



# Mechaninė lėlė

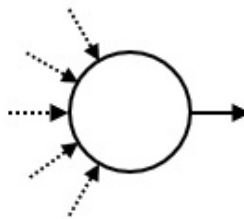
Mechaninė lėlė yra tokia lėlė, kuri automatiškai kartoja tam tikrą judesių seką. Nuo senų laikų Japonijoje buvo kuriamos tokios lėlės.

Mechaninės lėlės judesius valdo vamzdeliais sujungtų **prietaisų grandinė**. Kiekvienas prietaisas turi neribotą kiekį (galimai ir nulį) **įėjimų** ir vieną arba du **išėjimus**. Kiekvienas vamzdelis jungia vieną prietaiso išėjimą su to paties arba kito prietaiso įėjimu. Prie kiekvieno įėjimo yra prijungtas lygiai vienas vamzdelis, taip pat prie kiekvieno išėjimo yra prijungtas lygiai vienas vamzdelis.

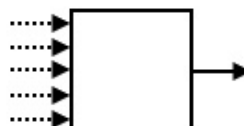
Aprašysime kaip lėlė atlieka savo judesius. Įsivaizduokite jog viename iš prietaisų padėtas **rutulys**. Šis rutulys be sustojimo keliauja grandine: jis išrieda iš prietaiso pro vieną jo išėjimą, rieda prie šio išvedimo prijungtu vamzdeliu ir įrieda į prietaisą esančiame kitame šio vamzdelio gale.

Egzistuoja trijų tipų prietaisai: **pradinis**, **trigeriai** ir **skirstytuvai**. Grandinėje yra lygiai vienas pradinis prietaisas,  $M$  trigerių ir  $S$  skirstytuvų ( $S$  gali būti lygus nuliui). Nuspręskite, kiek grandinėje turi būti skirstytuvų, t.y. kokia turi būti  $S$  reikšmė. Kiekvienas prietaisas turi unikalų serijos numerį.

Pradiniu momentu rutulys padėtas pradiniame prietaise. Prietaisas turi vieną išėjimą, jo serijinis numeris yra 0.

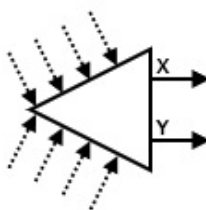


Kai rutulys patenka į trigerį, lėlė atlieka tam tikrą judesį. Kiekvienas trigeris turi vieną išėjimą. Trigerių serijiniai numeriai yra nuo 1 iki  $M$ .



Kiekvienas skirstytuvas turi du išėjimus, vadinamus 'X' ir 'Y'. Skirstytuvo **būsena** yra 'X' arba 'Y'. Rutuliui patekus į skirstytuvą, jis iš jo išrieda pro išėjimą, kuris sutampa su tuometine skirstytuvo būsena. Rutuliui išriedėjus skirstytuvą pakeičia savo būseną į

priešingą. Pradiniu momentu visų skirstytuvų būsenos yra 'X'. Skirstytuvų serijiniai numeriai yra nuo  $-1$  iki  $-S$ .



Duotas trigerių kiekis  $M$ . Taip pat duota  $N$  ilgio seka  $A$ , kurios kiekvienas narys yra trigerio serijinis numeris. Kiekvienas trigeris sekoje  $A$  gali būti sutinkamas nulį, vieną, arba daugiau kartų.

Sukurkite grandinę, tenkinančią šias sąlygas:

- Rutulys riedėdamas grandine būtinai sugrįžta į pradžią.
- Kai rutulys pirmą kartą grįžta į pradžią, visų skirstytuvų būklė turi būti 'X'.
- Rutulys pirmą kartą sugrįžta į pradžią prariedėjęs trigerius lygiai  $N$  kartų ir prariedėjęs juos tokia tvarka:  $A_0, A_1, \dots, A_{N-1}$ .
- Iki rutulys pirmą kartą grįžta į pradžią, skirstytuvai pakeičia būseną  $P$  kartų.  $P$  turi neviršyti 20 000 000.

Taip pat nenorite naudoti per daug skirstytuvų.

## Realizacija

Parašykite šią procedūrą:

```
create_circuit(int M, int[] A)
```

- $M$ : trigerių skaičius.
- $A$ : ilgio  $N$  masyvas, nurodantis seką trigerių serijos numerių, per kuriuos rutulys turi prariedėti.
- Ši procedūra kviečiama lygiai vieną kartą.
- Atkreipkite dėmesį, kad  $N$  yra masyvo  $A$  ilgis, ir jį galima rasti taip, kaip nurodyta pastabose apie realizaciją.

