



Kereta Mainan

Arezou dan saudara laki-lakinya, Borzou, adalah saudara kembar. Mereka telah menerima seperangkat kereta mainan canggih untuk ulang tahun mereka, dan mereka menggunakannya untuk membangun sebuah sistem rel kereta dengan n stasiun dan m jalur *satu arah*. Stasiun-stasiun tersebut dinomori dari 0 hingga $n - 1$. Setiap jalur dimulai pada sebuah stasiun dan berakhir pada stasiun yang sama atau berbeda. Terdapat setidaknya sebuah jalur yang dimulai pada setiap stasiun.

Beberapa stasiun adalah *stasiun isi ulang*. Setiap kali sebuah kereta tiba di sebuah stasiun isi ulang, baterai kereta tersebut menjadi penuh. Sebuah kereta berbaterai penuh memiliki energi yang cukup untuk melintasi n jalur berurutan. Yakni, kereta tersebut akan kehabisan energi tepat ketika kereta tersebut memasuki jalur ke- $(n + 1)$ setelah diisi ulang terakhir kali.

Pada setiap stasiun, terdapat sebuah kenop yang dapat diarahkan ke jalur manapun yang dimulai pada stasiun tersebut. Ketika sebuah kereta berada pada sebuah stasiun, kereta tersebut akan meninggalkan stasiun tersebut melintasi jalur yang diarahkan oleh kenop pada stasiun tersebut.

Saudara kembar tersebut akan bermain menggunakan kereta mereka. Mereka sudah membagi semua stasiun di antara mereka: setiap stasiun dimiliki oleh salah satu dari Arezou atau Borzou. Terdapat tepat sebuah kereta. Pada awal permainan, kereta tersebut berada pada stasiun s dan berbaterai penuh. Untuk memulai permainan, pemilik stasiun s mengarahkan kenop pada stasiun s ke salah satu jalur yang dimulai pada stasiun s . Kemudian, mereka menyalakan kereta tersebut dan kereta tersebut mulai melintasi jalur-jalur yang ada.

Setiap kali kereta tersebut memasuki sebuah stasiun untuk pertama kalinya, pemilik stasiun tersebut mengarahkan kenop pada stasiun tersebut. Setelah sebuah kenop diarahkan, kenop tersebut akan berada pada posisi yang sama sampai akhir permainan. Oleh karena itu, ketika sebuah kereta kembali memasuki stasiun yang sudah pernah dikunjungi sebelumnya, kereta tersebut akan meninggalkan stasiun tersebut melintasi jalur yang sama dengan sebelumnya.

Karena banyaknya stasiun adalah berhingga, kereta tersebut pada akhirnya akan berputar-putar pada sebuah *siklus*. Sebuah siklus adalah sebuah barisan stasiun *berbeda-beda* $c[0], c[1], \dots, c[k - 1]$ sedemikian sehingga kereta tersebut meninggalkan stasiun $c[i]$ (untuk $0 \leq i < k - 1$) melintasi sebuah jalur ke stasiun $c[i + 1]$, dan meninggalkan stasiun $c[k - 1]$ melintasi sebuah jalur ke stasiun $c[0]$. Perhatikan bahwa sebuah siklus mungkin saja terdiri atas sebuah stasiun (yakni, berlaku $k = 1$) jika kereta tersebut meninggalkan stasiun $c[0]$ melintasi sebuah jalur yang menuju ke $c[0]$ kembali.

Arezou memenangkan permainan tersebut jika kereta tersebut terus berjalan selamanya, dan Borzou menang jika kereta tersebut kehabisan energi. Dengan kata lain, jika terdapat setidaknya

sebuah stasiun isi ulang di antara $c[0], c[1], \dots, c[k-1]$, maka kereta tersebut dapat mengisi ulang dan berputar-putar selamanya, dan Arezou menang. Jika tidak, kereta tersebut akan kehabisan energi (mungkin saja setelah melintasi siklus beberapa kali), dan Borzou menang.

Anda diberikan deskripsi dari sistem rel kereta tersebut. Arezou dan Borzou akan memainkan n permainan. Pada permainan ke- s , untuk $0 \leq s \leq n-1$, kereta tersebut pada mulanya akan berada pada stasiun s . Tugas Anda adalah untuk menentukan, untuk setiap permainan, apakah terdapat sebuah strategi untuk Arezou yang menjamin ia menang, bagaimanapun Borzou bermain.

