



# Mechanical Doll

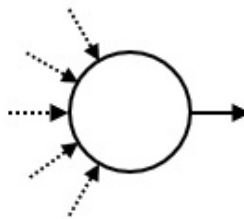
Een mechanische pop is een pop die automatisch een specifieke serie bewegingen herhaald. In Japan zijn er al eeuwenlang allerlei mechanische poppen gemaakt.

De bewegingen van een mechanische pop worden gestuurd door een **circuit** dat is opgebouwd uit **apparaten**. De apparaten zijn verbonden met buizen. Elk apparaat heeft een of twee **uitgangen** en kan willekeurig veel (mogelijk nul) **ingangen**. Elke buis verbindt een uitgang van een apparaat met een ingang van hetzelfde of een ander apparaat. Precies één buis is verbonden met elke ingang en precies één buis is verbonden met elke uitgang.

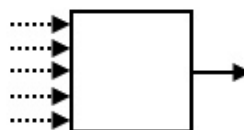
Om te beschrijven hoe de pop beweegt ga je uit van een **bal** die wordt geplaatst in één van de apparaten. De bal reist door het circuit. Bij elke stap van de reis verlaat de bal het apparaat door één van zijn uitgangen, en reist door de verbonden buis naar het apparaat aan de andere kant van deze buis.

Er zijn drie types apparaten: **oorsprong**, **trigger** en **schakelaar**. Er is precies één oorsprong, er zijn  $M$  triggers en  $S$  schakelaars ( $S$  kan nul zijn). Jij mag zelf een waarde van  $S$  kiezen. Ieder apparaat heeft een uniek serienummer.

De oorsprong is het apparaat waar de bal oorspronkelijk wordt geplaatst. De oorsprong heeft precies één uitgang. Het serienummer van de oorsprong is 0.

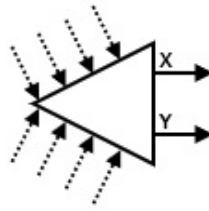


Een trigger zorgt ervoor dat de pop een bepaalde beweging maakt als de bal er binnenkomt. Elke trigger heeft één uitgang. De serienummers van de triggers lopen van 1 tot en met  $M$ .



Elke schakelaar heeft twee uitgangen, aangeduid met 'X' en 'Y'. De **toestand** van een schakelaar is 'X' of 'Y'. Als er een bal in een schakelaar komt, verlaat hij de schakelaar

weer door de uitgang die hoort bij de toestand van de schakelaar. Daarna verandert de toestand van de schakelaar. Aan het begin is de toestand van iedere schakelaar 'X'. De serienummers van de schakelaars lopen van  $-1$  tot en met  $-S$ .



Jij krijgt het aantal triggers  $M$  gegeven. Je krijgt ook een rij  $A$  van lengte  $N$ , waarvan elk element het serienummer is van een trigger. Elke trigger kan een aantal keren (dit kan ook nul zijn) voorkomen in de rij  $A$ . Jij moet een circuit ontwerpen dat aan de volgende voorwaarden voldoet:

- De bal keert na een aantal stappen weer om de oorsprong terug.
- Als de bal weer voor het eerst bij de oorsprong terugkomt is de toestand van iedere schakelaar 'X',
- De bal keert voor het eerst bij de oorsprong weer na precies  $N$  triggers. De serienummers van deze triggers, in de volgorde dat ze bezocht worden, zijn  $A_0, A_1, \dots, A_{N-1}$ .
- $P$  is het aantal, door de bal veroorzaakte toestandsveranderingen van alle schakelaars voordat de bal weer voor het eerst in de oorsprong terugkomt.  $P$  is niet groter dan 20 000 000.

Je wil niet te veel schakelaars gebruiken.

## Implementatiedetails

Implementeer de volgende procedure.

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

- $M$ : het aantal triggers.
- $A$ : een array met lengte  $N$ , waarin de serienummers worden gegeven van de triggers die de bal moet binnengaan, in de volgorde waarin ze bezocht moeten worden.
- Deze procedure wordt precies één keer aangeroepen.
- Let op: de waarde van  $N$  is de lengte van het array  $A$ , en die kun je verkrijgen zoals staat aangegeven in de implementatieopmerkingen.

