

1. Budilka

Napiši tri **podprograme** (funkcije), ki jih bo sistem klical, da mu bodo pomagali upravljati z budilko. Tvoji podprogrami naj bodo takšne oblike:

- **void NastaviBudilko(int t)** — sistem ga pokliče, ko želi uporabnik nastaviti, naj zvoni vsak dan ob času t . Če je bila budilka pred tem nastavljena na zvonjenje ob drugem času, naj se stari čas pobriše. Čas je podan kot celo število v minutah od polnoči (torej od 0 do $24 \cdot 60 - 1 = 1439$).
- **bool JeCasZaAlarm()** — sistem ga pokliče vsako minuto točno ob začetku minute. Če naj budilka to minuto zvoni, naj ta funkcija vrne **true**, sicer **false**.
- **void Dremez()** — sistem ga pokliče, ko pritisne uporabnik tipko za dremež. Če se to zgodi manj kot pet minut po originalnem zvonjenju, naj tvoj program poskrbi, da bo budilka še enkrat zazvonila pet minut po originalnem zvonjenju. Morebitne nadaljnje klice tega podprograma ignoriraj (prav tako tudi klice, ki nastopijo več kot pet minut po originalnem zvonjenju) — z drugimi besedami, poskrbeti moraš, da bo učinek takih klicev enak, kot če se podprogram takrat sploh ne bi klical. Nova nastavitvev (s podprogramom `NastaviBudilko`) hkrati tudi prekliče trenutno aktivno funkcijo dremeža.

Poleg teh podprogramov lahko tvoja rešitev uporablja tudi globalne spremenljivke, ki jim lahko tudi določiš začetne vrednosti.

Predpostavi, da se bo tvoj program začel izvajati točno opolnoči in da bo takrat sistem najprej poklical tvoj podprogram `JeCasZaAlarm`. Ob začetku izvajanja naj se program obnaša tako, kot da budilka ni nastavljena.

Še deklaracije v pythonu:

```
def NastaviBudilko(t: int) -> None: ...
def JeCasZaAlarm() -> bool: ...
def Dremez() -> None: ...
```

2. Odbojkaške točke

Tvoj prijatelj Jan z obale v svojem prostem času sodi na odbojkaških tekmah. Seveda pa bi on raje igral odbojko in na plaži ribaril, zato bi rad razmišljajoči del sojenja preložil nate. On bo zapisal zaporedje dogodkov, ti pa mu izračunaj, ali to točko dobi ekipa 1 ali 2.

Dogodki, ki jih bo Jan napisal, so treh vrst:

1. 01 oz. 02 pomeni odboj ekipe 1 oz. 2. Na začetku vsake točke mora nekdo iz ene ekipe žogo servirati (ima en odboj) na drugo stran mreže. Od takrat naprej mora vsaka ekipa žogo odbiti vsaj enkrat in največ trikrat ter jo z zadnjim odbojem spraviti nazaj čez mrežo.
2. P1 oz. P2 pomeni, da je žoga padla v polje ekipe 1 oz. 2. Če ekipi žoga pade v lastno polje, točko izgubi, če pa jo spravi v polje druge ekipe, točko dobi.
3. I pomeni, da je žoga šla izven igrišča (v out) ali neveljavno letela pod mrežo. Ekipa, ki se ji to zgodi, izgubi točko.

Napiši program (ali podprogram oz. funkcijo), ki iz seznama dogodkov izračuna, katera ekipa dobi prvo točko. Podrobnosti tega, v kakšni obliki je seznam dogodkov podan, določi sam in jih v svoji rešitvi tudi opiši.

Zaporedje se vedno začne z dogodkom 01 ali 02, kar nam pove, katera ekipa je servirala žogo. Zaporedje se ne konča nujno takoj po tistem, ko ena od ekip dobi točko.

3. Tekstonim

Na telefonski številčni tipkovnici imamo na vsaki tipki poleg številke zapisane še tri ali štiri črke, kar nam včasih olajša zapomniti si kakšno telefonsko številko.

Črkam ustrezajo številke takole:

a, b, c = 2
d, e, f = 3
g, h, i = 4
j, k, l = 5
m, n, o = 6
p, q, r, s = 7
t, u, v = 8
w, x, y, z = 9

(številki 1 in 0 nimata pripadajočih črk)

Za neko podano telefonsko številko (ki bi si jo radi lažje zapomnili) poišči med možnimi besedami iz podanega seznama take, ki (predelane v številke) predstavljajo strnjen podniz podane telefonske številke. Za vsak tak podniz izpiši niz, ki nastane, če v telefonski številki zamenjaš ta podniz z besedo iz seznama.

