

PROGRAMIRANJE 1

VJEŽBA 2: PRINT, INPUT

SADRŽAJ

VARIJABLE I PRIDRUŽIVANJE VRIJEDNOSTI

NAREDBA ZA ISPIS: PRINT

NAREDBA ZA UNOS: INPUT

ZADACI

VARIJABLE I PRIDRUŽIVANJE VRIJEDNOSTI

- **Varijabla** je simboličko ime koje je povezano s određenom vrijednošću i koje je pridruženo određenoj lokaciji memorije.
- Nazivi varijabli mogu se sastojati od **slova, znamenki ili znaka _** te nužno moraju započinjati sa slovom ili znakom _.

Python razlikuje velika i mala slova (eng. **case sensitive**):

`varA` \neq `VarA`

Naziv varijable ne smije sadržavati razmake, niti biti isti kao neka od ključnih (rezerviranih) riječi u Pythonu:

| | | | | |
|---------------------|-----------------------|----------------------|-----------------------|---------------------|
| <code>False</code> | <code>class</code> | <code>finally</code> | <code>is</code> | <code>return</code> |
| <code>None</code> | <code>continue</code> | <code>for</code> | <code>lambda</code> | <code>try</code> |
| <code>True</code> | <code>def</code> | <code>from</code> | <code>nonlocal</code> | <code>while</code> |
| <code>and</code> | <code>del</code> | <code>global</code> | <code>not</code> | <code>with</code> |
| <code>as</code> | <code>elif</code> | <code>if</code> | <code>or</code> | <code>yield</code> |
| <code>assert</code> | <code>else</code> | <code>import</code> | <code>pass</code> | |
| <code>break</code> | <code>except</code> | <code>in</code> | <code>raise</code> | |

VARIJABLE I PRIDRUŽIVANJE VRIJEDNOSTI

- **Pridruživanje vrijednosti** varijabli se obavlja tako da se najprije izračuna vrijednost izraza s **desne strane** znaka pridruživanja i nakon toga se ta vrijednost pridruži varijabli koja se nalazi s **lijeve strane** znaka pridruživanja.

Naziv varijable

← varijabla = izraz

↑

Vrijednost ili izraz koji se prvo računa, a tek zatim pridružuje varijabli.

- **Inicijalizacija varijabli** označava prvo pridruživanje varijable na neku vrijednost.

Npr. kako bi njihovo korištenje bilo pravilno izvršeno, brojčane vrijednosti se često inicijaliziraju na nulu (0), dok se stringovi postavljaju na prazan string ("").

VARIJABLE I PRIDRUŽIVANJE VRIJEDNOSTI

- U Pythonu je omogućeno i **višestruko pridruživanje vrijednosti**:

`var1...varN = izraz1...izrazN`

Primjer:

`a,b=5,10`

`suma,razlika=7+2,7-2`

- Zamjena dviju varijabli u većini programskih jezika moguća je uvođenjem treće varijable, na način:

`t=a`

`a=b`

`b=t`

U Pythonu se zamjena dviju varijabli može izvršiti kao:

`a,b=b,a`

PRINT

- Naredba za **ispis** podataka:

`print()`
Ključna riječ U zagrade navodimo ono što želimo ispisati (brojeve, stringove, aritmetički izraze)



```
>>>print(123)
```

```
123
```

```
>>>print(123,78,-25.56)
```

```
123 78 -25.56
```

```
>>>print("Pozdrav")
```

```
Pozdrav
```

```
>>>print(12*5)
```

```
60
```

```
>>>print("a+b=", 5+6, 11)
```

```
a+b= 11 11
```

1. Brojevi: brojevi se unutar zagrada odvajaju zarezom, dok su u Py. Console odvojeni razmakom

2. Stringovi: sve navedeno unutar navodnika će se točno tako i ispisati

3. Aritmetički izraz: izraz se računa i ispisuje se rezultat

4. Kombinacija gore navedenog

PRINT

- Ukoliko se ne naglasi **separator**, print naredba svaki član ispisuje s razmakom. Ukoliko članove želimo odvojiti drugačije, pišemo:



```
>>>print("a+b=",5+6, sep=', ')\na+b=, 11\n>>>print('Maja', 3, 'Iva', 4, sep='###')\nMaja###3###Iva###4
```

- Drugi način korištenja naredbe print je **formatirani ispis** u kojem se koristi funkcija **format**:

```
print('string1{broj1}...stringN{brojN}'.format(var1...varN))
```

u kojem string1...stringN predstavljaju proizvoljan tekst, var1...varN popis varijabli koje želimo ispisati, a broj1...brojN predstavljaju redne brojeve varijabli unutar popisa.

