



Verdeel de attracties

Er zijn n attracties in Baku, genummerd van 0 tot en met $n - 1$. Er zijn ook m tweerichtingswegen, genummerd van 0 tot en met $m - 1$. Elke weg verbindt twee verschillende attracties. Het is mogelijk om, over de wegen, tussen elk paar attracties te reizen.

Fatima is van plan alle attracties te bezoeken in drie dagen. Ze heeft al besloten dat ze op de eerste dag a attracties wil bezoeken, op de tweede dag b attracties, en op de derde dag c attracties. Daarom wil ze de n attracties in drie sets A , B en C verdelen die respectievelijk a , b en c groot zijn. Elke attractie hoort bij precies één set, dus $a + b + c = n$.

Fatima wil de sets A , B en C vinden zo dat **minstens twee** van de drie sets **verbonden** zijn. Een set S van attracties is verbonden als het mogelijk is om tussen elk paar attracties in S te reizen zonder een attractie te bezoeken die niet in S zit. Een verdeling van attracties in A , B en C is **geldig** als deze aan de hierboven beschreven voorwaarden voldoet.

Help Fatima om, gegeven a , b en c , een geldige verdeling van de attracties te vinden, of bepaal dat er geen geldige verdeling mogelijk is. Als er meerdere mogelijke verdelingen zijn mag je een willekeurige vinden.

Implementatiedetails

Implementeer de volgende procedure:

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

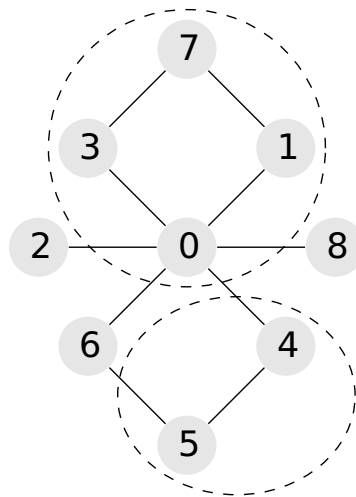
- n : het aantal attracties.
- a , b en c : de gewenste grootte van sets A , B en C .
- p en q : arrays van lengte m met de eindpunten van de wegen. Voor elke i ($0 \leq i \leq m - 1$) zijn $p[i]$ en $q[i]$ de twee attracties die verbonden worden door weg i .
- Deze procedure moet een array van lengte n teruggeven. Noem de array s . Als er geen geldige verdeling is moet s uit n nullen bestaan. Anders moet voor $0 \leq i \leq n - 1$, $s[i]$ een van 1, 2 of 3 zijn om aan te geven dat attractie i in respectievelijk set A , B of C zit.

Voorbeelden

Voorbeeld 1

Kijk naar de volgende aanroep:

```
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])
```

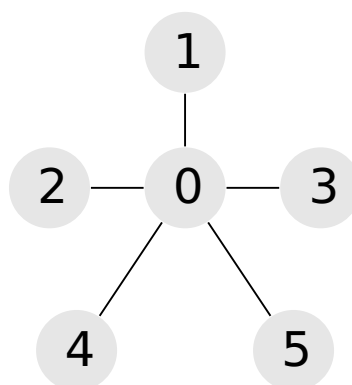


Een mogelijke correcte oplossing is $[1, 1, 3, 1, 2, 2, 3, 1, 3]$. Deze oplossing beschrijft de volgende verdeling: $A = 0, 1, 3, 7$, $B = 4, 5$ en $C = 2, 6, 8$. De sets A en B zijn verbonden.

Voorbeeld 2

Kijk naar de volgende aanroep:

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



Er is geen geldige verdeling. Het enige correcte antwoord is dus $[0, 0, 0, 0, 0, 0]$.

Randvoorwaarden

- $3 \leq n \leq 100\,000$
- $2 \leq m \leq 200\,000$
- $1 \leq a, b, c \leq n$
- $a + b + c = n$
- Er is maximaal één weg tussen ieder paar attracties.
- Het is mogelijk om tussen ieder paar attracties te reizen over de wegen.
- $0 \leq p[i], q[i] \leq n - 1$ en $p[i] \neq q[i]$ voor $0 \leq i \leq m - 1$

Subtaken

1. (7 punten) Elke attractie is een eindpunt van maximaal twee wegen.
2. (11 punten) $a = 1$
3. (22 punten) $m = n - 1$
4. (24 punten) $n \leq 2500, m \leq 5000$
5. (36 punten) Geen aanvullende voorwaarden.

Voorbeeld grader

De voorbeeld grader leest de input in het volgende formaat:

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

De voorbeeld grader schrijft een enkele regel met de array die `find_split` als resultaat heeft gegeven.