لعبة القطارات

أميرة وأخوها باسم توأم. حصلا مؤخراً على لعبة قطار رائعة كهدية لعيد ميلادهما, وسيقومان باستخدامها لبناء نظام سكك حديدية يحتوي على n محطة و m وصلة *وحيدة الجهة*. كل وصلة تبدأ بمحطة وتنتهي بمحطة أخرى أو بنفس المحطة. يوجد على الأقل وصلة واحدة تنطلق من كل محطة.

تعتبر بعض المحطات *محطات شحن* للقطار. عندما يصل القطار إلى إحدى محطات الشحن يتم شحنه بالكامل. عندما يكون القطار مشحوناً بالكامل فإنه يستطيع التجول لـ n وصلة متتالية. بمعنى آخر, يفرغ القطار من الشحن عندما يدخل الوصلة $(n+1)$ بعد آخر شحن بالكامل له.

عند كل محطة يوجد فاصمة (مفتاح) تؤشر على وصلة واحدة من الوصلات التي تنطلق من هذه المحطة, عندما يكون القطار موجود في محطة ما فإنه يغادرها باستخدام الوصلة التي تؤشر عليها فاصمة هذه المحطة.

سيقوم التوأم أميرة وباسم بلعب لعبة باستخدام هذا القطار, حيث قاما مسبقاً بتقسيم المحطات فيما بينهما, بحيث يكون لكل محطة مالك وحيد (أميرة أو باسم). طبعاً لدينا قطار وحيد, سينطلق في بداية اللعبة من المحطة s وسيكون مشحوناً بالكامل. عندما تبدأ اللعبة, يقوم مالك المحطة s بوضع فاصمة المحطة s بحيث تؤشر على إحدى الوصلات التي تنطلق من المحطة s, ثم يتم تشغيل القطار ليبدأ بالتجول على الوصلات.

عندما يدخل القطار إحدى المحطات لأول مرة, يقوم صاحب هذه المحطة بوضع الفاصمة الخاصة بها على وضعية ما. بعد أن يتم تحديد وضعية الفاصمة, تبقى الفاصمة على نفس وضعيتها طوال فترة اللعبة المتبقية. لذلك عندما يعاود القطار زيارة المحطة نفسها مرة ثانية فإنه سيغادر المحطة من نفس الوصلة التي غادرها في زيارته السابقة للمحطة.

بما أن هناك عدد محدود من المحطات, فإن القطار سيبدأ بالتجول ضمن *حلقة* في نهاية الأمر. يمكن تعريف *الحلقة* بأنها سلسلة من المحطات *المختلفة* $c[0], c[1], \cdots, c[k-1]$ بحيث يغادر القطار المحطة $c[i]$ مستخدماً وصلة تنقله إلى المحطة $c[i+1]$ (وذلك أياً كان i تحقق الشرط $0 \leq i < k-1$) , وسيقوم بمغادرة المحطة $c[k-1]$ مستخدماً وصلة تصله إلى المحطة $c[0]$. لاحظ بأن الحلقة قد تكون مؤلفة من محطة واحدة (أي $k=1$) وذلك عندما يغادر القطار المحطة $c[0]$ مستخدماً وصلة تصله بالمحطة $c[0]$.

تربح أميرة اللعبة إذا تابع القطار المسير بدون توقف إلى ما لانهاية, بينما يربح باسم اللعبة عندما ينتهي شحن القطار. بمعنى آخر, إذا كان هناك محطة شحن واحدة على الأقل ضمن سلسلة المحطات $c[0], c[1], \cdots, c[k-1]$, عندها يمكن للقطار أن يعاود شحنه ويتابع المسير في حلقة إلى ما لا نهاية وعندها تفوز أميرة. وإلا سينتهي شحن القطار ويفوز باسم (من الممكن أن يحصل ذلك بعد أن يقوم القطار بعدة دورات في الحلقة).

سيتم تزويدك بتوصيف بنية نظام السكك الحديدية, علماً أن كل المحطات مرقمة بأرقام متتالية من 0 حتى $n-1$.

أميرة وباسم سيلعبان n لعبة. في اللعبة رقم s (حيث $0 \leq s \leq n-1$) سيكون القطار في البداية في المحطة s. يتوجب عليك من أجل كل لعبة, تحديد فيما إذا كان هناك استراتيجية تمكن أميرة من ضمان ربحها مهما لعب باسم.

.

.

