



## Влакче-играчка

Арезу и Борзу са близнаци. За рождения си ден те получили прекрасна играчка–влакче и решили да си построят железопътна мрежа с  $n$  гари, номерирани от 0 до  $n - 1$  и  $m$  еднопосочни линии. Всяка линия тръгва от една гара и достига до друга гара, не непременно различна от началната. От всяка гара излиза поне една линия.

На някои от станциите влакчето може да се зарежда, като зареждането винаги е пълно. Зареденото влакче може да премине точно през  $n$  последователни линии. Т.е., в момента в който, след зареждане, премине през  $n$  линии и тръгне по  $(n + 1)$ -та, енергията му свършва и то спира.

На всяка гара има превключвател, с който влакчето може да се насочи по някоя от излизащите от гарата линии. Достигайки гара, влакчето може да я напусне само по линията, към която е насочен превключвателя.

Близнаците играят следната игра. Разделили са станциите помежду си – всяка станция е собственост или на Арезу или на Борзу. Влакчето е едно, заредено напълно и се намира на начална гара  $s$ . Собственикът на гарата избира с превключвателя една от излизащите от гарата линии и влакчето тръгва по нея. И това се повтаря на всяка гара, на която влакчето попадне. След като е избрана линията, по която влакчето да напусне гарата, превключването към друга линия е невъзможно и всеки път, когато то влезе отново в същата гара я напуска по същата линия.

Тъй като броят на гарите е краен, рано или късно влакчето влиза в *цикъл* – редица от *различни* гари  $c[0], c[1], \dots, c[k - 1]$  такива, че след като напусне гарата  $c[i]$  (за  $0 \leq i < k - 1$ ), то тръгва към гарата  $c[i + 1]$ , а когато напусне гарата  $c[k - 1]$  тръгва към  $c[0]$ . Цикълът може да бъде съставен само от една гара (т.е.  $k = 1$ ), ако има линия която тръгва от гара  $c[0]$  и стига до същата гара.

Арезу печели играта, ако влакчето може да се движи безкрайно, а Борзу – ако влакчето спре. С други думи, ако в цикъла има поне една зареждаща гара – влакът ще може да зарежда при всяко преминаване през нея и да се движи вечно, затова ще спечели Арезу. В противен случай, то ще свърши горивото след няколко преминавания по цикъла и Борзу ще спечели играта.

Зададено е описание на железопътната мрежа. Арезу и Борзу играят  $n$  игри. В  $s$ -тата игра, за  $0 \leq s \leq n - 1$ , влакчето в началото е на гарата  $s$ . Задачата е, за всяка игра да се определи, дали Арезу има стратегия, която да му гарантира, че печели играта, независимо как играе Борзу.

## Детайли на реализацията

Напишете следната процедура:

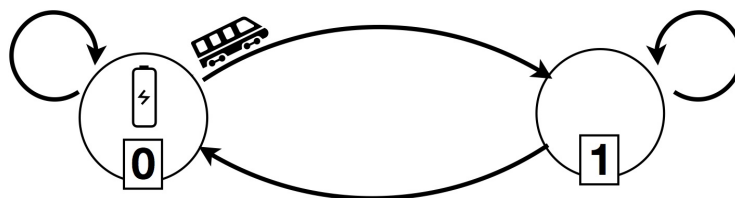
```
int[] who_wins(int[] a, int[] r, int[] u, int[] v)
```

където

- $a$ : масив с дължина  $n$ . Ако Арезу е собственик на гарата  $i$ ,  $a[i] = 1$ . Ако Арезу не е собственик на гарата  $i$ , тогава  $a[i] = 0$ .
- $r$ : масив с дължина  $n$ . Ако гарата  $i$  е зареждаща, тогава  $r[i] = 1$ . Иначе,  $r[i] = 0$ .
- $u$  и  $v$ : масиви с дължина  $m$ . За всяко  $i$ ,  $0 \leq i \leq m - 1$ , в мрежата има еднопосочна линия от  $u[i]$  до  $v[i]$ .
- Процедурата трябва да връща масив  $w$  с дължина  $n$ . За всяко  $i$ ,  $0 \leq i \leq n - 1$ , стойността на  $w[i]$  трябва да е 1, ако Арезу печели играта при започване от гарата  $i$ , независимо как играе Борозу. В противен случай, стойността на  $w[i]$  трябва да е 0.
- За означението `int[]` вижте първата таблица в Notice.

## Пример

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



- Мрежата е с 2 гари. Борозу е собственик на станция 0, която е зареждаща, а Арезу - на гарата 1, която не е зареждаща.
- Мрежата има 4 линии  $(0, 0)$ ,  $(0, 1)$ ,  $(1, 0)$  и  $(1, 1)$ , където  $(i, j)$  означава, че има еднопосочна линия от  $i$  до  $j$ .
- Нека в началото влакчето е на гара 0. Ако Борозу насочи влакчето към същата гара по линията  $(0, 0)$ , то ще влезе в цикъл със зареждаща гара и Арезу ще спечели. А ако го насочи към гара 1 по линията  $(0, 1)$ , тогава Арезу може да го върне обратно към станция 0 по линията  $(1, 0)$ , затваряйки цикъл, който преминава през двете станции, едната от които е зареждаща и Арезу печели отново. Значи Арезу ще спечели в този случай, независимо от играта на Борозу.
- Ако влакчето в началото е на гара 1, по аналогичен начин се установява, че Арезу печели играта, независимо какво играе Борозу. Затова процедурата връща  $[1, 1]$ .

## Ограничения

- $1 \leq n \leq 5000$ .
- $n \leq m \leq 20\,000$ .
- Винаги има поне една зареждаща гара.
- От всяка гара излиза поне една линия.
- Може да има линия, която започва от и завършва в една и съща гара.
- Всички линии са различни, т.е. няма две такива линии  $i$  и  $j$  ( $0 \leq i < j \leq m - 1$ ), че  $u[i] = u[j]$  и  $v[i] = v[j]$ .
- $0 \leq u[i], v[i] \leq n - 1$  (за всяко  $i$ ,  $0 \leq i \leq m - 1$ ).

## Подзадачи

1. (5 точки) За всяко  $i$ ,  $0 \leq i \leq m - 1$ , или  $v[i] = u[i]$  или  $v[i] = u[i] + 1$ .
2. (10 точки)  $n \leq 15$ .
3. (11 точки) Арезу притежава всички гари.
4. (11 points) Борзу притежава всички гари.
5. (12 точки) Зареждащата гара е точно една.
6. (51 точки) Без допълнителни ограничения.

## Примерен грейдър

Примерният грейдър чете вход във формат:

- line 1:  $n \ m$
- line 2:  $a[0] \ a[1] \ \dots \ a[n - 1]$
- line 3:  $r[0] \ r[1] \ \dots \ r[n - 1]$
- line  $4 + i$  (for  $0 \leq i \leq m - 1$ ):  $u[i] \ v[i]$

Грейдерът печата върнатите от процедурата `who_wins` във формата:

- line 1:  $w[0] \ w[1] \ \dots \ w[n - 1]$