



## Digitaal Circuit

Er is een circuit dat bestaat uit  $N + M$  poorten genummerd van 0 tot en met  $N + M - 1$ . Poorten 0 tot en met  $N - 1$  zijn **drempelpoorten** terwijl poorten  $N$  tot en met  $N + M - 1$  **bronpoorten** zijn.

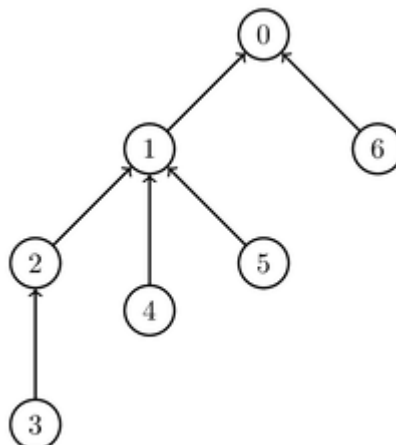
Elke poort, behalve poort 0 is een **input** voor precies één drempelpoort. Om precies te zijn, voor elke  $i$  met  $1 \leq i \leq N + M - 1$  is poort  $i$  een input voor poort  $P[i]$  waar  $0 \leq P[i] \leq N - 1$ . Belangrijk: we hebben ook  $P[i] < i$ . Bovendien nemen we aan  $P[0] = -1$ . Elke drempelpoort heeft één of meer inputs. Bronpoorten hebben geen inputs.

Elke poort heeft een **toestand** die 0 of 1 is. De initiële toestanden van de bronpoorten worden gegeven door een array  $A$  van  $M$  integers. Dat wil zeggen, voor elke  $j$  zodat  $0 \leq j \leq M - 1$  is  $A[j]$  de initiële toestand van bronpoort  $N + j$ .

De toestand van elke drempelpoort hangt af van de toestanden van zijn inputs en wordt op de volgende manier bepaald. Eerst krijgt elke drempelpoort een drempel **parameter** aangewezen. De parameter van een drempelpoort met  $c$  inputs moet een integer tussen 1 en  $c$  (inclusief) zijn. Vervolgens heeft een drempelpoort met parameter  $p$  de toestand 1, als tenminste  $p$  van zijn inputs 1 zijn, anders 0.

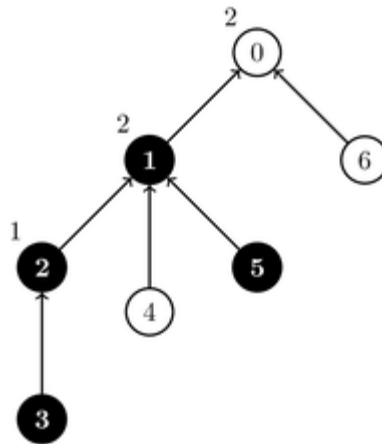
Bijvoorbeeld, stel dat er  $N = 3$  drempelpoorten en  $M = 4$  bronpoorten zijn. De inputs voor poort 0 zijn poorten 1 en 6, de inputs voor poort 1 zijn poorten 2, 4 en 5 en de enige input voor poort 2 is poort 3.

Het voorbeeld wordt afgebeeld in het volgende plaatje.



Stel dat bronpoorten 3 en 5 toestand 1 hebben terwijl bronpoorten 4 en 6 toestand 0 hebben. Neem aan dat we parameters 1, 2 en 2 toewijzen aan drempelpoorten 2, 1 en 0 respectievelijk.

In dit geval heeft poort 2 toestand 1, heeft poort 1 toestand 1 en heeft poort 0 toestand 0. De toewijzing van parameters en toestanden is afgebeeld in de volgende afbeelding. Poorten waarvan de toestand 1 is, staan gemarkeerd in het zwart.



De toestanden van de bronpoorten zullen  $Q$  updates ondergaan. Elke update wordt omschreven door twee integers  $L$  en  $R$  ( $N \leq L \leq R \leq N + M - 1$ ) en schakelt de toestanden van alle bronpoorten tussen  $L$  en  $R$  (inclusief). Dat wil zeggen, voor elke  $i$  zodat  $L \leq i \leq R$  verandert bronpoort  $i$  zijn toestand naar 1 als zijn toestand 0 is, of naar 0 als zijn toestand 1 is. De nieuwe toestand van elke geschakelde poort blijft onveranderd tot die mogelijk veranderd wordt door een van de latere updates.

Jouw doel is om te tellen, na elke update, hoeveel verschillende toewijzingen van parameters aan drempelpoorten er voor zorgen dat poort 0 toestand 1 heeft. Twee toewijzingen worden als verschillend behandeld als er tenminste één drempelpoort is met een andere parameter in beide toewijzingen. Omdat het aantal toewijzingen groot kan zijn moet je het berekenen modulo 1 000 002 022.

Merk op dat in het voorbeeld boven 6 verschillende toewijzingen van parameters aan drempelpoorten zijn, aangezien poorten 0, 1 en 2 respectievelijk 2, 3 en 1 inputs hebben. In 2 van deze 6 toewijzingen heeft poort 0 toestand 1.

## Implementatiedetails

Jouw taak is om twee functies te implementeren.

```
void init(int N, int M, int[] P, int[] A)
```

- $N$ : het aantal drempelpoorten.