PRINT

- Ukoliko smo argumente za ispis naveli uzlaznim redom, unutar vitičastih zagrada nije potrebno pisati redne brojeve. Također, argumente je moguće navoditi proizvoljnim redoslijedom.



```
>>>print('a+b={}'.format(5+4))
```

```
a+b=11
```

```
>>>print('{0} ima {1} dana.'.format('Godina',365))
```

```
Godina ima 365 dana.
```

```
>>>print('{0} {2} {1} = {3}'.format(3,4,'+',3+4))
```

```
3 + 4 = 7
```

- Ukoliko želimo pobliže označiti koji tip podatka ispisujemo, unutar zagrada možemo pisati: d – ako ispisujemo cijeli broj (integer), f – ako ispisujemo broj s pomičnom decimalnom točkom (float) i s – ako ispisujemo string.

PRINT



```
>>>print('{2:s} brojeva {0:d} i {1:d} je {3:f}'.format(3,4,'Kolicnik',3/4))  
Kolicnik brojeva 3 i 4 je 0.750000.
```

- Moguće je i dodatno specificirati na koliko će se mjesta ispisati određeni broj:



```
>>>print('{0:5d}'.format(365))  
365
```

```
>>>print('{0:15d}'.format(56789))  
56789
```

```
>>>print('{0:2d}'.format(365))  
365
```

Ispis broja počinje od krajnje desnog mjesta.

Ispis broja počinje od krajnje desnog mjesta.

Ukoliko je traženi broj mjesta manji od broja znamenaka broja, on će se po potrebi proširiti do broja znamenki koji se ispisuje.

PRINT

- U slučaju brojeva s pomičnom decimalnom točkom, moguće je dodatno specificirati broj decimalnih mjesta koja će se ispisivati:



```
>>>print('{0:7.2f}'.format(7/3))  
2.33
```

Za ispis broja koristit će se ukupno 7 mjesta, od kojeg je jedno rezervirano za decimalnu točku, dva za decimalni dio, dok su preostala četiri mjesta rezervirana za cjelobrojni dio broja.

```
>>>print('{0:.4f}'.format(10/3))  
3.3333
```

Moguće je izostaviti specifikaciju ukupnog broja mjesta i navesti samo broj decimalnih mjesta za ispis.

PRINT

- Posebni znakovi koji se koriste u ispisu stringova

`\n` – prijelaz u novi redak (tipke AltGr+Q za obrnutu kosu crtu)

`\t` – tabulator

`\\` – ispis kose crte

`\'` – ispis jednostrukog navodnika

`\"` – ispis dvostrukog navodnika



```
>>> print('prvi redak\n drugi redak\ntreći redak')
```

```
prvi red
```

```
drugi red
```

```
treći red
```

```
>>> print("Pjesma \"Voćka poslije kiše\" je lijepa pjesma.")
```

```
Pjesma "Voćka poslije kiše" je lijepa pjesma.
```

```
>>> print('jedan\tdva\ttri\nčetiri\tpet\tšest')
```

```
jedan dva tri
```

```
četiri pet šest
```

PRINT

- Za olakšan ispis u kojem se string sastoji od više linija, potrebno je početak i kraj stringa staviti unutar **trostrukih** navodnika.



```
>>> print("""prvi redak  
drugi redak  
treći redak""")  
prvi redak  
drugi redak  
treći redak
```

PRINT



Primjer: Ivona ima 50 kuna. Napiši program koji će izračunati koliko Ivona može kupiti čokolade po cijeni od 11 kuna i koliko će joj nakon kupnje ostati novca.

1. Pokrenuti Python
2. Otvoriti novi program i sačuvati ga pod nazivom PrviPrimjer.py
3. Napisati kod

```
print('Maja može kupiti {} čokolade, a ostat će joj {} kuna.'.format  
(50//11,50%11))
```
4. korak: Pokrenuti program i promatrati rezultat u Python Console

- Ukoliko prilikom pisanja koda programa želimo navesti **komentar** koji će Python preskočiti prilikom izvršavanja, takvu liniju koda započinjemo s ljestvama (**#**).

#ovo je komentar u Pythonu

INPUT

- Naredba za **unos** podataka
- U Pythonu unos podataka uključuje pridruživanje vrijednosti na varijablu:

`varijabla = input()`



Varijabla na koju se sprema
korisnikov unos

Unutar zagrada se navodi string
koji se ispisuje u Python Console
prilikom korisnikovog unosa!

- Sve što korisnik unese, Python smatra znakovima (stringom)!
- Ukoliko od strane korisnika očekujemo unos brojanog tipa podatka, korisnikov unos će se prevesti u određeni tip podatka kojeg navedemo prije ključne riječi input.