Detail implementasi

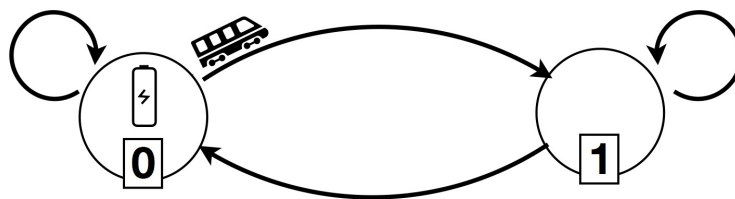
Anda harus mengimplementasikan prosedur berikut:

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

- a : array sepanjang n . Jika Arezou memiliki stasiun i , $a[i] = 1$. Jika tidak, Borzou memiliki stasiun i dan $a[i] = 0$.
- r : array sepanjang n . Jika stasiun i adalah stasiun isi ulang, $r[i] = 1$. Jika tidak, $r[i] = 0$.
- u dan v : array-array sepanjang m . Untuk setiap $0 \leq i \leq m-1$, terdapat sebuah jalur satu arah dimulai pada stasiun $u[i]$ dan berakhir pada stasiun $v[i]$.
- Prosedur ini harus mengembalikan sebuah array w sepanjang n . Untuk setiap $0 \leq i \leq n-1$, nilai dari $w[i]$ haruslah 1 jika Arezou dapat memenangkan permainan yang dimulai pada stasiun i , bagaimanapun Borzou bermain. Jika tidak, nilai dari $w[i]$ haruslah 0.

Contoh

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



- Terdapat 2 stasiun. Borzou adalah pemilik stasiun 0, yang merupakan stasiun isi ulang. Arezou adalah pemilik stasiun 1, yang bukan merupakan stasiun isi ulang.
- Terdapat 4 jalur $(0,0)$, $(0,1)$, $(1,0)$, dan $(1,1)$, yang mana (i,j) menyatakan sebuah jalur satu arah dari stasiun i menuju stasiun j .
- Pertimbangkan permainan yang mana kereta pada mulanya berada pada stasiun 0. Jika Borzou mengarahkan kenop pada stasiun 0 ke jalur $(0,0)$, kereta tersebut akan berputar-putar melintasi jalur ini (perhatikan bahwa stasiun 0 adalah stasiun isi ulang). Pada kasus ini, Arezou menang. Sebaliknya, jika Borzou mengarahkan kenop pada stasiun 0 ke jalur $(0,1)$, Arezou dapat mengarahkan kenop pada stasiun 1 ke $(1,0)$. Jika ini terjadi, kereta tersebut akan

berputar-putar selamanya melewati kedua stasiun. Lagi-lagi Arezou menang, karena stasiun 0 adalah stasiun isi ulang dan kereta tersebut tidak akan pernah berhenti. Maka, Arezou dapat memenangkan permainan, bagaimanapun Borzou bermain.

- Dengan alasan yang mirip, jika permainan dimulai pada stasiun 1 Arezou juga dapat menang, bagaimanapun Borzou bermain. Maka, prosedur tersebut harus mengembalikan $[1, 1]$.

Batasan

- $1 \leq n \leq 5000$.
- $n \leq m \leq 20\,000$.
- Terdapat setidaknya sebuah stasiun isi ulang.
- Terdapat setidaknya sebuah jalur yang dimulai pada setiap stasiun.
- Mungkin saja terdapat jalur-jalur yang dimulai dan berakhir pada stasiun yang sama (yakni, $u[i] = v[i]$).
- Semua jalur adalah berbeda-beda. Dengan kata lain, tidak akan terdapat dua indeks i dan j ($0 \leq i < j \leq m - 1$) yang memenuhi $u[i] = u[j]$ dan $v[i] = v[j]$.
- $0 \leq u[i], v[i] \leq n - 1$ (untuk setiap $0 \leq i \leq m - 1$).

Subsoal

1. (5 poin) Untuk setiap $0 \leq i \leq m - 1$, berlaku $v[i] = u[i]$ atau $v[i] = u[i] + 1$.
2. (10 poin) $n \leq 15$.
3. (11 poin) Arezou memiliki semua stasiun.
4. (11 poin) Borzou memiliki semua stasiun.
5. (12 poin) Terdapat tepat sebuah stasiun isi ulang.
6. (51 poin) Tidak ada batasan tambahan.

Grader contoh

Grader contoh membaca masukan dengan format berikut:

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

Grader contoh mencetak nilai kembalian dari `who_wins` dengan format berikut:

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