



玩具火車

Arezou和她的兄弟Borzou是雙胞胎。他們收到的生日禮物是一套好玩的玩具火車。他們用它建了一個有 n 個車站和 m 段單向軌道的鐵路系統。這些車站的編號是從 0 到 $n - 1$ 。每段軌道都始於某一車站，然後終於同一車站或其他車站。每個車站至少會有一段軌道以它為起點。

其中有些車站是充電車站。無論何時，如果火車抵達某個充電車站，它都會被充到滿電。滿電火車擁有足夠的動力連續地駛過 n 段軌道，但是如果不再充電的話，在即將進入第 $n + 1$ 段軌道時它就會因電已用光而停車。

每個車站都有一個軌道開關，可以扳向任一以該車站為起點的軌道。火車從某個車站駛出時，駛向的正是該車站的開關所扳向的軌道。

這對雙胞胎打算用他們的火車玩個遊戲。他們已經分完了所有的車站：每個車站要麼歸Arezou，要麼歸Borzou。遊戲裡面只有一列火車。遊戲開始時，這列火車停在車站 s ，並且充滿了電。為啟動遊戲，車站 s 的擁有者把車站 s 的開關扳向某個以 s 為起點的軌道。隨後他們啟動火車，火車也就開始沿著軌道行駛。

無論何時，在火車首次進入某一車站時，該車站的擁有者都要扳定車站開關。開關一旦扳定，它就會保持狀態不變直到遊戲結束。因此，火車如果開到了某個曾經進過的車站，就會沿著與之前相同的軌道離開。

由於車站數量是有限的，火車最終都會落入某個環而行駛。環是指一系列不同的車站 $c[0], c[1], \dots, c[k - 1]$ ，其中火車在離開車站 $c[i]$ （這裡 $0 \leq i < k - 1$ ）後駛上連向車站 $c[i + 1]$ 的軌道，在離開車站 $c[k - 1]$ 後駛上連向車站 $c[0]$ 的軌道。一個環可能只包括一個車站（此時 $k = 1$ ），即火車從車站 $c[0]$ 駛出後又駛上了連向 $c[0]$ 的軌道。

如果火車能夠連續行駛跑個沒完，Arezou就贏了。否則火車最後會把電用光而停車，這樣就是Borzou贏。換句話說，如果在車站 $c[0], c[1], \dots, c[k - 1]$ 中至少有一個充電車站，使得火車能夠不斷地充電而沿著環跑個沒完，Arezou贏。否則，它就會最終把電用光（有可能是在沿著環跑好幾圈後），Borzou贏。

現在給你一個這樣的鐵路系統。Arezou和Borzou將會玩 n 輪遊戲。其中在第 s 輪遊戲中（ $0 \leq s \leq n - 1$ ），火車的起始車站是 s 。你的任務是，對每一輪遊戲，判斷Arezou是否無論Borzou怎麼玩都必勝。

實現細節

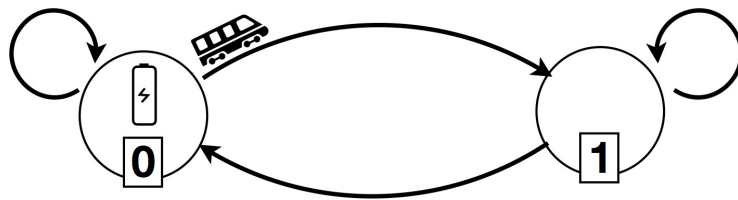
你需要實現下面的函數：

```
int[] who_wins(int[] a, int[] r, int[] u, int[] v)
```

- a : 長度為 n 的陣列。如果Arezou擁有車站 i ，則 $a[i] = 1$ ；否則Borzou擁有車站 i ，且 $a[i] = 0$ 。
- r : 長度為 n 的陣列。如果車站 i 是充電車站，則 $r[i] = 1$ 。否則 $r[i] = 0$ 。
- u 和 v : 長度為 m 的陣列。對於所有 $0 \leq i \leq m - 1$ ，存在某一單向軌道，且其起點為 $u[i]$ ，終點為 $v[i]$ 。
- 該函數需要返回一個長度為 n 的陣列 w 。對於每個 $0 \leq i \leq n - 1$ ，如果在火車的起始車站為 i 的遊戲中，Arezou不管Borzou怎麼玩都能贏， $w[i]$ 的值應為1。否則 $w[i]$ 的值應為0。

樣例

```
who_wins([0, 1], [1, 0], [0, 0, 1, 1], [0, 1, 0, 1])
```



- 這裡有2個車站。Borzou擁有充電車站0。Arezou擁有車站1，但是它不是充電車站。
- 這裡有4段軌道 $(0, 0)$, $(0, 1)$, $(1, 0)$ 和 $(1, 1)$ ，其中 (i, j) 表示一個以車站 i 為起點、車站 j 為終點的單向軌道。
- 考慮火車的起始車站為0的遊戲。如果Borzou將車站0的開關扳向軌道 $(0, 0)$ ，那麼火車就會沿著這個環形軌道跑個沒完（注意，車站0是一個充電車站）。在這種情況下，Arezou贏。否則，如果Borzou把車站0的開關扳向軌道 $(0, 1)$ ，Arezou可以把車站1的開關扳向軌道 $(1, 0)$ 。這樣的話，火車將會在兩個車站之間跑個不停。Arezou還是會贏，因為車站0是充電車站，火車將跑個沒完。因此，無論Borzou怎麼玩，Arezou都會贏。
- 根據類似的推演，在火車的起始車站為1的遊戲中，無論Borzou怎麼玩，Arezou也都會贏。因此，函數應當返回 $[1, 1]$ 。

資料範圍和限制

- $1 \leq n \leq 5000$.
- $n \leq m \leq 20\,000$.
- 至少會有一個充電車站。
- 每個車站至少會有一段軌道以它為起點。
- 可能會有起點和終點相同的軌道（即 $u[i] = v[i]$ ）。
- 所有軌道兩兩不同。也就是說，不存在這樣的兩個下標 i 和 j （ $0 \leq i < j \leq m - 1$ ），使得 $u[i] = u[j]$ 且 $v[i] = v[j]$ 。
- 對於所有 $0 \leq i \leq m - 1$ ，都有 $0 \leq u[i], v[i] \leq n - 1$.

子任務

1. (5分) 對於所有 $0 \leq i \leq m - 1$ ，都有 $v[i] = u[i]$ 或者 $v[i] = u[i] + 1$.
2. (10分) $n \leq 15$.
3. (11分) Arezou 擁有所有車站。
4. (11分) Borzou 擁有所有車站。
5. (12分) 充電車站的數量為1。
6. (51分) 無任何限制。

評測方式

評測系統會按照下述格式來讀取輸入資料：

- 第1行： $n \ m$
- 第2行： $a[0] \ a[1] \ \dots \ a[n - 1]$
- 第3行： $r[0] \ r[1] \ \dots \ r[n - 1]$
- 第4 + i 行 (對於所有 $0 \leq i \leq m - 1$)： $u[i] \ v[i]$

評測系統會按照下述格式列印出 who_wins 的返回結果：

- 第1行： $w[0] \ w[1] \ \dots \ w[n - 1]$