

1. Valovi

V nekem kraju je bilo število okuženih, po dnevih, takšno: $po_dnevi = [1, 5, 6, 0, 0, 0, 2, 3, 0, 5, 8, 0]$. V prvem valu je bilo torej okuženih $1 + 5 + 6 = 12$ ljudi, v drugem $2 + 3 = 5$ in v tretjem 13, skratka $[12, 5, 13]$.

Napiši funkcijo $po_valovih(po_dnevi)$, ki prejme seznam s številom okuženih po dnevih in vrne seznam s številom okuženih po valovih. Valovi so ločeni z enim ali več dnevi brez okužb. Dnevi brez okužb so lahko tudi na začetku ali koncu seznama. Za več primerov glej teste.

2. Sledilnik

Se spomnite zime? Takole je šlo.

- Prvi dan so zaprli gledališča in smučišča.
- Drugi dan so zaprli še šole, frizerje in muzeje, vendar so ob tem odprli smučišča.
- Tretji dan so zaprli knjižnice in odprli muzeje in smučišča (ki pa so bila itak že odprta!).
- Četrty dan so zaprli smučišča (in knjižnice in gledališča, ampak ta so bila zaprta že od prej) in odprli šole.
- Peti dan so odprli šole ter zaprli frizerje in smučišča.
- Šesti dan so zaprli smučišča.
- Sedmi dan so počivali(!).
- Osmi dan so odprli smučišča.

Vse skupaj predstavimo s seznamom parov seznamov. Vsak par ustreza enemu dnevu; prvi element para pove, kaj se je tisti dan zaprlo, drugi pa, kaj se je odprlo. Če je vse skupaj zapleteno: tako pač v resnici je.

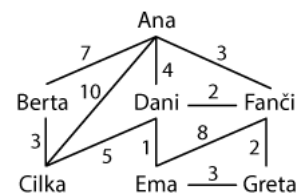
```
dnevi = [
  (["gledališča", "smučišča"], []),
  (["šole", "frizer", "muzeji"], ["smučišča"]),
  (["knjižnice"], ["muzeji", "smučišča"]),
  (["smučišča", "knjižnice", "gledališča"], ["šole"]),
  (["šole"], ["frizer", "smučišča"]),
  (["smučišča"], []),
  ([], []),
  ([], ["smučišča"])]
```

Napiši funkcijo $sledilnik(dnevi)$, ki prejme seznam, kot je gornji, in vrne, kolikšno je največje število hkrati zaprtih stvari. V gornjem primeru vrne 4, ker so bile na, recimo, drugi dan, istočasno zaprte štiri stvari (gledališča, šole, frizerji in muzeji).

3. Tvegani stiki

Vemo, kdo se je na kateri dan družil s kom (glej sliko). Podatki so shranjeni v slovarju (vsako druženje je zabeleženo dvakrat, pri obeh družabnikih):

```
druzenja = {
  "Ana": [("Berta", 7), ("Cilka", 10),
          ("Dani", 4), ("Fanči", 3)],
  "Berta": [("Ana", 7), ("Cilka", 3)],
  ... in tako naprej
```



Če se nekdo druži s kom, ki je okužen, postane sam kužen naslednji dan. (Na isti dan, ko se je okužil, pa še ni nevaren!)

Napiši funkcijo $okuzeni(oseba, cas, druzenja)$, ki prejme ime osebe, dan, na katerega se je okužila, in slovar druženj. Funkcija vrne množico vseh okuženih.

Klic $okuzeni("Ana", 3, druzenja)$ vrne $\{"Ana", "Berta", "Cilka", "Dani"\}$, ker Ana okuži Fanči in Berto, Berta pa še Dani. Fanči ni okužena: Ana na dan 3 še ni bila kužna, Dani pa se je tako ali tako okužila šele na dan 5.

Namig: ne boj se, da se kaj zacikla, saj nihče ni kužen na dan, ko se je okužil, zato časi vedno naraščajo.

OBRNI!

4. PCR

V laboratorij so dobili tri različne viruse, ki pa so vsi predstavniki seva delta. Njihova zaporedja mRNA so:

```
pozitivni = ['GGGUGCCCA', 'CCCAUAGGGU', 'CAGCUCGGUU'].
```

Tole pa so štirje primeri drugih sevov:

```
negativni = ['ACCUCAGGAG', 'UCGACCGAAG', 'GUCACUCGCA', 'CGCUUCCCGC'].
```

V laboratoriju uporabljajo naslednje testne fragmente (koščke) mRNA:

```
fragmenti = ["GGU", "CCC", "ACC", "GGG"]
```

Idealno bi bilo, če bi našli fragment, ki se pojavlja v vseh deltah in nobenem od ostalih sevov. Ker ni nujno, da tak fragment obstaja, pa bomo zadovoljni s takšnim, ki se pojavi v čimveč deltah in čim manj ostalih.

- GGU se pojavi v 3 pozitivnih in 0 negativnih, razlika je 3.
- CCC se pojavi v 2 pozitivnih in 1 negativnem, razlika je 1.
- GGG se pojavi v 2 pozitivnih in 0 negativnih, razlika je 0.
- ACC se pojavi v 0 pozitivnih in 2 negativnih, razlika je -1.

Najboljši je GGU.

Napiši funkcijo `identifikator(pozitivni, negativni, fragmenti)`, ki prejme seznam pozitivnih in negativnih primerov ter seznam fragmentov. Vrniti mora fragment, pri katerem je razlika med številom njegovih pojavitev v pozitivnih in v negativnih primerih virusov čim večja. Če je takšnih fragmentov več, lahko vrne kateregakoli od njih.

5. Izbijanje zob

Sev gama je posebej trdovraten in lotiti se ga je mogoče le s pestmi: izbiti mu je potrebno vse dele mRNA. Recimo, da je njegova mRNA v začetku takšna: "ACCUTCCUUGUACUUTAA". Če mu izbijemo "CUU", ostane še "ACCUTCGUATAA" (iz niza odstranimo obe pojavitvi CUU). Vendar je še vedno živ in brca in grize in davi. Če mu izbijemo A, ostane "CCUTCGUT". In tako naprej. Virus je mrtev šele, ko mu ne ostane nič več.

Napiši razred `Virus` z naslednjimi metodami:

- `konstruktor` sprejme začetno mRNA virusa.
- `izbij(fragment)` iz virusove mRNA odstrani vse pojavitve podanega fragmenta; metoda ne vrne ničesar,
- `ostanek()` vrne trenutno virusovo mRNA,
- `mrtev()` vrne `True`, če je virus mrtev (od njegove mRNA ni ostalo nič več) in `False`, če je živ.

(Vsaka metoda seveda prejme še `self`.)