Norint pateikti atsakymą, jūsų programa turėtų iškviešti šią procedūrą:

```
answer(int[] C, int[] X, int[] Y)
```

- $C$ : ilgio  $M + 1$  masyvas.  $i$ -ojo ( $0 \leq i \leq M$ ) prietaiso išėjimas yra sujungtas su prietaisu  $C[i]$ .
- $X, Y$ : Vienodo ilgio masyvai. Jų ilgis  $S$  lygus skirstytuvų kiekiui. Skirstytuvo  $-j$  ( $1 \leq j \leq S$ ), išėjimas 'X' yra sujungtas su prietaisu  $X[j - 1]$  ir jo išėjimas 'Y' yra

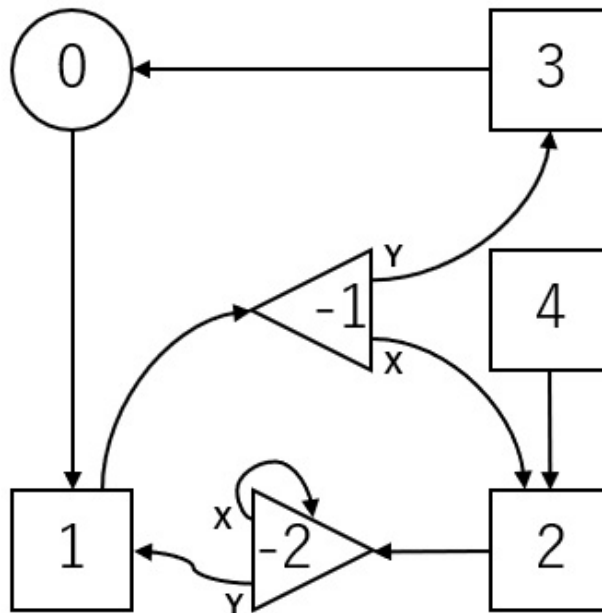
sujungtas su prietaisu  $Y[j - 1]$ .

- Kiekvienas  $C$ ,  $X$  arba  $Y$  elementas turi būti sveikasis skaičius nuo  $-S$  iki  $M$  imtinai.
- $S$  negali viršyti 400 000.
- Ši procedūra turi būti iškviečiama lygiai vieną kartą.
- Grandinė, kurią apibūdina masyvai  $C$ ,  $X$  ir  $Y$ , turi tenkinti uždavinyje nurodytas sąlygas.

Jei kuri nors iš aukščiau išvardytų sąlygų netenkinama, programa bus įvertinta **Wrong Answer**. Priešingu atveju, programa bus įvertinta **Accepted** ir skiriami taškai priklausys nuo  $S$  (žr. Dalinės užduotys).

## Pavyzdys

Tegu  $M = 4$ ,  $N = 4$  ir  $A = [1, 2, 1, 3]$ . Vertintojas iškviečia `create_circuit(4, [1, 2, 1, 3])`.



Viršuje pateiktoje schemoje parodyta grandinė, kuri aprašyta iškvietime `answer([1, -1, -2, 0, 2], [2, -2], [3, 1])`. Schemoje esantys skaičiai yra prietaisų serijiniai numeriai.

Panaudoti du skirstytuvai, todėl  $S = 2$ .

Pradžioje skirstytuvų  $-1$  ir  $-2$  būklės yra 'X'.

Rutulys rieda taip:

$0 \rightarrow 1 \rightarrow -1 \xrightarrow{X} 2 \rightarrow -2 \xrightarrow{X} -2 \xrightarrow{Y} 1 \rightarrow -1 \xrightarrow{Y} 3 \rightarrow 0$

- Rutuliui pirma patekus į skirstytuvą  $-1$ , jo būklė yra 'X', todėl rutulys toliau keliauja į trigerį 2. Tada  $-1$  skirstytuvo būklė pasikeičia į 'Y'.
- Rutuliui patekus į  $-1$  skirstytuvą antrąjį kartą, jo būklė yra 'Y', todėl rutulys toliau

keliauja į triggerį 3. Tada  $-1$  skirstytuvo būklė pasikeičia į 'X'.

Rutulys pirmą kartą grįžta į pradžią prariedėjęs triggerius 1, 2, 1, 3. Abiejų skirstytuvų,  $-1$  ir  $-2$ , būklės yra 'X'.  $P$  lygus 4. Taigi, ši grandinė tenkina užduoties sąlygas.