Primer: če imamo telefonsko številko 0586326 in seznam, v katerem so tudi besede *june*, *junec*, *kunec*, *nebo* in *voda*, bomo našli naslednje možne zapise:

0june26

0junec6

0kunec6

058nebo

05voda6

Napiši program, ki bo z vhodne datoteke (ali standardnega vhoda) prebral eno telefonsko številko, potem pa pregledal vse besede v preostalih vrsticah vhodne datoteke in izpisal najdene zapise.

Če se beseda (predelana v številko) pojavi kot podniz večkrat, izpiši vse tako dobljene zapise.

V prvi vrstici vhodne datoteke je zapisana telefonska številka (zaporedje števk med 0 in 9), v preostalih vrsticah vhoda so besede slovarja, vsaka v svoji vrstici, sestavljajo jih le male črke angleške abecede.

Primer vhoda:

0586326
june
junec
zmagovalec
kunec
nebo
vedro
voda

Pripadajoči izhod:

0june26
0junec6
0kunec6
058nebo
05voda6

4. Premešani mozaik

Umetnik Polde je sestavil mozaik v obliki pravokotne kariraste mreže, široke w stolpcev in visoke h vrstic; v vsaki celici mreže je kvadratna ploščica v eni od B možnih barv, zato lahko barve predstavimo kar s števili od 1 do B . Na Poldetovem mozaiku je bila j -ta ploščica v i -ti vrstici barve b_{ij} .

Žal so v noči pred otvoritvijo Poldetove razstave nepridipravi vdrli v galerijo in premešali ploščice njegovega mozaika; zdaj je j -ta ploščica v i -ti vrstici barve a_{ij} . Poudarimo, da gre še vedno za iste ploščice, le drugače so razporejene po pravokotni mreži (torej se število ploščic posamezne barve ni spremenilo).

Polde se je odločil, da bo povrnil mozaik v prvotno stanje, vendar bo iz tega naredil performans. Mozaik bo spreminjal po korakih, pri čemer v vsakem koraku zamenja med seboj dve sosednji ploščici (to sta taki, ki imata skupno eno od stranic).

Opiši postopek (ali napiši program ali podprogram oz. funkcijo, če ti je lažje), ki kot vhodne podatke dobi trenutno stanje mreže (vse a_{ij}) in njeno prvotno stanje (vse b_{ij}) ter izpiše poljubno zaporedje takšnih zamenjav, s katerim lahko Polde povrne mozaik v prvotno stanje. Podrobnosti izpisa niso pomembne, da bo le razvidno, kateri dve ploščici naj na posameznem koraku zamenja.

5. Rokomet

Spremljamo rokometno tekmo med dvema moštvo, A in B. Ker je moštvo A precej boljše od B, o rezultatih poročajo na nenavaden način. Namesto da bi poročali o vsakem голу posebej, poročajo samo o golih moštva B, hkrati pa povejo, koliko golov je vmes doseglo moštvo A.

Primer poročanja: „Ekipa B je končno dosegla gol, vmes je ekipa A dala 5 golov. Ponovno gol za B, vmes 2 gola za A. Od zadnjega gola za B do konca tekme je padlo še 7 golov za A.“

Napiši program ali podprogram oz. funkcijo, ki izpiše končni rezultat tekme in še to, kolikokrat se je vmes zamenjalo vodstvo.

Predpostaviš lahko, da so podatki podani kot seznam števil, npr. $[5, 2, 7]$ (kar ustreza gornjemu primeru). Vsa števila razen zadnjega predstavljajo en gol moštva B, velikost števila pa pove, koliko golov je vmes dalo moštvo A. Zadnje število predstavlja število golov od zadnjega gola B do konca tekme.

Za primer $[5, 2, 7]$ so posamezni rezultati (A : B) naslednji: $0:0$, $1:0$, $2:0$, ..., $5:0$, $5:1$, $6:1$, $7:1$, $7:2$, $8:2$, ..., $13:2$, $14:2$. Končni rezultat je 14 proti 2, vodstvo pa se ni nikoli zamenjalo.

Še en primer: pri seznamu $[2, 0, 0, 0, 1, 3]$ se je rezultat spreminjal takole: $0:0$, $1:0$, $2:0$, $2:1$, $2:2$, $2:3$, $2:4$, $3:4$, $3:5$, $4:5$, $5:5$, $6:5$. Končni rezultat je 6 proti 5, vodstvo pa se je zamenjalo dvakrat: najprej je nekaj časa vodilo moštvo A, nato nekaj časa B in nato spet A.

Opomba: če nekaj časa vodi eno moštvo, pa potem drugo izenači in nato prvo spet povede, se to ne šteje za spremembo vodstva.