Воз играчка

Ацко и брат му Бацко се близнаци. За нивниот роденден тие добија прекрасен сет играчка – воз, која ја користат за да изградат железница со n станици и m *еднонасочни* пруги. Станиците се нумерирани со броевите од 0 до $n-1$. Секоја пруга започнува од една станица и завршува во истата или друга станица. Од секоја станица започнува барем една пруга.

Некои од станиците се *напојувачки станици*. Секогаш кога возот ќе пристигне во ваква станица, тој се полни целосно (до даска). Возот кој е целосно наполнет има доволно енергија да помине n последователни пруги. Со други зборови, на возот му снемува енергија токму во моментот кога влегува во $(n+1)$-та пруга откога бил наполнет за последен пат пред тоа.

На секоја станица има пренасочувач кој може да се насочи кон било која пруга која почнува во таа станица. Кога возот е во некоја станица, тој ја напушта таа станица користејќи ја пругата кон која е насочен пренасочувачот на таа станица.

Близнаците си играат со нивниот воз. Тие веќе си ги имаат поделено станиците меѓу себе: со секоја станица управува или Ацко или Бацко. Има само еден воз. На почетокот на играта возот е во станицата s и тој е целосно наполнет. За да започне играта, оној кој ја управува станицата s го насочува пренасочувачот кон некоја од пругите што почнува од станицата s. Потоа тие го вклучуваат возот и тој почнува да се движи по железницата.

Секогаш кога возот ќе влезе за прв пат во некоја станица, оној кој управува со таа станица го подесува пренасочувачот на таа станица. Кога еднаш ќе се подеси, тој останува на истата позиција до крајот на играта. Според тоа, ако возот се врати во некоја станица, истата ќе ја напушти по истата пруга како и претходно.

Бидејќи постојат конечен број на станици, возот евентуално некогаш ќе почне да се движи во *циклус*. Циклус е низа од *различни* станици $c[0], c[1], \cdots, c[k-1]$ таква што возот ја напушта станицата $c[i]$ (за $0 \leq i < k-1$) преку пругата која оди кон станицата $c[i+1]$, и ја напушта станицата $c[k-1]$ преку пругата која оди до станицата $c[0]$. Забележете дека циклусот може да се состои и од само една станица (т.е. има $k=1$), ако возот ја напушта станицата $c[0]$ преку пруга која се враќа пак во $c[0]$.

Ацко победува во играта ако возот може да се движи бесконечно, додека Бацко е победник ако на возот му снема енергија. Поточно, ако во циклусот $c[0], c[1], \cdots, c[k-1]$ постојат доволно напојувачки станици, така да возот може постојано да врти во циклус, тогаш Ацко е победникот. Инаку, на возот ќе му снема енергија (може да се случи да го заврти циклусот неколку пати), па ќе победи Бацко.

Даден е описот на железницата. Ацко и Бацко ќе играат n игри. Во s-тата игра, за

$0 \leq s \leq n-1$, возот на почетокот ќе биде на станицата s. Ваша задача е да најдете, за секоја игра, дали постои стратегија која гарантира дека ќе победи Ацко, независно од тоа како ќе игра Бацко.

Детали за имплементација

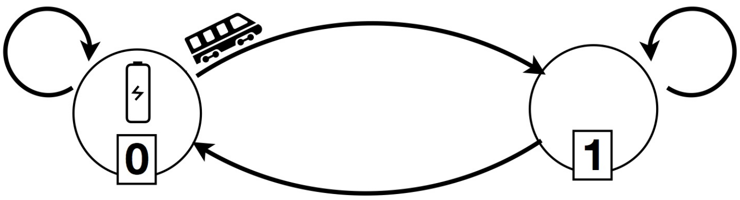
Треба да ја имплементирате следнава процедура:

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

- a: низа со должина n. Ако Ацко ја управува станицата i, $a[i] = 1$. Во спротивно, Бацко ја управува станицата i и $a[i] = 0$.
- r: низа со должина n. Ако станицата i е напојувачка станица,$r[i] = 1$. Во спротивно, $r[i] = 0$.
- u и v: низи со должина m. За секое $0 \leq i \leq m-1$, постои еднонасочна пруга која започнува од станицата $u[i]$ и завршува во станицата $v[i]$.
- Оваа процедура треба да врати низа w со должина n. За секое $0 \leq i \leq n-1$, вредноста на $w[i]$ треба да биде 1 ако Ацко може да победи во игра која започнува во станицата i, без разлика како игра Бацко. Инаку, вредноста $w[i]$ треба да биде 0.

Пример

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



- Има 2 станици. Бацко ја управува станицата 0, која е напојувачка станица. Ацко ја управува станицата 1, која не е напојувачка станица.
- Има 4 пруги $(0,0), (0,1), (1,0)$, и $(1,1)$, каде (i,j) ја означува еднонасочната пруга од станицата i до станицата j.
- Да ја разгледаме играта во која возот почнува од станицата 0. Ако Бацко го подеси пренасочувачот на станицата 0 кон пругата $(0,0)$, возот бесконечно ќе врти во циклусот кој ја содржи само оваа пруга (затоа што станицата 0 е напојувачка станица). Во овој случај победува Ацко. Од друга страна, ако Бацко го подеси пренасочувачот на станицата 0 кон пругата $(0,1)$, Ацко може да го подеси пренасочувачот на станицата 1 кон $(1,0)$. Ако се случи тоа, возот влегува во бесконечен цилус низ двете станици. Повторно победува Ацко, бидејќи станицата 0 е напојувачка станица и возот нема да застане. Така да, Ацко може да победи без разлика како ќе одигра Бацко.
- Со слично размислување, во играта која започнува од станицата 1, Ацко може секогаш

да победи, без разлика како ќе одигра Бацко. Оттука процедурата треба да врати $[1, 1]$.

Ограничувања

- $1 \le n \le 5000$.
- $n \le m \le 20\,000$.
- Има барем една напојувачка станца.
- Од секоја станица започнува барем една пруга.
- Може да има пруги што почнуваат и завршуваат во истата станица (т.е., $u[i] = v[i]$).
- Секоја пруга е различна. Со други зборови, не постојат два индекси i и j ($0 \le i < j \le m-1$) такви што $u[i] = u[j]$ и $v[i] = v[j]$.
- $0 \le u[i], v[i] \le n-1$ (за сите $0 \le i \le m-1$).

Подзадачи

1. (5 поени) За сите $0 \le i \le m-1$, или $v[i] = u[i]$ или $v[i] = u[i] + 1$.
2. (10 поени) $n \le 15$.
3. (11 поени) Ацко ги управува сите станици.
4. (11 поени) Бацко ги управува сите станици.
5. (12 поени) Има точно една напојувачка станица.
6. (51 поени) Нема дополнителни ограничувања.

Пример оценувач (grader)

Пример оценувачот го чита влезот во следниов формат:

- линија 1: $n\ \ m$
- линија 2: $a[0]\ \ a[1]\ \ \ldots\ \ a[n-1]$
- линија 3: $r[0]\ \ r[1]\ \ \ldots\ \ r[n-1]$
- линија $4+i$ (за $0 \le i \le m-1$): $u[i]\ \ v[i]$

Пример оценувачот ја печати вредноста која ја враќа процедурата `who_wins` во следниов формат:

- линија 1: $w[0]\ \ w[1]\ \ \ldots\ \ w[n-1]$