

1. Statistika

Senzor na kolesarski stezi beleži podatke o številu kolesarjev, ki jo prevozijo. Podatke zbira v Pythonov seznam s 24×60 elementi. Element z indeksom i pove število prehodov v i -ti minuti dneva. Napišite naslednje funkcije:

- `po_urah(a)` prejme opisani seznam in vrne seznam s 24 elementi, ki vsebujejo število prehodov po urah dneva.
- `naj_ura(a)` vrne uro z največ prehodi (npr. 5, če je največ prehodov med peto in šesto uro).
- `brez_klicev(a)` vrne število minut, ko ni bilo kolesarjev (torej število ničelnih elementov podanega seznama).

Za polne točke naj bosta dve od gornjih funkcij napisani v eni vrstici, z uporabo izpeljanega seznama ali generatorja.

2. Brez kolesarjev

Napišite funkcijo `obdobje_brez(a)`, ki prejme takšen argument kot funkcije iz prejšnje naloge, in vrne začetek in konec najdaljšega obdobja brez klicev. Če so vsi elementih na indeksih od, na primer, 150 do (vključno) 180 enaki 0 in je to tudi najdaljše zaporedje ničel, mora funkcija vrniti (150, 180). Da bo reševanje lažje, so v testih tudi trije primeri s tabelami, ki nimajo $24 * 60$ temveč le 13 števil.

3. Obremenitve

Kolesarskih stez s senzorji je v resnici več. Recimo, da imamo tri takšne postaje, imenujmo jih Anina, Bertina, Cilkina. Podatki o številu kolesarjev so zbrani v enem samem seznamu: prvi trije elementi se nanašajo na število prehodov čez Anino, Bertino in Cilkinjo postajo (v tem vrstnem redu) v ničti minuti dneva. Naslednji trije se nanašajo na prehode čez te tri postaje v prvi minuti dneva, naslednji trije na drugom minuto in tako naprej. Celotna dolžina tabele je torej enaka $24 \times 60 \times$ število postaj (ki seveda ni nujno vedno tri!).

Napišite funkcijo `obremenitve(imena, porocila)`, ki prejme imena postaj in število kolesarjev (kot ga opisujemo zgoraj). Vrniti mora ime najbolj obremenjene postaje. Če je teh več, lahko vrne ime poljubne med njimi.

4. Angelčina zlata minuta

Angelca dela doktorat iz kolesarske politike. V okviru doktorata je definirala koncept "zlate minute". Ta se nanaša na obremenjenost kolesarskih stez.

- Vse minute od ničte do (vključno) 59-te so zlate.
- Poleg tega so zlate vse minute, ko je stezo prevozil vsaj en kolesar, vendar pod pogojem, da je zlata tudi minuta ob pol manjšem času od polnoči. Če gre za liho minuto, poskusimo polovico zaokrožiti navzdol in navzgor; da bo 1017 zlata, zadošča, da je zlata bodisi 508 bodisi 509 (ali obe).

Angleco so vprašali, kakšen smisel ima vse to, je odgovorila, da je to doktorat in da tega ne more kar vsak razumeti.

- V primerih iz testov minuta 420 ni zlata: v 420 minuti so kolesarsko stezo prevozili 4 kolesarji, vendar v polovični, 210 minuti ni bilo nikogar.
- Minuta 1017 je zlata. Razpolovimo jo lahko v 508 ali 509; obe imata kolesarje. Če izberemo 508, bi to vodilo v 254 in 127, ko ni kolesarjev. 509 pa lahko razpolovimo v 254 ali 255. Če izberemo 254, bi spet padli v 127. Če pa razpolovimo 509 v 255, lahko nadaljujemo v 127 (ni kolesarjev) ali v 128 (so), tega v 64 (so) in tega v 32 (< 60).

5. Nadzorni sistem

Napišite razred `Postaja`:

- konstruktor prejme `id` postaje (`id` neko celo število),
- `prehod(self, smer)`, pri čemer je smer lahko "+" ali "-", zabeleži, da je kolesar prevozil postajo,
- `prehodov(self)`, ki vrne par s številom prehodov v smer "+" in v smer "-".

Poleg tega napišite razred `NadzorniSistem`, ki kot argument prejme seznam postaj, to je, seznam objektov tipa `Postaja`. `NadzorniSistem` ima primeren konstruktor in metode

- `prehod(self, id, smer)` postaji s podanim `id`-jem (vedno bo šlo za eno od postaj, ki so bile podane konstruktorju) sporoči, da so jo prevozili v podani smeri;
- `prehodov(self, id)` vrne par s številom prehodov prek postaje s podanim indeksom.

Pazi: `NadzorniSistem` naj ne shranjuje podatkov o prehodih. Podatki o prehodih so shranjeni v objektih razreda `Postaja`, `NadzorniSistem` pa shranjuje postaje.