Jouw programma moet de volgende procedure aanroepen om antwoord te geven.

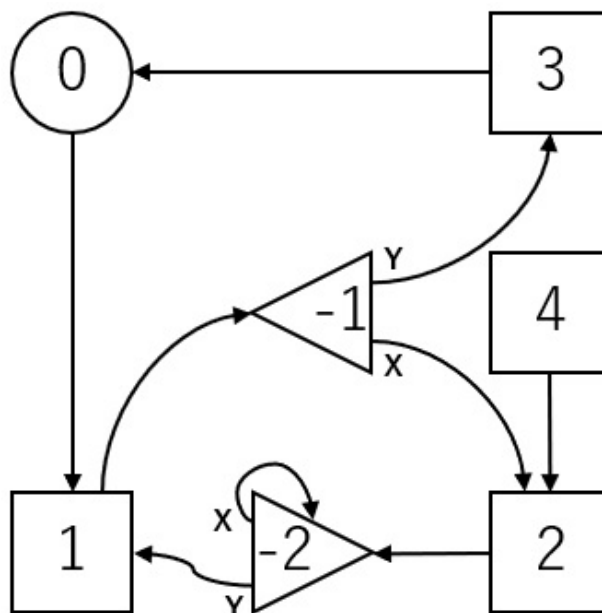
```
answer(int[] C, int[] X, int[] Y)
```

- C: een array van lengte  $M + 1$ . De uitgang van apparaat  $i$  ( $0 \leq i \leq M$ ) is verbonden met apparaat  $C[i]$ .
- X, Y: arrays van dezelfde lengte. De lengte  $S$  van deze arrays is het aantal schakelaars. Voor schakelaar  $-j$  ( $1 \leq j \leq S$ ), is uitgang 'X' is verbonden met apparaat  $X[j - 1]$  en uitgang 'Y' verbonden met apparaat  $Y[j - 1]$ .
- Elk element van C, X, and Y moet een geheel getal zijn, van  $-S$  tot en met  $M$ .
- $S$  mag hoogstens 400 000 zijn.
- Deze procedure moet precies één keer worden aangeroepen.
- Het circuit dat wordt gerepresenteerd door C, X, en Y moet voldoen aan de voorwaarden in de probleembeschrijving..

Als niet aan alle voorwaarden hierboven wordt voldaan, wordt je programm beoordeeld als **Wrong Answer**. Anders wordt je programma beoordeeld als **Accepted** en wordt je score bepaald op basis van  $S$  (zie Subtasks).

## Voorbeeld

Stel  $M = 4$ ,  $N = 4$ , en  $A = [1, 2, 1, 3]$ . De grader roept `create_circuit(4, [1, 2, 1, 3])` aan.



De figuur hierboven laat een circuit zien dat is beschreven door een aanroep `answer([1, -1, -2, 0, 2], [2, -2], [3, 1])`. De getallen in de figuur zijn de serienummers van de apparaten.

Er worden twee schakelaars gebruikt, dus geldt  $S = 2$ .

In het begin staan schakelaars  $-1$  en  $-2$  beide in toestand 'X'.

De bal reist als volgt:

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

- Als de bal de eerste keer schakelaar  $-1$  ingaat, is de toestand ervan 'X'. Daarom reist de bal naar trigger 2. De toestand van schakelaar  $-1$  wordt gewijzigd in 'Y'.
- Als de bal de schakelaar  $-1$  voor de tweede keer ingaat is de toestand ervan 'Y'. Daarom reist de bal naar trigger 3. De toestand van schakelaar  $-1$  wordt gewijzigd in 'X'.

De bal komt voor de eerste keer terug in de oorsprong, nadat de triggers 1, 2, 1, 3 zijn bezocht. De toestand van de schakelaars  $-1$  en  $-2$  is allebei 'X'. De waarde van  $P$  is 4. Daarom voldoet dit circuit aan de voorwaarden.