Šį pavyzdį atitinka failas `sample-01-in.txt` pateiktas pridėtame zip formato archyve. Tame pačiame archyve pateikti ir kiti pavyzdiniai testai.

## Ribojimai

- $1 \leq M \leq 100\,000$
- $1 \leq N \leq 200\,000$
- $1 \leq A_k \leq M$  ( $0 \leq k \leq N - 1$ )

## Dalinės užduotys

Toliau pateikti kiekvieno testo taškai bei papildomi ribojimai:

1. (2 taškai) Visi sveikieji skaičiai  $i$  ( $1 \leq i \leq M$ ) pasirodo sekoje  $A_0, A_1, \dots, A_{N-1}$  daugiausiai vieną kartą.
2. (4 taškai) Visi sveikieji skaičiai  $i$  ( $1 \leq i \leq M$ ) pasirodo sekoje  $A_0, A_1, \dots, A_{N-1}$  daugiausiai du kartus.
3. (10 taškų) Visi sveikieji skaičiai  $i$  ( $1 \leq i \leq M$ ) pasirodo sekoje  $A_0, A_1, \dots, A_{N-1}$  daugiausiai 4 kartus.
4. (10 taškų)  $N = 16$
5. (18 taškų)  $M = 1$
6. (56 taškai) Be papildomų ribojimų

Taškai už kiekvieną testą, jei programa įvertinama kaip **Accepted**, yra skaičiuojami pagal  $S$  reikšmę:

- Jeigu  $S \leq N + \log_2 N$ , už testą gaunate visus taškus.
- Kiekvienam testui 5-oje ir 6-oje dalinėse užduotyse, jeigu  $N + \log_2 N < S \leq 2N$ , gaunate dalį taškų, apskaičiuojamų padauginus dalinei užduočiai numatytų taškų kiekį iš  $0.5 + 0.4 \times \left( \frac{2N - S}{N - \log_2 N} \right)^2$ .
- Priešingu atveju, surenkama 0 taškų.

Atkreipkite dėmesį, kad už dalinę užduotį gaunamų taškų skaičius yra lygus minimaliam taškų skaičiui, gautam už tos dalinės užduoties testus.

## Pavyzdinė vertinimo programa

Pavyzdinė vertinimo programa skaito duomenis iš standartinės įvesties tokiu formatu:

- 1-a eilutė:  $M N$

- 2-a eilutė:  $A_0 A_1 \dots A_{N-1}$

Pavyzdinė vertinimo programa pateikia tris išvestis.

Pirmiausia, vertintojas išveda jūsų atsakymą į failą pavadinimu `out.txt` tokiu formatu:

- 1-oji eilutė:  $S$
- $(2 + i)$ -oji eilutė ( $0 \leq i \leq M$ ):  $C[i]$
- $(2 + M + j)$ -oji eilutė ( $1 \leq j \leq S$ ):  $X[j - 1] Y[j - 1]$

Toliau vertintojas simuliuoja rutulio judėjimą. Programa eilės tvarka išveda rutulio aplankytų prietaisų serijinius numerius į failą pavadinimu `log.txt`.

Trečia, vertintojas atspausdina jūsų atsakymo įvertinimą į standartinį išvesties įrenginį.

- Jei jūsų programa įvertinama **Accepted**, pavyzdinis vertintojas išveda  $S$  ir  $P$  formatu: `Accepted: S P`.
- Jei jūsų programa įvertinama **Wrong Answer**, vertintojas išveda `Wrong Answer: MSG`. Galimos MSG reikšmės yra:
  - `answered not exactly once`: Procedūra `answer` iškviesta ne lygiai vieną kartą.
  - `wrong array length`: Masyvo `C` ilgis nelygus  $M + 1$ , arba nesutampa `X` ir `Y` ilgiai.
  - `over 400000 switches`:  $S$  viršija 400 000.
  - `wrong serial number`: Kuris nors iš `C`, `X`, arba `Y` elementų yra mažesnis už  $-S$  arba didesnis už  $M$ .
  - `over 20000000 inversions`: Rutulys nesugrįžo į pradžią pakeitęs skirstytuvų būklės 20 000 000 kartų.
  - `state 'Y'`: Egzistuoja skirstytuvus kurio būklė yra 'Y' kai rutulys pirmą kartą grįžta į pradžią.
  - `wrong motion`: Trigeriai neatitinka nurodytų sekoje  $A$ .

Atkreipkite dėmesį jog pavyzdinis vertintojas gali nesukurti `out.txt` ir/arba `log.txt` failų jei programa įvertinama `Wrong Answer`.