INPUT

1. Unos brojčanih podataka

```
broj1 = int(input("Unesi cijeli broj:"))
```

Ovakva linija koda izvršava se na način da najprije promatramo unutarnje zagrade pomoću kojih se korisniku u Python Console ispiše “Unesi cijeli broj”. Python očekuje da to bude cijeli broj (int) te ga sprema na varijablu *broj1*. U slučaju greške, ispisuje se poruka.

```
broj2 = float(input("Unesi realni broj:"))
```

```
a = eval(input("Unesi broj: "))
```

Ukoliko želimo da Python samostalno raspozna o kojem brojčanom tipu se radi, koristimo ključnu riječ **eval** (eng. evaluate).

```
x, y = eval(input("Unesi dva broja:"))
```

Ključnu riječ eval koristimo i za **unos više brojeva** u jednoj liniji.

INPUT

2. Unos znakovnih podataka

```
rijec = input("Unesi neku rijec:")
```

Prilikom unošenja niza znakova nije potrebno navoditi tip podataka za unos, jer sve što korisnik unese Python smatra stringom.



Primjer: Maja ima x kuna. Napiši program koji će izračunati koliko Maja može kupiti čokolade po cijeni od y kuna i koliko će joj nakon kupnje ostati novca.

1. Pokrenuti Python IDLE
2. Otvoriti novi program i sačuvati ga pod nazivom DrugiPrimjer.py
3. Napisati kod

#potrebno je na početku programa unijeti budžet x i cijenu y

```
x=int(input("Unesi koliko Maja ima kuna:"))
```

```
y=int(input("Unesi cijenu jedne cokolade:"))
```

```
print('Maja može kupiti {} čokolade, a ostat će joj {}  
kuna.'.format(x//y,x%y))
```

4. korak: Pokrenuti program i promatrati rezultat u Python Console

ZADATAK 2.1.

- Napišite program za igru “Pogađanja zamišljenog broja” između korisnika i računala.

Računalo korisniku daje uputu da zamisli neki broj, pribroji mu vrijednost 2, dobiveni zbroj pomnoži sa 4, a od zadnjeg umnoška oduzme 8.

Korisnik unosi rezultat, a računalo pogađa zamišljeni broj na način da rezultat podijeli sa 4 i ispiše poruku:

“Zamislili ste broj x.”

Primjer:

Za zamišljeni broj 14, račun je $14+2=16*4=64-8=56$ koji korisnik unosi preko Python Console. Računalo računa $56/4=14$ te ispisuje “Zamislili ste broj 14!”.

ZADATAK 2.1.

- Ispis:

```
>>>
Zamislite jedan broj,
pribrojite mu vrijednost 2,
dobiveni zbroj pomnozite sa 4,
a od zadnjeg umnoska oduzmite 8!

Koji ste rezultat dobili? 20

Zamislili ste broj 5!
>>>
```

- Zadatke koje rješavamo testiramo s obzirom na sve mogućnosti koje smo predvidili prilikom programiranja (npr. slučaj kada korisnik ispravno unese broj). Međutim, ukoliko se pojedina linija koda ne izvrši kako je zadana (npr. ukoliko korisnik unese pogrešan tip podatka: string umjesto broja, integer umjesto floata itd.) prilikom izvršavanja programa ispisat će se greška.

ZADATAK 2.2.

- Napišite program za jednostavni kamatni izračun:

Korisnik želi oročiti x HRK na 5 godina uz godišnju kamatnu stopu od 3%. Kolika je konačna vrijednost (C_5) tog iznosa na kraju pete godine, ako je obračun kamata jednostavan, godišnji i dekurzivan?

Koliko iznose ukupne kamate (K_5) na kraju 5. godine?

Konačne iznose je potrebno zaokružiti na 2 decimalna mjesta.

$$C_0 = x$$

$$n=5 \text{ (godina)}$$

$$p=3 \text{ (\%)}$$

$$C_5 = ? \rightarrow C_5 = C_0 * \left(1 + \frac{n*p}{100}\right)$$

$$K_5 = ? \rightarrow K_5 = C_5 - C_0$$

ZADATAK 2.2.

- Ispis:

```
>>>
Koliki iznos novca u kunama zelite orociti? 5000

Vrijeme orocavanja: 5 godina
Kamatna stopa: 3%

Konacna vrijednost iznosa:5750.00 kuna
Ukupne kamate:750.00 kuna
>>> |
```

ZADATAK 2.3.

- Richterova ljestvica (1-10) definirana je brojem kojim se označava seizmička energija proizišla iz potresa. Ljestvica se procjenjuje prema logaritamskom zapisu najveće amplitude zapisane na specifičnom uređaju koji se zove Wood-Andersonov seizmograf.

