



Подели ги атракциите

Во Баку постојат n атракции, нумерирани со целите броеви од 0 до $n - 1$. Исто така, постојат и m двонасочни патишта, нумерирани со целите броеви од 0 до $m - 1$. Секој пат поврзува две различни атракции. Може да се патува помеѓу секој пар од атракции користејќи ги патиштата.

Благојче планира да ги посети сите атракции во три дена. Тој веќе одлучил дека сака да посети a атракции во првиот ден, b атракции во вториот ден и c атракции во третиот ден. Според тоа, тој ќе ги подели n -те атракции во три множества: A , B и C , со големини a , b и c , соодветно. Секоја атракција ќе припаѓа точно на едно множество, па $a + b + c = n$.

Благојче посакува да ги најде множествата A , B и C , така што **барем две** од трите множества ќе бидат внатрешно **поврзани**. За едно множество S од атракции се вели дека е поврзано ако е возможно да се патува помеѓу кој било пар од атракции во S користејќи ги патиштата пришто нема да се помине ниту една атракција што не е во S . За една партиција (поделба) на атракции во множества A , B и C се вели дека е **валидна** ако таа ги задоволува условите опишани погоре.

Помогнете му на Благојче да пронајде валидна партиција (поделба) на атракциите (за дадени a , b и c), или пак да определи дека не постои ниту една валидна партиција. Ако постојат повеќе валидни партиции, може да пронајдете која било од нив.

Имплементациски детали

Треба да ја имплементирате следнава процедура:

```
int[] find_split(int n, int a, int b, int c, int[] p, int[] q)
```

- n : бројот на атракции.
- a , b и c : посакуваните големини на множествата A , B и C .
- p и q : низи со должина m , кои што ги содржат крајните точки на патиштата. За секое i ($0 \leq i \leq m - 1$), $p[i]$ и $q[i]$ се двете атракции што се поврзани со патот i .
- Оваа процедура треба да врати низа со должина n . Да ја означиме низата со s . Ако не постои валидна партиција (поделба), s треба да содржи n нули. Во

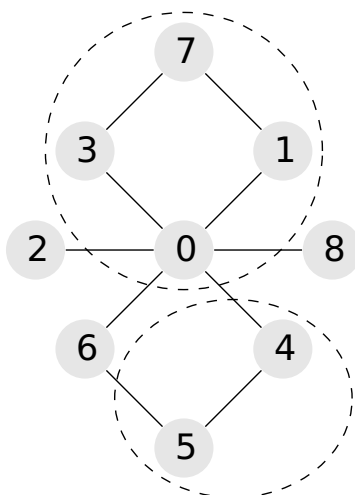
спротивно, за $0 \leq i \leq n - 1$, $s[i]$ треба да биде еднакво на 1, 2 или 3, што означува дека атракцијата i е доделена на множеството A , B , односно C , соодветно.

Примери

Пример 1

Да го разгледаме следниот повик:

```
find_split(9, 4, 2, 3, [0, 0, 0, 0, 0, 0, 1, 3, 4, 5],  
           [1, 2, 3, 4, 6, 8, 7, 7, 5, 6])
```

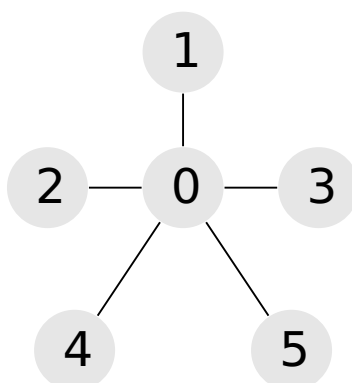


Можно точно решение е $[1, 1, 3, 1, 2, 2, 3, 1, 3]$. Ова решение ја опишува следнава партиција: $A = \{0, 1, 3, 7\}$, $B = \{4, 5\}$ и $C = \{2, 6, 8\}$. Множествата A и B се поврзани.

Пример 2

Да го разгледаме следниот повик:

```
find_split(6, 2, 2, 2, [0, 0, 0, 0, 0], [1, 2, 3, 4, 5])
```



Не постои валидна партиција. Според тоа, единствениот точен одговор е $[0, 0, 0, 0, 0, 0]$.

Ограничувања

- $3 \leq n \leq 100\,000$
- $2 \leq m \leq 200\,000$
- $1 \leq a, b, c \leq n$
- $a + b + c = n$
- Постои најмногу еден пат помеѓу секој пар од атракции.
- Возможно е да се патува помеѓу секој пар од атракции користејќи ги патиштата.
- $0 \leq p[i], q[i] \leq n - 1$ и $p[i] \neq q[i]$ за $0 \leq i \leq m - 1$

Подзадачи

1. (7 поени) Секоја атракција е крајна точка на најмногу два патишта.
2. (11 поени) $a = 1$
3. (22 поени) $m = n - 1$
4. (24 поени) $n \leq 2500, m \leq 5000$
5. (36 поени) Нема дополнителни ограничувања.

Пример-оценувач

Пример-оценувачот ги чита влезните податоци во следниот формат:

- линија 1: $n \ m$
- линија 2: $a \ b \ c$
- линии $3 + i$ (за $0 \leq i \leq m - 1$): $p[i] \ q[i]$

Пример-оценувачот печати една линија која што ја содржи низата вратена од процедурата `find_split`.