

## 地牢遊戲 (Dungeons Game)

羅伯特正在設計一款新的電腦遊戲。遊戲涉及一個英雄,  $n$  個對手和  $n + 1$  個地牢。對手的編號從  $0$  到  $n - 1$ , 地牢的編號從  $0$  到  $n$ 。對手  $i$  ( $0 \leq i \leq n - 1$ ) 位於地牢  $i$  並且有力量  $s[i]$ 。地牢  $n$  中沒有對手。

英雄開始時進入了地牢  $x$ , 力量  $z$ 。每當英雄進入任何地牢  $i$  ( $0 \leq i \leq n - 1$ ) 時, 他們都會與對手  $i$  對抗, 並且會發生以下情況之一:

- 如果英雄的力量大於或等於對手的力量  $s[i]$ , 則英雄獲勝。這會導致英雄的力量增加  $s[i]$  ( $s[i] \geq 1$ )。在這種情況下, 英雄接下來進入地牢  $w[i]$  ( $w[i] > i$ )。
- 否則, 英雄失敗。這會導致英雄的力量增加  $p[i]$  ( $p[i] \geq 1$ )。在這種情況下, 英雄接下來進入地牢  $l[i]$ 。

注意  $p[i]$  可能小於、等於或大於  $s[i]$ 。此外,  $l[i]$  可能小於、等於或大於  $i$ 。不管對抗的結果如何, 對手都留在地牢  $i$  並保持力量  $s[i]$ 。

當英雄進入地牢  $n$  時遊戲結束。可以證明, 不管英雄的起始地牢和力量如何, 遊戲可在有限次數的對抗後結束。

羅伯特讓你通過運行  $q$  次模擬來測試他的遊戲。對於每次模擬, 羅伯特定義了一個起始地牢  $x$  和起始力量  $z$ 。您的任務是在每次模擬中找出遊戲結束時英雄的力量值。

## 實現細節

您應該編寫以下函數:

```
void init(int n, int[] s, int[] p, int[] w, int[] l)
```

- $n$ : 對手的數量。
- $s, p, w, l$ : 長度為  $n$  的數組。對於  $0 \leq i \leq n - 1$ :
  - $s[i]$  是對手  $i$  的力量。也是英雄戰勝對手  $i$  後獲得的力量。
  - $p[i]$  是英雄在輸給對手  $i$  後獲得的力量。
  - $w[i]$  是英雄戰勝對手  $i$  後進入的地牢。
  - $l[i]$  是英雄輸給對手  $i$  後進入的地牢。
- 在任何對 `simulate` 的調用之前, 這個函數只被調用一次(見下文)。

```
int64 simulate(int x, int z)
```

- $x$ : 英雄首先進入的地牢。

- $z$  : 英雄的起始力量。
- 這個程序應該在遊戲結束時返回英雄的力量, 假設英雄通過進入地牢  $x$  開始遊戲, 力量  $z$ 。
- 這個函數被恰好調用  $q$  次。

## 樣例

考慮以下的調用：

```
init(3, [2, 6, 9], [3, 1, 2], [2, 2, 3], [1, 0, 1])
```

上圖說明了這個調用。每個方格顯示一個地牢。對於地牢  $0$ 、 $1$  和  $2$ , 值  $s[i]$  和  $p[i]$  顯示在方塊內。紅色箭頭表示英雄贏得對抗後移動的位置, 而黑色箭頭表示英雄失敗後移動的位置。

假設評分程式調用 `simulate(0, 1)`。

遊戲函數如下：

地牢	英雄對抗前的實力	結果
0	1	輸
1	4	輸
0	5	贏
2	7	輸
1	9	贏
2	15	贏
3	24	遊戲結束

因此, 該函數應返回 **24**。

假設評分者調用 `simulate(2, 3)`。

遊戲函數如下：

地牢	英雄對抗前的實力	結果
2	3	輸
1	5	輸
0	6	贏
2	8	輸
1	10	贏
2	16	贏
3	25	遊戲結束

因此, 該函數應返回 25。

## 限制

- $1 \leq n \leq 400\,000$
- $1 \leq q \leq 50\,000$
- $1 \leq s[i], p[i] \leq 10^7$  (對於所有  $0 \leq i \leq n-1$ )
- $0 \leq l[i], w[i] \leq n$  (對於所有  $0 \leq i \leq n-1$ )
- $w[i] > i$  (對於所有  $0 \leq i \leq n-1$ )
- $0 \leq x \leq n-1$
- $1 \leq z \leq 10^7$

## 子任務

1. (11 分)  $n \leq 50\,000$ ,  $q \leq 100$ ,  $s[i], p[i] \leq 10\,000$  (對於所有  $0 \leq i \leq n-1$ )
2. (26 分)  $s[i] = p[i]$  (對於所有  $0 \leq i \leq n-1$ )
3. (13 分)  $n \leq 50\,000$ , 所有對手力量都相同, 即  $s[i] = s[j]$  (對於所有  $0 \leq i, j \leq n-1$ )。
4. (12 分)  $n \leq 50\,000$ , 且最多只有 5 個不同的  $s[i]$  值。
5. (27 分)  $n \leq 50\,000$
6. (11 分) 沒有額外的限制。

## 樣例評分程式

樣例評分程式按以下格式讀取輸入：

- 第 1 行:  $n\ q$
- 第 2 行:  $s[0]\ s[1]\ \dots\ s[n-1]$
- 第 3 行:  $p[0]\ p[1]\ \dots\ p[n-1]$
- 第 4 行:  $w[0]\ w[1]\ \dots\ w[n-1]$
- 第 5 行:  $l[0]\ l[1]\ \dots\ l[n-1]$
- 第  $6+i$  行 ( $0 \leq i \leq q-1$ ):  $x\ z$  用於對 `simulate` 的第  $i$  次調用。

樣例評分程式按以下格式打印您的答案：

- 第  $1+i$  行 ( $0 \leq i \leq q-1$ )：第  $i$  次調用 `simulate` 的返回值。