- $M$ : het aantal bronpoorten.
- $P$ : een array van lengte  $N + M$  die de inputs voor de drempelpoorten beschrijft.
- $A$ : een array van lengte  $M$  die de initiële toestanden van de bronpoorten beschrijft.
- Deze functie wordt precies één keer aangeroepen, voor enige aanroep van `count_ways`.

```
int count_ways(int L, int R)
```

- $L, R$ : de grenzen van de subreeks van de bronpoorten wiens toestanden geschakeld worden.
- Deze functie moet eerst de gegeven update uitvoeren en dan het aantal manieren geven, modulo 1 000 002 022, waarop de drempelpoorten parameters aangewezen kunnen krijgen die er voor zorgt dat poort 0 toestand 1 heeft.
- Deze functie wordt precies  $Q$  keer aangeroepen.

## Voorbeeld

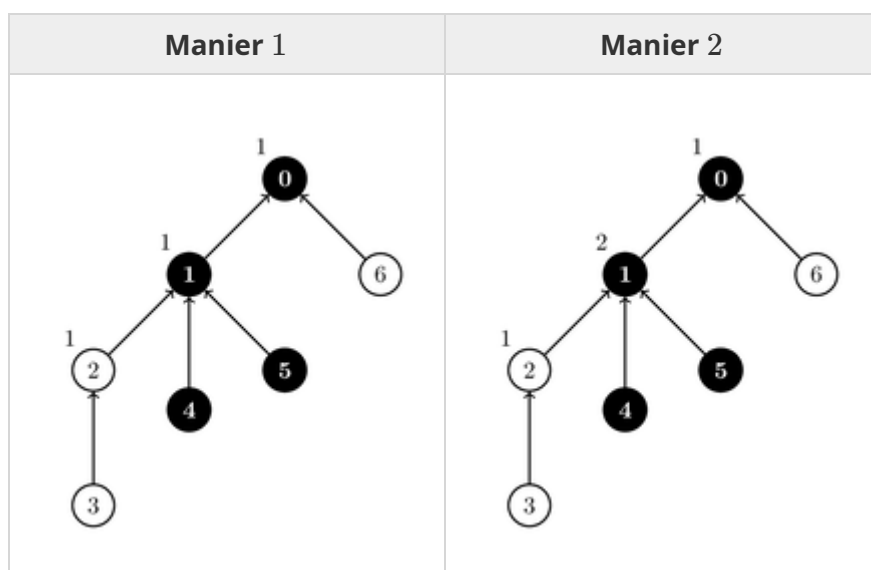
Beschouw de volgende reeks aanroepen.

```
init(3, 4, [-1, 0, 1, 2, 1, 1, 0], [1, 0, 1, 0])
```

Dit voorbeeld staat afgebeeld in de beschrijving boven.

```
count_ways(3, 4)
```

Dit schakelt de toestanden van poorten 3 en 4, dat wil zeggen dat poort 3 de toestand 0 krijgt en 4 krijgt toestand 1. Twee manieren van toewijzen van de parameters die er voor zorgen dat poort 0 toestand 1 heeft zijn afgebeeld in de volgende afbeeldingen.



In alle andere toewijzingen van parameters heeft poort 0 toestand 0. De functie moet dus 2 teruggeven.

```
count_ways(4, 5)
```

Dit schakelt de toestanden van poorten 4 en 5. Als gevolg hebben alle bronpoorten toestand 0 en voor elke toewijzing van parameters heeft poort 0 toestand 0. De functie moet dus 0 teruggeven.

```
count_ways(3, 6)
```

Dit verandert de toestand van alle bronpoorten naar 1. Als gevolg heeft poort 0 toestand 1 voor elke toewijzing van parameters. De functie moet dus 6 teruggeven.

## Randvoorwaarden

- $1 \leq N, M \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $P[0] = -1$
- $0 \leq P[i] < i$  en  $P[i] \leq N - 1$  (voor elke  $i$  met  $1 \leq i \leq N + M - 1$ )
- Elke drempelpoort heeft tenminste één input (voor elke  $i$  ( $0 \leq i \leq N - 1$ ) is er een index  $x$  zodat  $i < x \leq N + M - 1$  en  $P[x] = i$ ).
- $0 \leq A[j] \leq 1$  (voor elke  $j$  met  $0 \leq j \leq M - 1$ )
- $N \leq L \leq R \leq N + M - 1$

## Deelopgaven

1. (2 punten)  $N = 1, M \leq 1000, Q \leq 5$
2. (7 punten)  $N, M \leq 1000, Q \leq 5$ , elke drempelpoort heeft precies twee inputs.
3. (9 punten)  $N, M \leq 1000, Q \leq 5$
4. (4 punten)  $M = N + 1, M = 2^z$  (voor een positieve integer  $z$ ),  $P[i] = \lfloor \frac{i-1}{2} \rfloor$  (voor elke  $i$  met  $1 \leq i \leq N + M - 1$ ),  $L = R$
5. (12 punten)  $M = N + 1, M = 2^z$  (voor een positieve integer  $z$ ),  $P[i] = \lfloor \frac{i-1}{2} \rfloor$  (voor elke  $i$  met  $1 \leq i \leq N + M - 1$ )
6. (27 punten) Elke drempelpoort heeft precies twee inputs.
7. (28 punten)  $N, M \leq 5000$
8. (11 punten) Geen aanvullende randvoorwaarden.

## Voorbeeldgrader

De voorbeeldgrader leest de invoer in het volgende formaat:

- regel 1:  $N M Q$
- regel 2:  $P[0] P[1] \dots P[N + M - 1]$

- regel 3:  $A[0] A[1] \dots A[M - 1]$
- regel  $4 + k$  ( $0 \leq k \leq Q - 1$ ):  $L R$  voor update  $k$

De voorbeeldgrader print jouw antwoorden in het volgende formaat:

- regel  $1 + k$  ( $0 \leq k \leq Q - 1$ ): de teruggegeven waarde van `count_ways` voor update  $k$