Napišite program koji će za sljedeće Richterove stupnjeve izračunati odgovarajuću energiju (u džulima, J) i ekvivalentnu količinu eksploziva (u tonama, t): 1 R.s., 9.0 R.s. (Japan, 2011.) i 9.5 R.s. (Čile, 1960.).

Energija = $10^{(1.5 * \text{Richter}) + 4.8}$ (J) 1 tona eksploziva = 4.184×10^9 (J)

Podatke Stupanj/Energija/TNT ispisati u tabličnom obliku (poravnate pomoću tabulatora), a vrijednosti zaokružiti na 4 decimalna mjesta.

Također, pitati korisnika da unese proizvoljan stupanj Richtera te ispisati odgovarajuću energiju i ekvivalentnu količinu eksploziva.

ZADATAK 2.3.

- Ispis:

```
Richter  Joules          TNT
1          1995262.3150      0.0005
9.0        1995262314968882944.0000  476879138.3769
9.5        11220184543019653120.0000  2681688466.3049

Unesi stupanj u Richterima:4
Ekvivalentna energija u Joulesima: 63095734448.019424
Ekvivalentna kolicina eksploziva u tonama: 0.00047687913837688307
>>> |
```

ZADATAK 2.4.

- Napiši program koji će služiti za ispis “Potvrdnice o prosjeku ocjena tijekom studija” koju referada izdaje na zahtjev studenta.

Podaci potrebni za unos i ispis uključuju: matični broj studenta, današnji datum, evidencijski broj potvrdnice, ime i prezime studenta, datum rođenja studenta, mjesto rođenja studenta, vrsta studija, akademska godina završetka studija, prosjek ocjena za svaku od 5 godina studija, svrha izdavanja potvrde te ime i prezime voditelja studentske službe.

Ukupni prosjek ocjena studija ispisati zaokružen na 2 decimalna mjesta.

ZADATAK 2.4.

Sveučilište u Splitu

„Prirodoslovno-matematički fakultet“

Matični broj studenta:.....

Split,.....

Broj:.....

Na temelju čl.17. Zakona o općem upravnom postupku izdaje se

POTVRDNICA

.....rođen-a.....u.....

zavšio/la sveučilišni studij.....u akademskoj godini.....

i ostvario/la je sljedeći prosjek ocjena:.....

Ova potvrdnica izdaje se u svrhu.....

i u druge svrhe se ne može upotrijebiti.

Voditelj studentske službe

.....

ZADATAK 2.4.

- Ispis:

```
>>> Unesite maticni broj studenta/ice:6543444
Unesite danasnji datum:14.10.2015.
Unesite evidencijski broj potvrdnice:54345/4
Unesite ime i prezime studenta/ice:Ana Anic
Unesite datum rođenja studenta:5.3.1992.
Unesite mjesto rođenja studenta:Split
Unesite naziv studija studenta:Matematika i informatika
Unesite akademsku godinu zavrsetka studija:2015.
Unesite ukupni prosjek prve godine studija:3.9
Unesite ukupni prosjek druge godine studija:4.2
Unesite ukupni prosjek trece godine studija:4.2
Unesite ukupni prosjek cetvrte godine studija:4.6
Unesite ukupni prosjek pete godine studija:4.8
Unesite svrhu izdavanja potvrdnice:Natjecaj za posao
Unesite ime i prezime voditelja studentske sluzbe:Ivana Ivanic

Sveuciliste u Splitu
"Prirodoslovno-matematicki fakultet"
Maticni broj studenta: 6543444
Split, 14.10.2015.
Broj: 54345/4
Na temelju cl.17. Zakona o opcem upravnom postupku izdaje se
POTVRDNICA

Ana Anic roden/a 5.3.1992. u Split
zavrshio/la sveucilisni studij Matematika i informatika u akademskoj godini 2015.
i ostvario/la je sljedeci prosjek ocjena:4.34
Ova potvrdnica izdaje se u svrhu: Natjecaj za posao
i u druge svrhe se ne moze upotrijebiti.

Voditelj studentske referade

Ivana Ivanic
>>> |
```

ZADATAK 2.5.

- Napiši program koji će unositi dva prirodna broja a i b te ispisivati osnovne aritmetičke operacije (zbrajanje, oduzimanje, množenje, dijeljenje – drugi broj je uvijek različit od nule). Ispis izraza treba biti u “punom obliku”, npr. za unos brojeva 5 i 7 i operaciju zbrajanja, treba ispisati “ $5 + 7 = 12$ ”.

ZADATAK 2.5.

- Ispis:

```
>>>
Unesi prvi broj:5
Unesi drugi broj:7
5 + 7 = 12
5 - 7 = -2
5 * 7 = 35
5 / 7 = 0.7142857142857143
>>> |
```