Het bestand `sample-01-in.txt` in de gezipte bijlage komt overeen met dit voorbeeld. Er zijn ook andere voorbeelden van invoer in de bijlage.

## Randvoorwaarden

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

## Subtaken

De score en de randvoorwaarden voor ieder testgeval zijn als volgt:

1. (2 punten) Voor elke  $i$  ( $1 \leq i \leq M$ ), komt getal  $i$  hoogstens eenmaal voor in de rij  $A_0, A_1, \dots, A_{N-1}$ .
2. (4 punten) Voor elke  $i$  ( $1 \leq i \leq M$ ), komt getal  $i$  hoogstens tweemaal voor in de rij  $A_0, A_1, \dots, A_{N-1}$ .
3. (10 punten) Voor elke  $i$  ( $1 \leq i \leq M$ ), komt getal  $i$  hoogstens 4 keer voor in de rij  $A_0, A_1, \dots, A_{N-1}$ .
4. (10 punten)  $N = 16$
5. (18 punten)  $M = 1$
6. (56 punten) Geen aanvullende voorwaarden

Voor elk testgeval waarbij je programma wordt beoordeeld als **Accepted**, wordt je score berekend op basis van de waarde van  $S$ :

- Als  $S \leq N + \log_2 N$ , krijg je de volledige score voor het testgeval.
- Voor elke test in Subtasks 5 en 6 geldt dat als  $N + \log_2 N < S \leq 2N$  je een gedeeltelijke score verdient. De score voor het testgeval is  $0.5 + 0.4 \times \left( \frac{2N - S}{N - \log_2 N} \right)^2$ , vermenigvuldigd met de score die bij de subtask hoort.
- Anders is de score 0.

Let op: je score voor iedere subtask is het minimum van de scores die je programma haalt voor de testgevallen van die subtask.

## Voorbeeldgrader

De voorbeeldgrader leest invoer van standard input in dit format.

- regel 1:  $M N$
- regel 2:  $A_0 A_1 \dots A_{N-1}$

De voorbeeldgrader produceert drie soorten uitvoer.

Allereerst schrijft de voorbeeldgrader jouw antwoord weg naar een bestand `out.txt` met het volgende format.

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

Ten tweede simuleert de voorbeeldgrader de bewegingen van de bal. Hij voert de opeenvolgende serienummers van de apparaten in een bestand `log.txt`.

Tenslotte geeft de voorbeeldgrader de evaluatie van jouw antwoord weer op standaard uitvoer.

- Wordt je programma beoordeeld als **Accepted**, dan voert de voorbeeldgrader  $S$  en  $P$  uit in het volgende format: `Accepted: S P`.
- Wordt je programma beoordeeld als **Wrong Answer**, dan is de uitvoer `Wrong Answer: MSG`. `MSG` kan het volgende betekenen:
  - `answered not exactly once`: De procedure `answer` is niet precies eenmaal aangeroepen.
  - `wrong array length`: De lengte van  $C$  is niet gelijk aan  $M + 1$ , of de lengtes van  $X$  en  $Y$  verschillen.
  - `over 400000 switches`:  $S$  is groter dan 400 000.
  - `wrong serial number`: Er is een element van  $C$ ,  $X$ , of  $Y$  dat kleiner is dan  $-S$  of groter dan  $M$ .
  - `over 20000000 inversions`: De bal keert niet in de oorsprong terug voordat er 20 000 000 keer een schakelaar van toestand is gewijzigd.
  - `state 'Y'`: Er staat minstens een schakelaar in toestand 'Y' als de bal voor het eerst in de oorsprong terugkeert.
  - `wrong motion`: De triggers die voor beweging zorgen komen niet overeen met de rij  $A$ .

Let op: het kan zijn dat de voorbeeldgrader geen bestanden `out.txt` en/of `log.txt` maakt als je programma beoordeeld wordt als **Wrong Answer**.

