



## Kalėjimo iššūkis

Kalėjime yra 500 kalinių. Vieną dieną kalėjimo prižiūrėtojas jiems pasiūlė šansą atgauti laisvę. Jis kambaryje padės du krepšius - krepšį A ir krepšį B. Kiekviename krepšyje bus nuo 1 iki  $N$  monetų imtinai. Krepšiuose bus **skirtingi** skaičiai monetų. Prižiūrėtojas kaliniams metė iššūkį. Kaliniams reikia išsiaiškinti, kuriame krepšyje yra mažiau monetų.

Kambaryje be krepšių su pinigais yra ir rašymo lenta. Bet kuriuo metu ant lentos turi būti parašytas lygiai vienas skaičius. Iš pradžių ant lentos užrašytas skaičius 0.

Tuomet prižiūrėtojas kalinius po vieną įleidžia į kambarį. Kiekvienas kalinys įeidamas nežino, kurie arba kiek kalinių jau buvo įėję į kambarį. Kai kalinys įeina į kambarį, jis perskaito skaičių, tuo metu užrašytą ant lentos. Perskaitęs skaičių jis turi pasirinkti krepšį A arba krepšį B. Tuomet jis **patikrina** pasirinktą krepšį ir taip sužino, kiek jo viduje yra monetų. Tuomet kalinys atlieka vieną iš dviejų **veiksmų**:

- Nutrinti dabar parašytą skaičių ir vietoj jo užrašyti savo pasirinktą neneigiamą sveikąjį skaičių bei palikti kambarį. Atkreipkite dėmesį, kad jis gali užrašyti tokį pat skaičių koks buvo prieš jam atėjus. Po to iššūkis tęsiasi (nebent visi 500 kalinių jau aplankė kambarį).
- Nurodyti krepšį, kuriame yra mažiau monetų. Tokiu atveju iššūkis iš karto baigiasi.

Prižiūrėtojas niekada neįleis į kambarį kalinio, kuris jau buvo jį aplankęs.

Kaliniai įveikia iššūkį, jei vienas iš jų teisingai nurodo, kuriame krepšyje yra mažiau monetų. Jie neįveikia iššūkio, jei vienas iš jų neteisingai nurodo krepšį su mažiau monetų arba visi 500 iš jų aplanko kambarį ir nei vienas nenurodo, kuriame krepšyje mažiau monetų.

Prieš iššūkiui prasidedant kaliniai susirenka kalėjimo salėje ir sukuria bendrą trijų žingsnių **strategiją** iššūkiui įveikti.

- Jie pasirenka neneigiamą sveikąjį skaičių  $x$ , kuris bus didžiausias skaičius, kurį jie galimai norės užrašyti ant lentos
- Bet kokiam ant lentos užrašytam skaičiui  $i$  ( $0 \leq i \leq x$ ), jie nusprendžia, kurį krepšį turėtų patikrinti kalinys, kuris įėjęs į kambarį pamato skaičių  $i$  ant lentos.
- Jie nusprendžia, kokį veiksma turi atlikti kalinys, suskaičiavęs, kiek pasirinktame krepšyje yra monetų. Konkrečiai, bet kokiam ant lentos užrašytam skaičiui  $i$  ( $0 \leq i \leq x$ ) ir bet kokiam patikrintame krepšyje pamatytų monetų skaičiui  $j$  ( $1 \leq j \leq N$ ), jie nusprendžia arba
  - kokį skaičių nuo 0 iki  $x$  (imtinai) užrašyti ant lentos arba
  - kurį krepšį nurodyti kaip tą, kuriame yra mažiau monetų.

Kaliniams įveikus iššūkį prižiūrėtojas paleis kalinius jiems atsėdėjus dar  $x$  dienų.

Sukurkite strategiją, užtikrinančią, kad kaliniai įveiks iššūkį nesvarbu kiek krepšiuose A ir B bus monetų. Skiriami taškai priklauso nuo  $x$  reikšmės (detaliau aprašyta skyrelyje *Dalinės užduotys*).

## Realizacija

Parašykite tokią funkciją:

```
int[][] devise_strategy(int N)
```

- $N$ : maksimalus galimas monetų skaičius bet kuriame viename krepšyje.
- Ši funkcija turi grąžinti masyvą  $s$ , sudarytą iš ilgio  $N + 1$  sveikųjų skaičių masyvų, aprašantį jūsų strategiją.  $x$  reikšmė yra lygi masyvo  $s$  ilgiui minus vienas. Kiekvienam  $i$  ( $0 \leq i \leq x$ ), masyvas  $s[i]$  nurodo, ką kalinys darys, jei atėjęs į kambarį ant lentos perskaitys skaičių  $i$ :
  1.  $s[i][0]$  reikšmė lygi 0, jei kalinys tikrins krepšį A, arba 1, jei kalinys tikrins krepšį B.
  2. Tegu  $j$  yra pasirinktame krepšyje esančių monetų skaičius. Tuomet kalinys atliks tokį veiksmą:
    - Jei  $s[i][j]$  reikšmė lygi  $-1$ , kalinys nurodys, kad krepšyje A yra mažiau monetų nei krepšyje B.
    - Jei  $s[i][j]$  reikšmė lygi  $-2$ , kalinys nurodys, kad krepšyje B yra mažiau monetų nei krepšyje A.
    - Jei  $s[i][j]$  reikšmė yra neneigiamas skaičius, kalinys parašys tą skaičių ant lentos. Atkreipkite dėmesį, kad  $s[i][j]$  turi būti ne didesnis nei  $x$ .
- Ši funkcija bus kviečiama lygiai vieną kartą.

