



PROGRAMIRANJE
Predavanje 03

Ishod 1

1

**UPRAVLJANJE TIJEKOM
PROGRAMA**

Strana • 3



3

Sadržaj

- Danas ćemo naučiti:
 - Logički izrazi
 - Uvjetno izvršavanje dijelova kôda:
 - Naredbe `if`, `elif`, `else`
 - Petlje:
 - Naredba `while`
 - Naredba `for`

Strana • 4



4

Logički izrazi

- Rezultat logičkog izraza je istina (**True**) ili laž (**False**)
- Logički izraz se može sastojati od **varijabli** (`a`, broj, `i`, ...), **literala** (`72`, `3.55`, `"Miro"`, `True`, ...) i **operatora**
- Najčešći operatori su:
 - Aritmetički: `+`, `-`, `*`, `/`, `%`, `//`, `**`
 - Složeni: `+=`, `-=`, `*=`, `/=`
 - Relacijski: `==`, `!=`, `>`, `<`, `>=`, `<=`
 - Logički: `and`, `or`, `not`

Strana • 5



5

Primjeri logičkih izraza

▪ Primjeri:

- $1 == 2$ (rezultat je **False**)
- $17 == 17$ (rezultat je **True**)
- $87.0 > 11.46$ (rezultat je **True**)
- $a = 46$
 $b = 8$
 $a + b != 82$ (rezultat je **True**)
- $(17 >= 2)$ and $(8 \% 2 == 0)$ (rezultat je **True**)
- $(500 > 1000)$ or $(True)$ (rezultat je **True**)
- $\text{not}(500 > 1000)$ (rezultat je **True**)

Strana • 6



6

NAREDBA IF

Strana • 7



7

Potreba za naredbom **if**

- Uvjetno izvršavanje dijelova kôda se ostvaruje naredbom **if**
- Kako bismo riješili sljedeći zadatak:
 - Učitajte ime, prezime i godinu rođenja korisnika. Ako korisnik nije punoljetan, ispišite „Možete naruciti samo sok”.

Strana • 8



8

Naredba **if**

- Koristimo je kad želimo da se izvrši dio kôda (tj. blok naredbi), ali samo ako je ispunjen zadani uvjet
 - Ako uvjet nije ispunjen, blok naredbi se neće izvršiti
 - Uvjet je neki logički izraz
- Primjer naredbe **if** u pseudokôdu:
 - **Pseudokôd** je neformalan opis nekog algoritma i namijenjen je ljudima, a ne računalima (nije programski jezik)
 1. učitaj x od korisnika
 2. if $x > 0$:
 3. ispiši x
 4. ispiši "Broj je pozitivan!"

Strana • 9



9

Sintaksa naredbe `if`

- Sintaksa naredbe `if`:

```
if logički_izraz:
    # Dio koda koji se izvršava samo ako
    # je logički izraz true
```

- Primjeri naredbe:

```
if a + b >= 20 and c < 50:
    print("Uvjet je zadovoljen!")

a = int(input('Upišite neki broj:'))
if a % 2 == 0:
    print('Upisali ste parni broj')
```

Strana * 10



10

Naredba `elif`

- Ako imamo više od jednog uvjeta za provjeriti, naredba `if` nam nije dovoljna
- Naredba `elif` ne može stajati samostalno, može se samo koristiti iza `if` naredbe
 - Možemo je navesti onoliko puta koliko nam treba
 - Njome kažemo „ako prethodni uvjet nije ispunjen, provjeri ovaj”
- Primjer naredbe `elif` u pseudokôdu:

1. učitaj `x` od korisnika
2. `if x > 0:`
3. ispiši "Broj je pozitivan!"
4. `elif x < 0:`
5. ispiši "Broj je negativan!"

Strana * 11



11

Sintaksa naredbe elif

▪ Sintaksa naredbe elif:

```
if logički_izraz_1:
    # Dio koda koji se izvršava ako je logički izraz 1 true
elif logički_izraz_2:
    # Dio koda koji se izvršava ako je logički izraz 2 true
...
elif logički_izraz_n:
    # Dio koda koji se izvršava ako je logički izraz n true
```

▪ Primjer:

```
ime = input('Upišite ime: ')
if ime == "Ana" or ime == "Pero":
    print('Pozdrav, šefe, kava je skuhanana.')
elif ime == "Iva":
    print('Pozdrav, zaposleniče, ured te čeka.')
```

Strana * 12



12

Način izvršavanja if i elif naredbi

▪ Način izvršavanja if i elif naredbi je sljedeći:

- Izvršavanje kreće od if naredbe i izračunava se njen uvjet:
 - Ako je ispunjen:
 - Izvršava se njen cijeli blok naredbi
 - Izvršavanje se prebacuje na prvu naredbu iza svih elif naredbi
 - Ako nije ispunjen:
 - Izvršavanje se prebacuje na sljedeću elif naredbu
- Izračunava se uvjet uz sljedeću elif naredbu i dalje priča ide jednako kao kod if

Strana * 13



13

Naredba else

- Ponekad želimo izvršiti dio kôda, ali samo ako nije ispunjen niti jedan uvjet
- Naredba **else** ne može stajati samostalno, može se koristiti samo iza **if** ili **elif** naredbi
 - Može se navesti samo jednom i uz nju se ne piše uvjet
- Primjer naredbe **else** u pseudokôdu:
 1. učitaj x od korisnika
 2. **if** $x > 0$:
 3. ispiši "Broj je pozitivan!"
 4. **elif** $x < 0$:
 5. ispiši "Broj je negativan!"
 6. **else**:
 7. ispiši "Broj je nula!"

Strana * 14



14

Sintaksa naredbe else

- Sintaksa naredbe **else**:

```

if logički_izraz_1:
    # Dio koda koji se izvršava ako je logički izraz 1 true
elif logički_izraz_2:
    # Dio koda koji se izvršava ako je logički izraz 2 true
...
elif logički_izraz_n:
    # Dio koda koji se izvršava ako je logički izraz n true
else:
    # Dio koda koji se izvršava ako niti jedan od prethodnih
    # logičkih izraza nije true
  
```

Strana * 15



15

Napomena

- Upoznali smo dva dosta slična operatora: = i ==
- Vrijedi sljedeće:
 - Operator = služi da dodjeljivanje vrijednosti varijabli
 - Operator == služi za provjeru uvjeta

- Primjerice:

```
a = 15
if a == 15:
    # Napravi nešto...
```

Strana • 16



16

Primjer naredbi if, elif i else

- Primjer:

```
a = int(input("Upišite neki cijeli broj: "))
if a < 100:
    print("a je manji od 100!")
elif a < 200:
    print("a je manji od 200!")
else:
    print("a je veći ili jednak 200!")
```

- Opišite tijek izvršavanja programa ako korisnik unese: 150, 300, 50.

Strana • 17



17

Primjeri

1. Napišite program koji od korisnika učitava jedan cijeli broj te ispisuje je li paran ili ne.
2. Napišite program koji od korisnika učitava dva cijela broja i operaciju. Ispravne operacije su $+$, $-$, $*$ i $/$. Ispišite rezultat operacije lijepo formatiran, uz uvjete:
 - Ako operacija nije ispravna, ispišite pogrešku.
 - Ako je drugi broj 0, a operacija dijeljenje, ispišite pogrešku.

Strana • 18



18

Primjeri

3. Napišite program koji od korisnika učitava riječ:
 - Ako korisnik upiše "jedan", napravite sljedeće:
 - Učitajte tri decimalna broja i ako su sva tri broja veća od 10.6 i ako je njihov umnožak manji od 5000.2, ispišite "Slučaj 1"
 - Ako prethodni uvjet nije ispunjen, ispišite "Slučaj 2"
 - Ako korisnik upiše "dva", napravite sljedeće:
 - Učitajte od korisnika tri *bool*. Ako su sva tri istinita ili ako je barem treći istinit, ispišite "Slučaj 3"
 - Ako prethodni uvjet nije ispunjen, ispišite "Slučaj 4"
 - Za sve ostale unose ispišite grešku.

Strana • 19



19

*Napredne opcije

- Skraćena (jednolinijska) opcija naredbe if:

```
if a > b: print("a is greater than b")
```

- Skraćena opcija naredbe if..else:

```
a = int(input("a: "))
b = int(input("b: "))
print("a") if a > b else print("b")
```

- Prazna naredba if:

```
a = int(input("a: "))
b = int(input("b: "))
if b > a:
    pass
```

Strana • 20



20

NAREDBA WHILE

Strana • 21



21

Naredba **while**

- Naredba **while** služi za izradu petlji
 - Petlja je dio kôda koji se izvršava više puta
 - **while** petlja se izvršava dok god je logički izraz jednak **True**
- Sintaksa je sljedeća:

```
while Logički_izraz:
    // Blok koji se izvršava dok je logički izraz true
```

- Primjer:

```
broj = 21
while broj <= 25:
    print("Trenutni broj: " + str(broj))
    broj += 1
```

