

Radni list – Uvod u programiranje

Modul: Uvod u programiranje

Trajanje: 90 minuta

Nivo: Početni

Ciljna grupa: Gluve mlade osobe koje uče osnovnu digitalnu pismenost

Napomena za nastavnika:

Podstaknite učesnike da rade u parovima. Koristite stvarne ili simulirane pseudokodove ili primere iz imejlova i dozvolite praktično korišćenje uređaja radi angažovanja.

Zagrevanje

Brza provera (plenum, cela grupa)

Pitanje

1. Coding = pisanje insturkcija, Kodiranje je deo programiranja.

Programiranje je šire: znači ceo process stvaranja digitalnog produkta

yes ☐

no ☐

2. Kodiranje i programiranje imaju isto značenje. Nema razlike.

yes ☐

no ☐

Deo 1 — Značaj u današnjem digitalnom svetu

Brainstorming:

Vaš svakodnevni život: Razmislite o svim aplikacijama koje koristite i kojima je potrebno programiranje..

Primeri: džepni kalkulator, semafor, budilnik sa svetlom i funkcijom odlaganja

1) Napišite ili nacrtajte još najmanje 5 primera.

2) Grupna diskusija

Predstavite svoje primere i diskutujte sa drugima

Deo 2 – Ključni elementi – osnovni koncepti

U nastavku ćete videti važne ključne elemente:

2.1 Algoritmi

2.2 Strukture podataka, promenljive i tipovi podataka

2.3 Strukture upravljanja, tok upravljanja

2.4 Osnovni operatori i izrazi

2.5 Debugovanje

2.1 Algoritam

= korak po korak vodič za rešavanje problema, kao što je recept za kuvanje

Vežba: Proverite svoj frižider

Razvijte algoritam sortiranja za frižider sa pravilom „ako je istekao rok, bacite“. Testirajte sa stvarnim proizvodima i prilagodite efikasnost.

Ovako bi frižider trebalo da izgleda kada se sve skloni.



Počnimo:

Mleko > pakovanje 1 > na bočnim vratima frižidera, donji odeljak

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

2.2 Struktura podataka, promenljive i tipovi podataka

Metode za organizovanje i čuvanje podataka, kao što su liste ili rečnici

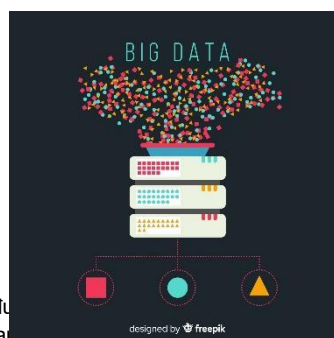
Primeri: brojevi, tekst

Vežba: Recept za kolač od jabuka

160 g putera - 150 g šećera - 1 kesica vanilin šećera - 3 jaja - 270 g glatkog brašna - 3 kašičice praška za pecivo - 1 prstohvat soli - 4 jabuke - 4 kašike mleka - 50 g badema

Razmislite o strukturi podataka u obliku tabele i unesite sve ispravno.

Svaka vrednost mora biti dodeljena..



[illegible]

2.3 Kontrolne strukture, tok kontrole, if-else-flow

if-else: dozvolite programu da donosi odluke.

„ako“ je uslov tačan > uradite jednu stvar.

„inače“ > uradite nešto drugo.

Primer: Da li mi je danas potrebna jakna?

Napravite tok programa: „Ako je hladno, obući ću jaknu, inače neću.“

Uslov: „Da li je hladno?“

Ako je odgovor „da“ → obucite jaknu.

Ako je odgovor „ne“ → ne oblačite jaknu..



Primer: Napišite if-else petlju sa funkcijom semafora.

.....

.....

.....

.....

.....

.....



2.4 operatori i izrazi

Delovi koda koji se mogu ponovo koristiti i obavljaju određene zadatke.

Operatori i izrazi se mogu dobro objasniti pomoću „izračunavanja i upoređivanja“ i „delova koji se mogu ponovo koristiti“ kao što su male pomoćne funkcije.

Neki operatori će vam biti poznati sa časova matematike.

Primeri:

+ (plus): Sabira dva broja, npr. $5 + 3$ jednako je 8.

- (minus): Oduzima, npr. $10 - 4$ jednako je 6..

* (Množenje): Množi, npr. $4 * 3$ jednako je 12

/ (Deljenje): Deli, npr. $12 / 3$ jednako je 4.

Vežba:

Zamislite da želite da izračunate koliko sati ukupno provodite na internetu.

Izračunajte ovo za jednu nedelju..

- 1) Koje operatore ste koristili?
- 2) Kakav je vaš rezultat?

.....

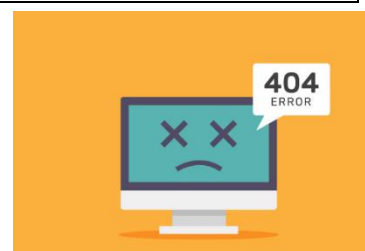
.....

2.5 Debugovanje

Debugovanje znači pronalaženje i ispravljanje grešaka u programu tako da radi ono što želite.

vežba: Igra crtanja

Rad u paru: testiranje – poboljšanje – testiranje



Jedan učesnik je „programer“, a drugi je „robot“.

Programer daje veoma precizna uputstva korak po korak kako da nacrtá sliku.

Kada nešto krene po zlu ili izgleda čudno, zaustavite se i recite: **„Postoji greška u našem programu.“**

Zatim programer mora da promeni instrukciju kako bi robot mogao pravilno da obavi zadatak.

Primeri za crteže:



Deo3 — Brzi kviz

Označite tačan odgovor::

Šta je program?

- A) Vrsta hardvera koji se koristi u računarima.
- B) Igra koja se igra na računaru.
- C) Skup Instrukcije koje računar prati da bi izvršio zadatak.
- D) Uređaj koji čuva podatke.

2. Šta je primer promenljive u programiranju?

- A) Komanda za štampanje teksta na ekranu.
- B) Mesto za skladištenje podataka, kao što su broj ili tekst.
- C) Poruka koja objašnjava šta kod radi.
- C) Vrsta programskog jezika.

3. Šta petlja radi u programu?

- A) Proverava da li je uslov tačan ili netačan.
- B) Ponavlja skup instrukcija više puta.
- C) Čuva podatke za kasniju upotrebu.
- C) Šalje podatke na drugi računar.

4. Zašto je debugovanje važno u kodiranju?

- A) Da bi se programu dodale nove funkcije.
 - B) Da bi se identifikovale i ispravile greške ili bagovi u kodu.
 - C) Da bi se program brže izvršavao.
 - D) Da bi se kod preveo na različite jezike
- .

Deo 4 — Refleksija

- Koja digitalna veština je bila nova za vas? _____
- Koji zadatak je bio najzabavniji? _____
- Šta biste želeli više da vežbate? _____