadatke/pitanja/materijale na petlji?

Ne još, ali je u planu - pre svega za klasične algoritamske zadatke.

Da li Petlja može izdati knjigu?

Nije u planu.

Da li će izaći udžbenik za 5. i 6. razred?

To je na izdavačkim kućama. Petlja se ne bavi time.

Pitanja i odgovori

Hoće li biti dodato još materijala za učenike 5. razreda, konkretno materijali za ostalih 20 časova iz informatike, jer se Skreč radi samo na 16 časova?

Petlji je u fokusu samo programiranje.

Da li mogu da se „pišu“ programi u Skreču direktno na Petlji kao što mogu za Pajton?

Ne. Komplikovana je implementacija tako nečega, a postoji *Scratch online*.

Pitanja i odgovori

Da li se pored Pajtona može raditi i neki drugi programski jezik?

Da, ali nema potrebe u 6. razredu.

Koji programski jezici su predviđeni da se rade u sedmom i osmom razredu, da li se isto radi Pajton?

Da.

(1) Zašto se ne koristi Basic?

(2) Zašto je Visual Basic zabranjen na takmičenjima?

Basic stvara loše navike u programiranju.

Pitanja i odgovori

Da li je moguće pomeriti termin drugog dela obuke?

Ne, nije moguće iz logističkih razloga (osim u izuzetnim okolnostima).

Da li je moguće pohađati drugi deo obuke na drugom mestu i u drugom terminu?

Ne, nije moguće iz logističkih razloga (osim u izuzetnim okolnostima).

Pregled

- 1 Uvod
- 2 Nastava zasnovana na ishodima
- 3 Osnovni pojmovi i ideje
- 4 Raspoloživi nastavni materijali
- 5 Algoritmi i primeri
- 6 Između dva dela obuke
- 7 **Zaključci**
 - Šta smo sve čuli
 - Neki saveti
 - Šta nas čeka?

Šta smo sve čuli

Pregled materijala

Napravili smo pregled materijala iz programiranja za predmet Informatika za peti i šesti razred

Osnovni cilj

- **Kroz informatiku unaprediti rasuđivanje učenika i zainteresovati ih za programiranje**
- I mali pomak u rasuđivanju kod učenika značiće mnogo
- I mali pomak u poznavanju programiranja značiće mnogo i učenicima i celom IT sektoru i svima nama

Šta smo sve čuli

Programiranje u petom i šestom razrdu

- Nastavni planovi nisu zamišljeni preambiciozno!
- Ovi predmeti moraju da budu shvaćeni ozbiljno kao i svi drugi predmeti, ali ne smeju da budu teški
- Ne smeju se svesti na „bubanje“, već na usvajanje načina razmišljanja i nekoliko najvažnijih ideja
- Objasnije učenicima smisao učenja programiranja

Iste ideje i pojmovi - peti i šesti razred

Insistirati na vezi između ideja i pojmova u petom i šestom razredu

Šta smo sve čuli

Željeni ishodi

Osnovni cilj je unaprediti rasuđivanje učenika i zaniteresovati ga za programiranje, a neki od ishoda su

- Učenik ima predstavu o tome šta je algoritam
- Učenik shvata kako se jednostavna matematička izračunavanja mogu sprovesti programski
- Učenik shvata pojmove linearnog postupka, granajućeg postupka, ponavljajućeg postupka
- Učenik je u stanju da jednostavne zadatke iz svakodnevnog života opiše precizno i reši programski

Neki saveti

Ako deca znaju bolje

- Nemojte imati strah od učenika koji dobro barataju računarima
- Deca (od 11-12) godina mogu već biti veoma vešta u raznim stvarima, ali još uvek ne u programerskom rasuđivanju
- Ako neko dete već jeste vešto i u programiranju – tim bolje! Ohrabrite ga dodatno, neka povuče napred i drugu decu, pošaljite ga na takmičenje...

Neki saveti

Deca vole nadmetanja

- Deca vole nadmetanja i takmičenja: organizujte mala takmičenja u razredu, školi
- Motivшите decu i takmičenjima na višem nivou

Deca vole igre

- Pretvorite časove u igru, mnogi zadaci se i odnose na male igre
- Dozvolite im da i sami smisle nove zadatke i igre
- Neka deca vole da prave stvari od drveta i papira, pokažite im da i na računaru mnogu da prave mnogo toga
- Pravite što više veza sa stvarnim svetom

Neki saveti

Deca vole da eksperimentišu

- Imajmo na umu da decu novih generacija mrzi da čitaju i da uvek traže lakši način
- Iskoristimo tu osobinu, kao prednost u ovom predmetu: neka eksperimentišu i bez mnogo čitanja, neka traže više načina da urade istu stvar
- Mnogi zadaci mogu da se urade na više načina, budite otvoreni za različita učenička rešenja

Neki saveti

Šta ako nema struje?

- Čas programiranja može se održati i bez struje
- Na primer, za vežbanje robota Karela, možete poredati klupe u lavirint, jedno dete postaviti da bude robot, a zatim sa ostalom decom na tabli možete smišljati programa koje dete u lavirintu izvršava. Greške u programima će biti lako uočljive
- Budite kreativni!

Šta nas čeka?

Komunikacija

- Nemojte imati strah od ovih časova
- Svi smo po prvi put u ovakvom predmetu
- Niste sami: postoji organizovana podrška
- Budimo u stalnoj komunikaciji, budite u komunikaciji sa drugim nastavnicima informatike preko namenskog foruma

<http://algora.petlja.org/c/zbornica>

Šta nas čeka?

Ovaj predmet je izazov za sve nas

- Iskoristimo priliku da deci pružimo nešto novo i vredno
- Za pravljenje softvera nije potrebna laboratorija niti skupa oprema!
- U Srbiji se pravi softver koji je jednako dobar kao i onaj koji se pravi u Americi!
- Isto tako: u bilo kom selu, neko dete može da pravi programe jednako dobre kao i oni koje prave deca u velikim gradovima!

Šta nas čeka?

Ne zaboravite

Algoritamski način razmišljanja je ključan za uspešnost u bilo kom poslu

Imajte na umu

Mnoge od vas će đaci pominjati kao nastavnike koji su ih naučili da rasuđuju!