

## Laboratorijska vježba 4

### Zadatak 1

Napišite program koji za unesene duljine kateta računa hipotenuzu pravokutnog trokuta. Napomena: korijen se računa uz pomoć funkcije `math.sqrt()` koja se nalazi u biblioteci `math` (na početku programa potrebno je uključiti biblioteku naredbom `import`). Npr.

```
import math
print ('Korijen iz 4 je', math.sqrt(4))
```

### Zadatak 2

Varijablama `a` i `b` pridružiti proizvoljne cjelobrojne vrijednosti. Napisati izraze koji provjeravaju iduće činjenice (izraze koji rezultiraju u `True` ili `False` kada se izvrše u interpreteru). Brojčane pozicije bitova brojimo od najmanje značajnog bita:

- a) je li zadnji bit broja `a` postavljen na 1
- b) je li četvrti bit broja `a` postavljen na 1
- c) jesu li zadnja tri bita varijabli `a` i `b` jednaki
- d) jesu li predzadnja tri bita varijabli `a` i `b` jednaki
- e) jesu li četvrti i šesti bit broja `a` postavljen na 1
- f) jesu li četvrti ili šesti bit broja `a` postavljen na 1
- g) je li samo četvrti ili šesti bit broja `a` postavljen na 1

Predložak:

```
# a
if...
    print...
```

```
#b
...
```

### Zadatak 3

IP adresa je sastavljena od četiri dijela (npr. 196.68.33.1) gdje je svaki dio kodiran na 8 bitova, tako da je IP adresa zapravo 32-bitni broj. Adresa iz prethodnog primjera kodira se brojem 3292799233.

Treba napisati izraz (ili izraze) koji za uneseni cijeli broj između 0 i 2147483647 provjerava:

- a) jesu li prva dva dijela adrese 196 i 68
- b) jesu li zadnja dva dijela adrese manji od 128
- c) jesu li prvi i zadnji dio adrese jednaki.

#### Zadatak 4

Napisati program koji od korisnika prima 10 brojeva. Na kraju program ispisuje koliko je brojeva bilo parno i koliko neparno. Program ignorira unesene nule.

#### Zadatak 5

Napisati program koji prima brojeve dok se ne upiše 0. Ako je uneseni broj negativan, program traži ponovan unos broja dok se ne unese pozitivan broj. Na kraju program ispisuje broj pogrešnih (negativnih) unosa.