

# Izpit

Tekom izpita boste napisali svoj modul. Poimenujte ga **Ime\_Priimek**. Ko končate z izpitom ga pošljite na [gregor.balkovec@ltfe.org](mailto:gregor.balkovec@ltfe.org) in [anze.glusic@ltfe.org](mailto:anze.glusic@ltfe.org).

## Naloga 01

Točka: / 5

Napišite funkcijo `funkcija01`, ki kot parametra prejme dolžino **a** in **b** stranice pravokotnega trikotnika. S pomočjo Pitagorevega pravila naj funkcija izračuna dolžino **c** in to vrednost vrne.

Korenjenje lahko najdete v *math* knjižnici.

INPUT:

```
funkcija01(3, 4)
```

OUTPUT:

5

INPUT:

```
funkcija01(7, 12)
```

OUTPUT:

13.892

In [ ]:

## Naloga 02

Točka: / 5

Napišite funkcijo `funkcija02`, ki kot parameter prejme list besed. Funkcija naj vrne nov list, v katerem so samo besede, ki ne vsebujejo šumnikov (čšž).

INPUT:

```
l = ["avto", "kočija", "banana", "matematika", "šola", "žolna", "čevljar",  
     "zrezek", "pes"]  
funkcija02(l)
```

OUTPUT:

```
['avto', 'banana', 'matematika', 'zrezek', 'pes']
```

In [ ]:

## Naloga 03

Točka: / 5

Napišite funkcijo `funkcija03`, ki kot parameter prejme podatke našega senzorja. Podatki vsebujejo zadnjih 5 meritev za **temperaturo** (TC), **vlažnost** (HUM) in **zračni pritisk** (PRES). Pod ključem *ts*

imamo podatek kdaj je bila vrednost izmerjena (unix timestamp), pod ključem *value* imamo dejansko meritev.

Funkcija naj preveri ali je katerakoli merjena vrednost preseгла prag:

- TC prag je 35
- HUM prag je 90
- PRES prag je 1000

V kolikor je bil prag presežen naj se izpiše opozorilo. Primer:

- Preveri TC. TS: 1673748552, VALUE: 39

Funkcija ne vrne nobene vrednosti.

INPUT:

```
data = {
  "TC": [
    {"ts": 1673748552,
     "value": 39},
    {"ts": 1673648552,
     "value": 36},
    {"ts": 1673548552,
     "value": 28},
    {"ts": 1673448552,
     "value": 23},
    {"ts": 1673348552,
     "value": 13},
  ],
  "HUM": [
    {"ts": 1673748552,
     "value": 12},
    {"ts": 1673648552,
     "value": 56},
    {"ts": 1673548552,
     "value": 86},
    {"ts": 1673448552,
     "value": 74},
    {"ts": 1673348552,
     "value": 23},
  ],
  "PRES": [
    {"ts": 1673748552,
     "value": 969},
    {"ts": 1673648552,
     "value": 957},
    {"ts": 1673548552,
     "value": 998},
    {"ts": 1673448552,
     "value": 1023},
    {"ts": 1673348552,
     "value": 989},
  ],
}
funkcija03(data)
```

OUTPUT:

Preveri TC. TS: 1673748552, VALUE: 39

Preveri PRES. TS: 1673448552, VALUE: 1023

## Naloga 04

Točke: /5

Napišite funkcijo `funkcija04`. Funkcija naj prebere podatke v datoteki `naloga04_input.txt`.

V datoteki imamo shranjene podatke o knjigah, kjer vsaka vrstica predstavlja eno knjigo. Oblika vrstic:

```
<ID>; <naslov knjige>; <avtor>; <leto>
```

Funkcija naj ustvari novo datoteko `naloga04_output.txt` in vanjo zapiše vrstice iz datoteke `naloga04_input.txt` v naraščajočem vrstnem redu, glede na `ID`.

## Naloga 05

Točke: /5

Napišite razred `Gorivo`. Ko ustvarimo novo instanco vanj shranimo `ime` in `euro_na_liter`.

Napišite razred `Crpalka`. Ko ustvarimo novo instanco vanj shranimo seznam `goriva` (list) katere lahko tankamo na tej črpalki.

Temu razredu implementirajte sledeče funkcije:

- `najcenejse_gorivo(self)` --> vrne ime goriva, ki je najceneše glede na `euro_na_liter`
- `najdrazje_gorivo(self)` --> vrne ime goriva, ki je najdražje glede na `euro_na_liter`
- `dodaj_gorivo(self, novo_gorivo)` --> doda novo\_gorivo v seznam, če gorivo s takim imenom še ne obstaja na tej črpalki. Če gorivo s takim imenom že obstaja naj goriva ne doda ampak naj napiše "To gorivo že obstaja".
- `vsa_goriva(self)` --> vrne seznam imen goriv na tej črpalki

```
gorivo1 = Gorivo("bencin-7", 3.864)
```

```
gorivo2 = Gorivo("dizel", 2.999)
```

```
gorivo3 = Gorivo("nafta-3", 4.3333)
```

```
crpalka = Crpalka([gorivo1, gorivo2, gorivo3])
```

```
print("Vsa goriva na črpalki:")
```

```
for g in crpalka.vsa_goriva():
    print("* ", g.ime, g.euro_na_liter)
```

OUTPUT:

```
Vsa goriva na črpalki:
```

```
*   bencin-7 3.864
```

```
*   dizel 2.999
```

```
*   nafta-3 4.3333
```

```
# -----
```

```
print("Najcenejše gorivo: ", crpalka.najcenejse_gorivo())
```

```
print("Najdražje gorivo: ", crpalka.najdrazje_gorivo())
```

OUTPUT:

Najcenejše gorivo: dizel

Najdražje gorivo: nafta-3

# -----

```
gorivo4 = Gorivo("bencin-100",3.101)
```

```
gorivo5 = Gorivo("dizel-100",3.432)
```

```
gorivo6 = Gorivo("dizel", 3.001)
```

```
crpalka.dodaj_gorivo(gorivo4)
```

```
crpalka.dodaj_gorivo(gorivo5)
```

```
crpalka.dodaj_gorivo(gorivo6)
```

OUTPUT:

To gorivo že obstaja

# -----

```
print("Vsa goriva na črpalki:")
```

```
for g in crpalka.vsa_goriva():
```

```
    print("* ", g.ime, g.euro_na_liter)
```

```
print("Najcenejše gorivo: ", crpalka.najcenejse_gorivo())
```

```
print("Najdražje gorivo: ", crpalka.najdrazje_gorivo())
```

In [ ]:

In [ ]: