# Circuito Digital

Existe um circuito que consiste em $N + M$ **portas** numeradas de $0$ a $N + M - 1$. As portas $0$ a $N - 1$ são **portas de limiar**, enquanto as portas $N$ a $N + M - 1$ são **portas de origem**.

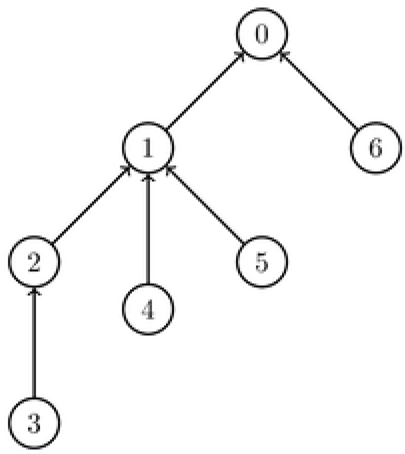
Cada porta, exceto a porta $0$, é um **input** para exatamente uma porta de limiar. Especificamente, para cada $i$ tal que $1 \le i \le N + M - 1$, a porta $i$ é uma entrada para a porta $P[i]$, onde $0 \le P[i] \le N - 1$. Especialmente, também temos que $P[i] < i$. Adicionalmente, assumimos que $P[0] = -1$. Cada porta de limiar tem um ou mais inputs. Portas de origem não têm qualquer input.

Cada porta tem um **estado** que é $0$ ou $1$. Os estados iniciais das portas de entradas são dados por um array $A$ com $M$ inteiros. Isto é, para cada $j$ tal que $0 \le j \le M - 1$, o estado inicial da porta de entrada $N + j$ é $A[j]$.

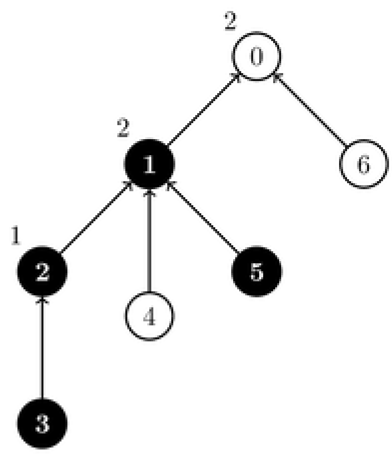
O estado de cada porta de limiar depende do estado dos seus inputs e é determinado da seguinte forma. Primeiro, a cada porta de limiar é atribuído um **parâmetro** de limite. O parâmetro atribuído à porta de limiar com $c$ inputs deve ser um inteiro entre $1$ e $c$ (inclusive). A seguir, o estado da porta de limiar com um parâmetro $p$ é $1$, se pelo menos $p$ dos seus inputs têm estado $1$, ou $0$ caso contrário.

Por exemplo, supõe que existem $N = 3$ portas de limiar e $M = 4$ portas de entrada. Os inputs para a porta $0$ são as entradas $1$ e $6$, os inputs para a porta $1$ são as portas $2$, $4$, e $5$, e o único input para a porta $2$ é a porta $3$.

Este exemplo é ilustrado na figura seguinte.



Supõe que a porta de entrada $3$ e $5$ tem o estado $1$, enquanto as portas de entrada $4$ e $6$ têm o estado $0$. Assume que atribuímos os parâmetros $1$, $2$ e $2$ às portas de limiar $2$, $1$ e $0$, respetivamente. Neste caso, a porta $2$ tem o estado $1$, a porta $1$ tem o estado $1$ e a porta $0$ tem o estado $0$. Esta atribuição de valores de parâmetros e de estados é ilustrada na figura seguinte. Portas cujo estado é $1$ estão destacadas a preto.




Os estados das portas de entrada irão sofrer $Q$ mudanças. Cada mudança é descrita por dois inteiros $L$ e $R$ ($N \leq L \leq R \leq N + M - 1$) e troca os estados de todas as portas de entradas numeradas entre $L$ e $R$, inclusive. Isto é, para cada $i$ tal que $L \leq i \leq R$, o estado da porta de entrada $i$ muda para $1$, se o estado era $0$, ou para $0$, se o estado era $1$. O novo estado de cada porta modificada mantém-se inalterado até que seja possivelmente trocado por uma das mudanças futuras.

O teu objetivo é contar, após cada mudança, quantas atribuições distintas de parâmetros das portas de limiar que resultam que a porta $0$ tenha o estado $1$. Duas atribuições são consideradas diferentes se existe pelo menos uma porta de limiar que tem um valor diferente do seu parâmetro em ambas as atribuições. Como o número de maneiras pode ser grande, deves calculá-lo módulo $1\;000\;002\;022$.

Nota que no exemplo acima, existem $6$ diferentes atribuições de parâmetros das portas de limiar, já que as portas $0$, $1$ e $2$ têm $2$, $3$ e $1$ inputs, respetivamente. Em $2$ destas $6$ atribuições a porta $0$ tem o estado $1$.

## Detalhes de Implementação

A tua tarefa é implementar duas funções.

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

- $N$: o número de portas de limiar.

- $M$: o número de portas de entrada.
- $P$: um array de tamanho $N + M$ que descreve os inputs das portas de limiar.
- $A$: um array de tamanho $M$ que descreve os estados iniciais das portas de entrada.
- Este procedimento é chamado exatamente um vez, antes de qualquer chamada a `count_ways`.

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

- $L$, $R$: os limites do intervalo de portas de entrada, cujo estado é mudado.
- Esta função deverá primeiro aplicar a mudança especificada e depois devolver o número de maneiras, módulo $1\ 000\ 002\ 022$, das atribuições de parâmetros das portas de limiar, que resultam que a porta $0$ tenha o estado $1$.
- Esta função é chamada exatamente $Q$ vezes.

## Exemplo

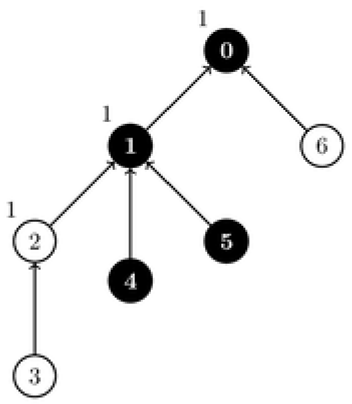
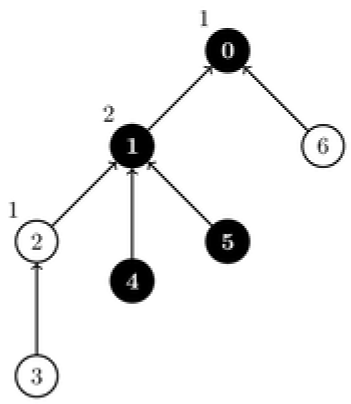
Considera a seguinte sequência de chamadas.

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

Este exemplo é ilustrada na descrição da tarefa acima.

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

Isto troca os estados das portas $3$ e $4$, isto é, o estado da porta $3$ torna-se $0$ e o estado da porta $4$ torna-se $1$. Duas formas de atribuir os parâmetros que resultam que a porta $0$ tenha estado $1$ são os ilustrados nas figuras abaixo.

| **Forma** $1$ | **Forma** $2$ |
| --- | --- |
|  |  |

Em todas as outras atribuições de parâmetros, a porta $0$ tem estado $0$. Portanto a função deverá devolver $2$.

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

Isto troca o estado das portas $4$ e $5$. Como resultado, todas as portas de entrada têm estado $0$, e para qualquer atribuição de parâmetros, a porta $0$ tem estado $0$. Portanto a função deverá devolver $0$.

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

Isto altera todos os estados das portas de entrada para $1$. Como resultado, qualquer atribuição de parâmetros, a porta $0$ tem o estado $1$. Portanto a função deverá devolver $6$.

## Restrições

- $1 \le N, M \le 100\,000$
- $1 \le Q \le 100\,000$
- $P[0] = -1$
- $0 \le P[i] < i$ e $P[i] \le N-1$ (para cada $i$ tal que $1 \le i \le N+M-1$)
- Cada porta de limiar tem pelo menos um input (para cada $i$ tal que $0 \le i \le N-1$ existe um índice $x$ tal que $i < x \le N+M-1$ e $P[x] = i$).
- $0 \le A[j] \le 1$ (para cada $j$ tal que $0 \le j \le M-1$)
- $N \le L \le R \le N+M-1$

## Subtarefas

1. (2 pontos) $N = 1$, $M \le 1000$, $Q \le 5$
2. (7 pontos) $N, M \le 1000$, $Q \le 5$, cada porta de limiar tem exatamente duas entradas.
3. (9 pontos) $N, M \le 1000$, $Q \le 5$
4. (4 pontos) $M = N+1$, $M = 2^z$ (para um inteiro positivo $z$), $P[i] = \lfloor \frac{i-1}{2} \rfloor$ (para cada $i$ tal que $1 \le i \le N+M-1$), $L = R$
5. (12 pontos) $M = N+1$, $M = 2^z$ (para um inteiro positivo $z$), $P[i] = \lfloor \frac{i-1}{2} \rfloor$ (para cada $i$ tal que $1 \le i \le N+M-1$)
6. (27 pontos) Cada porta de limiar tem exatamente duas entradas.
7. (28 pontos) $N, M \le 5000$
8. (11 pontos) Sem restrições adicionais.

## Avaliador Exemplo

O avaliador exemplo lê o input no seguinte formato:

- linha 1: $N\ M\ Q$

- linha $2$: $P[0] \; P[1] \; \ldots \; P[N+M-1]$
- linha $3$: $A[0] \; A[1] \; \ldots \; A[M-1]$
- linha $4+k$ ($0 \le k \le Q-1$): $L \; R$ para a mudança $k$

O avaliador exemplo devolve as respostas no seguinte formato:

- linha $1+k$ ($0 \le k \le Q-1$): o valor devolvido por `count_ways` para a mudança $k$