## Pavyzdys

Panagrinėkime tokį iškvietimą:

```
devise_strategy(3)
```

Tegu  $v$  yra skaičius, kurį kalinys pamato įėjęs į kambarį. Viena iš galimų strategijų yra tokia:

- Jei  $v = 0$  (įskaitant iš pradžių ant lentos parašytą skaičių), patikrinti krepšį A.
  - Jei jame yra 1 moneta, nurodyti, kad krepšyje A yra mažiau monetų.
  - Jei jame yra 3 monetas, nurodyti, kad krepšyje B yra mažiau monetų.
  - Jei jame yra 2 monetas, ant lentos parašyti skaičių 1 (ir nutrinti skaičių 0).
- Jei  $v = 1$ , patikrinti krepšį B.
  - Jei jame yra 1 moneta, nurodyti, kad krepšyje B yra mažiau monetų.
  - Jei jame yra 3 monetas, nurodyti, kad krepšyje A yra mažiau monetų.
  - Jei jame yra 2 monetas, ant lentos užrašyti skaičių 0 (ir nutrinti skaičių 1). Atkreipkite dėmesį, kad taip niekada nebus, nes tam abiejuose krepšiuose turėtų būti po 2 monetas, o tai nėra leidžiama.

Kad pateiktą šią strategiją kaip sprendinį, funkcija turėtų grąžinti  $[[0, -1, 1, -2], [1, -2, 0, -1]]$ . Grąžinamo masyvo ilgis lygus 2, todėl šiai grąžinamai reikšmei  $x$  yra lygus  $2 - 1 = 1$ .

## Ribojimai

- $2 \leq N \leq 5000$

## Dalinės užduotys

1. (5 taškai)  $N \leq 500$ ,  $x$  reikšmė turi neviršyti 500.
2. (5 taškai)  $N \leq 500$ ,  $x$  reikšmė turi neviršyti 70.
3. (90 taškų)  $x$  reikšmė turi neviršyti 60.

Bet kuriai dalinei užduočiai, jei `devise_strategy` grąžinama reikšmė neaprašo teisingos strategijos, sprendimas vertinamas 0 taškų.

Už 3-iąją dalinę užduotį galite gauti dalinius taškus. Tegu  $m$  yra didžiausia  $x$  reikšmė tarp visų testų šioje dalinėje užduotyje. Jūsų gaunami taškai skaičiuojami pagal šią lentelę:

Sąlyga	Taškai
$40 \leq m \leq 60$	20
$26 \leq m \leq 39$	$25 + 1.5 \times (40 - m)$
$m = 25$	50
$m = 24$	55
$m = 23$	62
$m = 22$	70
$m = 21$	80
$m \leq 20$	90

## Pavyzdinė vertinimo programa

Pavyzdinė vertinimo programa skaito duomenis šiuo formatu:

- 1-oji eilutė:  $N$
- $(2 + k)$ -oji ( $0 \leq k$ ) eilutė:  $A[k] B[k]$
- paskutinė eilutė:  $-1$

Kiekviena eilutė, išskyrus pirmą ir paskutinę, atitinka vieną scenarijų.  $(2 + k)$ -ojoje eilutėje aprašytą scenarijų vadinsime  $k$ -uoju scenarijumi.  $k$ -ajame scenarijuje krepšyje A yra  $A[k]$  monetų ir krepšyje B yra  $B[k]$  monetų.

Pavyzdinė vertinimo programa iš pradžių iškviečia `devise_strategy(N)`.  $x$  reikšmė yra lygi gražinamo masyvo ilgiui minus vienas. Tuomet, jei pavyzdinė vertinimo programa nustato, kad `devise_strategy` neatitinka skyrelyje *Realizacija* aprašytų sąlygų, ji išveda vieną iš šių klaidų pranešimų ir baigia darbą:

- `s` is an empty array:  $s$  yra tuščias masyvas (neatitinkantis teisingos strategijos).
- `s[i]` contains incorrect length: Egzistuoja indeksas  $i$  ( $0 \leq i \leq x$ ), toks, kad `s[i]` ilgis nėra lygus  $N + 1$ .
- First element of `s[i]` is non-binary: Egzistuoja indeksas  $i$  ( $0 \leq i \leq x$ ), toks, kad `s[i][0]` nėra lygus nei 0, nei 1.
- `s[i][j]` contains incorrect value: Egzistuoja indeksai  $i, j$  ( $0 \leq i \leq x, 1 \leq j \leq N$ ), tokie, kad `s[i][j]` nepriklauso intervalui nuo  $-2$  iki  $x$  imtinai.

Priešingu atveju pavyzdinė vertinimo programa pateikia du rezultatus.

Iš pradžių ji išveda jūsų strategijos gaunamus rezultatus tokiu formatu:

- $(1 + k)$ -oji ( $0 \leq k$ ) eilutė: jūsų strategijos gaunamas rezultatas  $k$ -ojo scenarijaus atveju. Jei naudojant šią strategiją kuris nors kalinys nurodo, kad mažiau monetų yra krepšyje A, vertinimo programa išveda raidę A. Jei naudojant šią strategiją kuris nors kalinys nurodo, kad mažiau monetų yra krepšyje B, vertinimo programa išveda raidę B. Jei naudojant šią strategiją joks kalinys nenurodo, kuriame krepšyje yra mažiau monetų, vertinimo programa išveda raidę X

Po to pavyzdinė vertinimo programa į failą `log.txt` rašo tokiu formatu:

- $(1 + k)$ -oji ( $0 \leq k$ ) eilutė: `w[k][0] w[k][1] ...`

$(1 + k)$ -ojoje eilutėje įrašoma seka atitinka  $k$ -ąjį scenarijų ir nurodo skaičius, kurie buvo užrašyti ant lentos. Konkrečiai, `w[k][l]` yra  $(l + 1)$ -ojo kalinio, įėjusio į kambarį, užrašytas skaičius.