Strana * 22



22

Dodatne naredbe

- U petlji ponekad koristimo dvije dodatne naredbe:
 - **break** = odmah prekida izvršavanje petlje
 - **continue** = odmah odlazi na provjeru istinitosti izraza

```
while True:
    ip_adresa = input("Upišite IP adresu za dodati na
whitelist ili samo enter za kraj: ")

    if ip_adresa == "":
        break
    elif ip_adresa.startswith("127"):
        print("Ignoriram adresu", ip_adresa)
        continue
    print("IP adresa", ip_adresa, "je dodana")
print("Kraj")
```

Strana * 23



23

Primjeri

4. Napišite program koji ispisuje brojeve od 200 do 300 odvojene razmacima.
5. Napišite program od korisnika učitava dva broja i ispisuje sve neparne brojeve između manjeg i većeg (ali bez ta dva broja). Brojeve odvojite zarezom i razmakom.
6. Napišite program koji će ispisati tablicu množenja. Svaki zapis stavite u vlastiti red i lijepo formatirajte, primjerice "5 * 5 = 25"

Strana * 24



24

Primjeri

7. *Zamislite ploču za igru veličine 5 x 5 polja:
 - Donje lijevo polje ima koordinate x i y jednake (1, 1).
 - Donje desno polje ima koordinate (5, 1)
 - Gornje lijevo polje ima koordinate (1, 5)
 - Gornje desno polje ima koordinate (5, 5)

Neka se igrač inicijalno nalazi na (1, 1), neprijatelji Nep1 i Nep2 na (2, 4) i (4, 3), a blago na (5, 5).

Napišite program koji će omogućiti korisniku da upravljajući igračem dođe do blaga. Ako padne s ploče ili naleti na neprijatelja, igri je kraj. Igrač se može kretati tipkama 'a' (lijevo), 'w' (gore), 'd' (desno) i 's' (dolje). Nakon svake kretnje iscrtajte ploču.

Koristite: `print("Isti redak", end="")`

Strana * 25



25

Ispis bez prelaska u novi redak

- Kad napišemo

```
print('a')
```

- To u stvari znači:

```
print('a', end='\n')
```

- Parametar end kaže da nakon ispisa želim prijeći u novi red

- Ako ne želimo prijeći u novi red, možemo pisati:

```
print('a', end='')
```

- Primjerice:

```
print("Jedan", end='')
print("Jedan", end='')
print("Jedan", end='\n')
print("Dva")
print("Tri")
```

Strana 26



26

NAREDBA FOR

Strana 27



27

Funkcija range

- Sekvenca je niz vrijednosti
 - Primjer: 1, 2, 2, 3, 4, 5, 6, 7, 8, 10
 - Primjer: 10, 8, 6, 4, 2, 0
 - Primjer: 100, 200, 300, 400, 500
- Sekvencu je najlakše dobiti funkcijom range
 - $\text{range}(5)$ kreira sekvencu brojeva 0, 1, 2, 3 i 4
 - ↑
 - max
 - $\text{range}(3, 10)$ kreira sekvencu 3, 4, 5, 6, 7, 8 i 9
 - ↑ ↑
 - min max
 - $\text{range}(10, 20, 2)$ kreira sekvencu 10, 12, 14, 16, 18
 - ↑ ↑ ↑
 - min max korak

Strana • 28



28

Naredba for

- Naredba **for** također služi za izradu petlji
 - Koristi se kad je broj ponavljanja petlje unaprijed poznat
 - Služi za prolazak po sekvenci
- Primjer:


```
for broj in range(5):
    print(broj)
```

Strana • 29



29

Primjeri

8. Ispišite brojeve:
 - a. Od 1 do 100
 - b. Od 1000 do 900
9. Ispišite brojeve djeljive sa 7 između 1 i 2000. Brojeve odvojite zarezima i razmacima.
10. Napišite program koji od korisnika učitava broj h i koji iscrtava piramidu od znakova '*' visine h . Primjer za $h = 5$:

```

      *
     ***
    *****
   ********
  **********

```

Strana * 30



30

Za sljedeće predavanje

- ✓ Ponoviti sve iz ovog predavanja
- ✓ Pročitati i isprobati primjere:
 - programiz.com/python-programming/operators
 - programiz.com/python-programming/if-elif-else
 - programiz.com/python-programming/for-loop
 - programiz.com/python-programming/while-loop
 - programiz.com/python-programming/break-continue
 - programiz.com/python-programming/pass-statement

Strana * 31



31