تفاصيل التنجيز

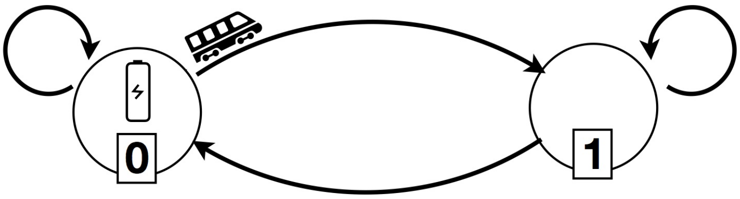
يتوجب عليك تنجيز التابع التالي:

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

- a: هي مصفوفة أحادية البعد بطول n. إذا كانت أميرة تملك المحطة رقم i فإن $a[i] = 1$ وإلا إذا كان باسم يملك المحطة رقم i فإن $a[i] = 0$.
- r: هي مصفوفة أحادية البعد بطول n. إذا كانت المحطة i هي محطة شحن, فإن $r[i] = 1$ وإلا فإن $r[i] = 0$.
- u و v: مصفوفتان بطول m. من أجل أي رقم $0 \leq i \leq m - 1$, هنالك وصلة وحيدة الجهة تنطلق من المحطة $u[i]$ وتصل إلى المحطة $v[i]$.
- يجب على هذا التابع أن يعيد مصفوفة أحادية البعد w بطول n. من أجل أي رقم $0 \leq i \leq n - 1$, يجب أن تحتوي قيمة المصفوفة $w[i]$ على القيمة 1 إذا كانت أميرة تستطيع الفوز باللعبة إذا بدأت من المحطة i بغض النظر عن طريقة لعب باسم. وإلا يجب أن تحتوي قيمة المصفوفة $w[i]$ على القيمة 0.

مثال

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



- هناك محطتين في المثال, باسم يمتلك المحطة رقم 0 والتي هي محطة شحن, بينما تمتلك أميرة المحطة رقم 1 والتي ليست محطة شحن.
- هناك 4 وصلات هي $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$, حيث الوصلة (i, j) تعبر عن وصلة وحيدة الجهة تنطلق من المحطة i إلى المحطة j.
- إذا أردنا الحديث عن اللعبة التي تبدأ من المحطة رقم 0, إذا وضع باسم الفاصمة التي في المحطة 0 لتؤشر على الوصلة $(0, 0)$, فإن القطار سيبقى يدور إلى ما لانهاية في هذه المحطة (لاحظ بأن المحطة 0 هي محطة شحن) وفي هذه الحالة فإن أميرة تفوز. وإلا إذا وضع باسم الفاصمة التي في المحطة 0 على الوصلة $(0, 1)$, فإن أميرة تستطيع وضع الفاصمة التي في المحطة 1 على الوصلة $(1, 0)$ وعندها القطار سيبقى يدور إلى ما لانهاية بين كلا المحطتين وبما أن المحطة 0 هي محطة شحن فإن أميرة ستفوز أيضاً. لذلك فإن أميرة تستطيع الفوز دائماً بغض النظر على سلوك باسم.
- بشكل مشابه يمكننا التأكد من أنه عندما تبدأ اللعبة من المحطة 1 فإن أميرة تستطيع الفوز بغض النظر عن كيفية لعب باسم, لذلك فإن التابع يعيد المصفوفة $[1, 1]$.

القيود

- $1 \leq n \leq 5000$.
- $n \leq m \leq 20\,000$.
- يوجد على الأقل محطة شحن واحدة.
- يوجد على الأقل وصلة واحدة تنطلق من كل محطة.
- يمكن أن يكون هناك وصلات تنطلق وتنهي في نفس المحطة (i.e., $u[i] = v[i]$).
- جميع الوصلات مختلفة عن بعضها البعض, بمعنى آخر, لا توجد وصلتنان لهما نفس محطة البداية والنهاية معاً .
- $0 \leq u[i], v[i] \leq n-1$ (for all $0 \leq i \leq m-1$).

المهمات الجزئية

1. (5 نقاط) من أجل $0 \leq i \leq m-1$, إما $v[i] = u[i]$ أو $v[i] = u[i] + 1$.
2. (10 نقاط) $n \leq 15$.
3. (11 نقاط) أميرة تمتلك جميع المحطات.
4. (11 نقاط) باسم يمتلك جميع المحطات .
5. (12 نقاط) هناك محطة شحن واحدة فقط حتماً.
6. (51 نقاط) ليس هناك قيود إضافية.

Sample grader

The sample grader reads the input in the following format:

- line1 : $n \;\; m$
- line2 : $a[0] \;\; a[1] \;\; \ldots \;\; a[n-1]$
- line3 : $r[0] \;\; r[1] \;\; \ldots \;\; r[n-1]$
- line$4+i$: (من أجل $0 \leq i \leq m-1$) $u[i] \;\; v[i]$

The sample grader prints the return value of `who_wins` in the following format:

- line1 : $w[0] \;\; w[1] \;\; \ldots \;\; w[n-1]$