



## Podziel atrakcje

W Baku jest  $n$  atrakcji oznaczonych liczbami od 0 do  $n-1$ . Jest także  $m$  dwukierunkowych dróg oznaczonych liczbami od 0 do  $m-1$ . Każda droga łączy dwie różne atrakcje. Pomiędzy każdą parą atrakcji można przemieścić się za pomocą ciągu dróg.

Fatima planuje odwiedzenie wszystkich atrakcji w ciągu trzech dni. Zdecydowała, że odwiedzi  $a$  atrakcji pierwszego dnia,  $b$  atrakcji drugiego dnia, i  $c$  atrakcji trzeciego dnia. W tym celu podzieli  $n$  atrakcji na trzy zbiory  $A$ ,  $B$  oraz  $C$  o rozmiarach (odpowiednio)  $a$ ,  $b$  oraz  $c$ . Każda atrakcja będzie należeć do dokładnie jednego z tych zbiorów, więc  $a + b + c = n$ .

Fatima chciałaby znaleźć zbiory  $A$ ,  $B$  oraz  $C$  o takiej własności, że **przynajmniej dwa** spośród tych trzech zbiorów są **spójne**. Zbiór atrakcji  $S$  jest nazywany spójnym, gdy możliwe jest przemieszczenie się za pomocą dróg pomiędzy każdą parą atrakcji należących do  $S$ , nie przechodząc przy tym przez żadną atrakcję nie należącą do  $S$ . Podział atrakcji na zbiory  $A$ ,  $B$  oraz  $C$  nazywamy **prawidłowym**, gdy spełnia opisane powyżej warunki.

Pomóż Fatimie w znalezieniu prawidłowego podziału atrakcji (dla danych  $a$ ,  $b$  oraz  $c$ ) lub stwierdzeniu, że taki podział nie istnieje. Jeśli istnieje wiele prawidłowych podziałów, możesz znaleźć dowolny z nich.

## Szczegóły implementacyjne

Twoim zadaniem jest zaimplementowanie następującej funkcji:

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

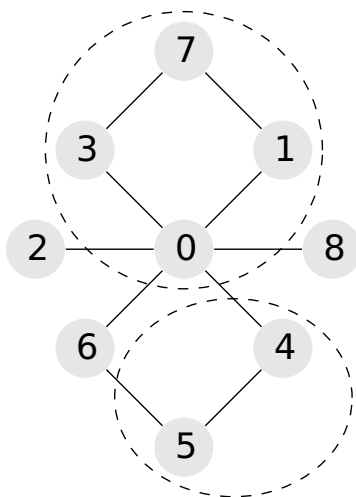
- $n$ : liczba atrakcji.
- $a$ ,  $b$  oraz  $c$ : rozmiary zbiorów  $A$ ,  $B$  oraz  $C$ .
- $p$  oraz  $q$ : tablice długości  $m$ , które zawierają końce dróg. Dla każdego  $i$ ,  $0 \leq i \leq m-1$ ,  $p[i]$  oraz  $q[i]$  są dwiema atrakcjami połączonymi drogą  $i$ .
- Funkcja powinna zwracać tablicę długości  $n$ . Oznaczmy tę tablicę przez  $s$ . Jeśli prawidłowy podział nie istnieje,  $s$  powinna zawierać  $n$  zer. W przeciwnym wypadku, dla każdego  $0 \leq i \leq n-1$ ,  $s[i]$  powinno być równe 1, 2 lub 3 oznaczając, że atrakcja  $i$  należy do (odpowiednio) zbioru  $A$ ,  $B$  lub  $C$ .

# Przykłady

## Przykład 1

Rozważmy następujące wywołanie:

```
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])
```

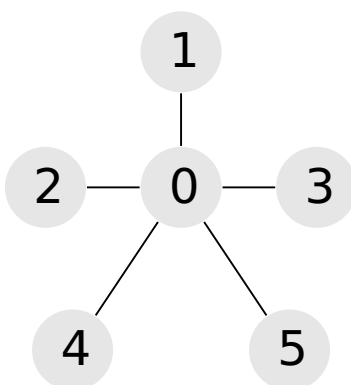


Jedno z możliwych prawidłowych rozwiązań to  $[1, 1, 3, 1, 2, 2, 3, 1, 3]$ . To rozwiązanie opisuje następujący podział:  $A = \{0, 1, 3, 7\}$ ,  $B = \{4, 5\}$  oraz  $C = \{2, 6, 8\}$ . Zbiory  $A$  oraz  $B$  są spójne.

## Przykład 2

Rozważmy następujące wywołanie:

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



Nie ma żadnego prawidłowego podziału. W związku z tym, jedyna prawidłowa odpowiedź to  $[0, 0, 0, 0, 0]$ .

## Ograniczenia

- $3 \leq n \leq 100\,000$
- $2 \leq m \leq 200\,000$
- $1 \leq a, b, c \leq n$
- $a + b + c = n$
- Każda para atrakcji jest połączona bezpośrednio za pomocą co najwyżej jednej drogi.
- Pomiędzy każdą parą atrakcji można przemieścić się za pomocą ciągu dróg.
- $0 \leq p[i], q[i] \leq n - 1$  oraz  $p[i] \neq q[i]$  dla  $0 \leq i \leq m - 1$

## Podzadania

1. (7 punktów) Każda atrakcja jest końcem co najwyżej dwóch dróg.
2. (11 punktów)  $a = 1$
3. (22 punkty)  $m = n - 1$
4. (24 punkty)  $n \leq 2500, m \leq 5000$
5. (36 punktów) Brak dodatkowych ograniczeń.

## Przykładowa sprawdzaczka

Przykładowa sprawdzaczka wczytuje wejście w następującym formacie:

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

Przykładowa sprawdzaczka wypisuje jeden wiersz zawierający tablicę będącą wynikiem wywołania funkcji `find_split`.