電子回路

$N+M$ 個の**ゲート**からなる回路があり，各ゲートには 0 から $N+M-1$ までの番号がついている．ゲート 0 からゲート $N-1$ までは**閾値付きゲート**であり，ゲート N からゲート $N+M-1$ は**ソースゲート**である．

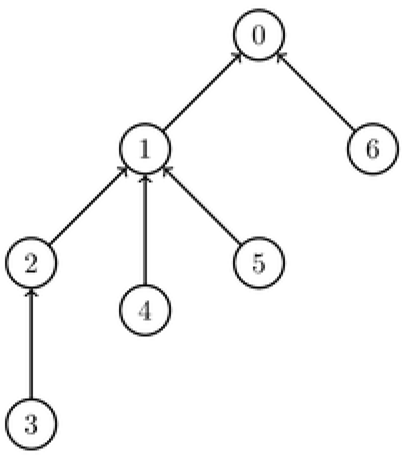
ゲート 0 以外のそれぞれのゲートは，ちょうど 1 つの閾値付きゲートの**入力**となっている．具体的には，$1 \leqq i \leqq N+M-1$ を満たすそれぞれの i について，$0 \leqq P[i] \leqq N-1$ を満たす $P[i]$ があり，ゲート i がゲート $P[i]$ の入力となっている．さらに，$P[i]$ は $P[i] < i$ を満たしている．また，$P[0] = -1$ とする．それぞれの閾値付きゲートは，少なくとも 1 つのゲートを入力にもつ．どのソースゲートも入力をもたない．

各ゲートには 0 と 1 のいずれかで表される**状態**が定まっている．ソースゲートの初期状態は，M 個の整数からなる数列 A によって表される．$0 \leqq j \leqq M-1$ を満たすそれぞれの j について，ソースゲート $N+j$ の初期状態は $A[j]$ である．

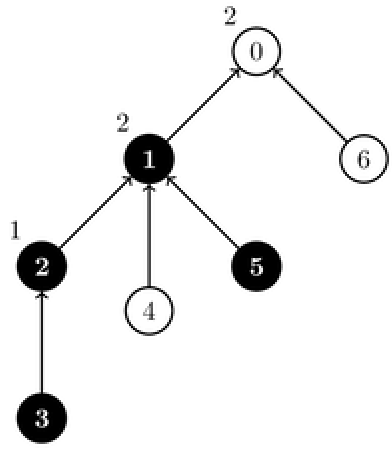
それぞれの閾値付きゲートの状態は，そのゲートの入力の状態に応じて，次のように定まる．まず，それぞれの閾値付きゲートには**パラメータ**が定まっている．c 個の入力をもつ閾値付きゲートのパラメータは 1 以上 c 以下の整数である．パラメータが p の閾値付きゲートの状態は，状態が 1 のゲートからの入力を p 個以上もつとき 1 となり，そうでない場合 0 となる．

例えば，$N=3$ 個の閾値付きゲートがあり，$M=4$ 個のソースゲートがある場合を考える．ゲート 0 はゲート 1 とゲート 6 を，ゲート 1 はゲート $2, 4, 5$ を，ゲート 2 はゲート 3 を入力とする．

この例は，次の図に示されている．



ゲート $3, 5$ の状態が 1 であり，ゲート $4, 6$ の状態が 0 であるとし，ゲート $2, 1, 0$ に割り当てられたパラメータが，それぞれ $1, 2, 2$ であるとする．この場合，ゲート 2 の状態は 1 に，ゲート 1 の状態は 1 に，ゲート 0 の状態は 0 になる．このように初期状態とパラメータを定めた場合の例は，次の図に示されている．状態が 1 のゲートは黒で塗られている．



ソースゲートの状態は Q 回更新される． 各更新は $N \leqq L \leqq R \leqq N + M - 1$ を満たす 2 つの整数 L, R を用いて表され，その更新では L 以上 R 以下の番号のソースゲートの状態が切り替わる．つまり，$L \leqq i \leqq R$ を満たすそれぞれの i について，ソースゲート i の状態は，元の状態が 0 ならば 1 に，元の状態が 1 ならば 0 に変わる．切り替わったゲートの状態は，その後の更新で再び切り替わるまで，変化することはない．

あなたの目標は，ゲート 0 の状態が 1 となるような閾値ゲートのパラメータの定め方が何通りあるかを，各更新の後に求めることである．2 つのパラメータの定め方は，ある閾値付きゲートのパラメータが異なる場合に，異なるものとする．求める値は大きくなりうるので，$1\,000\,002\,022$ で割ったあまりを求めることとする．

先程の例においては，ゲート $0, 1, 2$ への入力の個数がそれぞれ $2, 3, 1$ のため，パラメータの定め方は 6 通りある．そのうち，2 通りの定め方において，ゲート 0 の状態は 1 になる．

実装の詳細

あなたは，次の関数を実装する必要がある．

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

- N: 閾値付きゲートの個数
- M: ソースゲートの個数
- P: 閾値付きゲートの入力を表す長さ $N + M$ の配列
- A: ソースゲートの初期状態を表す長さ M の配列
- この関数は， `count_ways` のどの呼び出しよりも先に，ちょうど 1 回呼び出される．

