

1. Migracije

Napiši funkcijo migracije(ime_datoteke), ki prejme ime datoteke, ki vsebuje vrstice oblike

```
8: Maribor -> Ljubljana
3: Maribor -> Nova Gorica
10: Ljubljana -> Maribor
5: Koper -> Nova Gorica
3: Novo mesto -> Nova Gorica
```

Vsaka vrstica pove, koliko ljudi se je preselilo odkod kam. Funkcija vrne par imen: kraj, iz katerega je odšlo največ ljudi in kraj, v katerega se je jih največ priselilo. V gornjem primeru vrne ("Maribor", "Nova Gorica").

2. Zakladi

Imamo razred Robot. Ta se v začetku nahaja na koordinatah (0, 0) in je obrnjen na desno. Koordinatni sistem je takšen kot pri matematiki: koordinata y narašča navzgor. Robot ima naslednje metode.

- naprej(d) gre za d naprej v podani smeri;
- desno() se obrne za 90 stopinj v desno;
- levo() se obrne za 90 stopinj levo;
- koordinate() vrne trenutne koordinate (x in y)

Večini je razred znan iz domače naloge; tokrat je že podan. Napisati pa morate funkcijo zakladi(navodila), ki prejme navodila za robota v obliki niza, na primer "7D3221LL", kar bi pomenilo, da gre za 7 polj naprej, se obrne desno, nato gre za 3 polja naprej, za 2 polji naprej, za 2 polji naprej, za 1 polje naprej, nato se obrne levo in še enkrat levo. Na vseh poljih, za katera velja, da je vsota absolutnih vrednosti njunih koordinat deljiva s 7, je zaklad. (To velja tudi za začetno polje, (0, 0)!).

Funkcija naj vrne, koliko polj z zakladom je robot obiskal. Če večkrat obiše isto polje, dobi zaklad le enkrat.

3. Roboti

Recimo, da imamo tri robote in jim damo navodila "JSVZZVJ". Vsi roboti začnejo na (0, 0) in izmenično jemljejo navodila iz niza. Prvi robot bo šel za eno polje na **J**, drugi na **S**, tretji na **V**, potem spet prvi na **Z**, drugi **Z**, tretji **V**, prvi **J**. Njihove koordinate po tem so $(-1, -2)$, $(-1, 1)$, $(2, 0)$. Prvi, recimo, je šel namreč na J, Z in J, zato $(-1, -2)$.

Napiši funkcijo roboti(navodila, n), ki prejme navodila in število robotov (niso nujno trije!), ter vrne seznam terk njihovih koordinat.

4. Brez ponavljanja

Napiši **rekurzivno** funkcijo brez_ponavljanja(s), ki prejme seznam in vrne nov seznam, v katerem vse zaporedne ponovitve istega elementa zamenja z enim samim elementom. Klic brez_ponavljanja([3, 1, 1, 4, 2]) vrne [3, 1, 4, 2].

Pomoč: Seznam brez ponavljanja dobimo tako, da prvi element obdržimo, ali pa ne. Slediti pa mora seznam brez ponavljanja.

5. Čarovnik

Napiši razred Carovnik, katerega **konstruktor** kot argument dobi niz, ki pove, kaj čarovnik zna in koliko zaračuna za to, na primer alkemist = Carovnik("svinec -> zlato", 10). Niz bo vedno vseboval dve enobesedni stvari, ločeni z ->.

Razred ima metodo caraj(s), ki prejme seznam in vrne nov seznam, v katerem so vsi elementi prečarani v nove. Tako bi klic alkemist(["svinec", "les", "zlato", "svinec"]) vrnil ["zlato", "les", "zlato", "zlato"]. Alkemist bi za to zaračunal 20 evrov. (Poceni!)

Poleg tega ima razred metodo zasluzek(), ki pove, koliko je čarovnik doslej zaslužil. V gornjem primeru bi klic alkemist.zasluzek() torej vrnil 20.