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

- L, R: 状態が変更されるソースゲートの範囲の境界値
- この関数は，更新後のソースゲートの状態において，ゲート 0 の状態が 1 になるような閾値ゲートのパラメータの定め方の個数を $1\,000\,002\,022$ で割ったあまりを返さなければならない．
- この関数はちょうど Q 回呼び出される．

入出力例

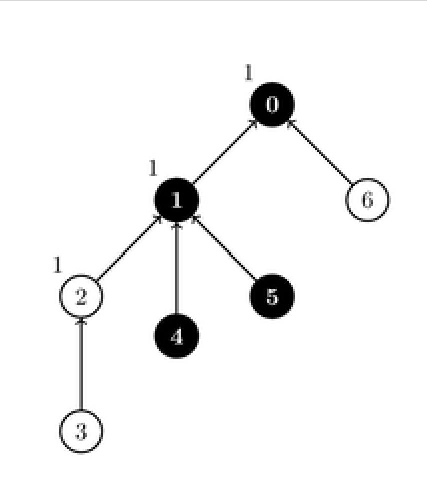
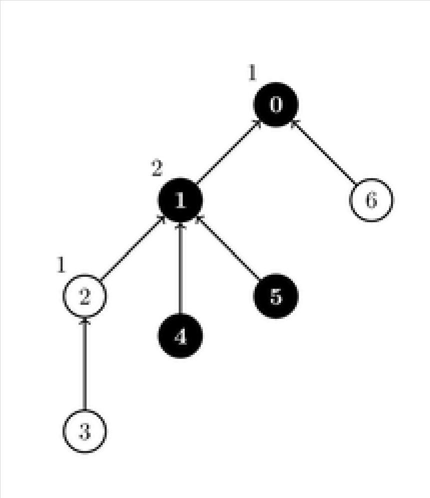
次のような呼び出しを考える．

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

この例は問題文に示されている例である．

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

この更新では，ゲート 3 とゲート 4 の状態が切り替わる．つまり，ゲート 3 の状態は 0 になり，ゲート 4 の状態は 1 になる．ゲート 0 の状態が 1 になるような2 通りのパラメータの定め方が，次の図に示されている．

Way 1	**Way** 2
	

その他のパラメータの定め方においては，ゲート 0 の状態は 0 になる．そのため，この関数は 2 を返さなければならない．

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

この更新では，ゲート 4 とゲート 5 の状態が切り替わる．結果として，すべてのソースゲートの状態が 0 となり，どのようなパラメータの定め方においても，ゲート 0 の状態は 0 となる．よって，この関数

は 0 を返さなければならない．

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

この更新により，すべてのソースゲートの状態が 1 になる．そのため，どのようなパラメータの定め方においても，ゲート 0 の状態は 1 となる．よって，この関数は 6 を返さなければならない．

制約

- $1 \leqq N, M \leqq 100\,000$
- $1 \leqq Q \leqq 100\,000$
- $P[0] = -1$
- $0 \leqq P[i] < i, P[i] \leqq N-1$ ($1 \leqq i \leqq N+M-1$)
- それぞれの閾値付きゲートは少なくとも 1 つ入力をもつ．($0 \leqq i \leqq N-1$ を満たすそれぞれの i に対して，$i < x \leqq N+M-1$ かつ $P[x] = i$ を満たす x が存在する．)
- $0 \leqq A[j] \leqq 1$ ($0 \leqq j \leqq M-1$)
- $N \leqq L \leqq R \leqq N+M-1$

小課題

1. (2 点) $N = 1, M \leqq 1000, Q \leqq 5$
2. (7 点) $N, M \leqq 1000, Q \leqq 5$, 各閾値付きゲートはちょうど 2 つの入力をもつ．
3. (9 点) $N, M \leqq 1000, Q \leqq 5$
4. (4 点) $M = N+1$, $M = 2^z$ (z は正の整数）と表せる, $P[i] = \lfloor \frac{i-1}{2} \rfloor$ ($1 \leqq i \leqq N+M-1$), $L = R$
5. (12 点) $M = N+1$, $M = 2^z$ (zは正の整数）と表せる, $P[i] = \lfloor \frac{i-1}{2} \rfloor$ ($1 \leqq i \leqq N+M-1$)
6. (27 点) 各閾値付きゲートはちょうど 2 つの入力をもつ．
7. (28 点) $N, M \leqq 5000$
8. (11 点) 追加の制約はない．

採点プログラムのサンプル

採点プログラムのサンプルは次の形式で入力を読み込む．

- 1 行目: $N\ M\ Q$
- 2 行目: $P[0]\ P[1]\ \ldots\ P[N+M-1]$
- 3 行目: $A[0]\ A[1]\ \ldots\ A[M-1]$
- $4+k$ ($0 \leqq k \leqq Q-1$) 行目: $k+1$ 回目の更新における $L\ R$

採点プログラムのサンプルは以下の形式であなたの答えを出力する．

- $1+k$ ($0 \leqq k \leqq Q-1$) 行目: $k+1$ 回目の更新に対応する `count_ways` 関数の呼び出しの戻り値．