



ატრაქციონების დაყოფა

ბაქოში n რაოდენობის ატრაქციონია, რომლებიც გადანომრილია 0-დან $(n - 1)$ -მდე. ისინი შეერთებულია 0-დან $(m - 1)$ -მდე გადანომრილი m რაოდენობის ორმხრივი გზით. ყოველი გზა ორ განსხვავებულ ატრაქციონს აერთებს. გზების საშუალებით შეგვიძლია მგზავრობა ატრაქციონთა ნებისმიერ წყვილს შორის.

ფატიმა გეგმავს სამ დღეში მოინახულოს ყველა ატრაქციონი. მან გადაწყვიტა, რომ პირველ დღეს მოინახულოს a ატრაქციონი, მეორე დღეს - b ატრაქციონი და მესამე დღეს - c ატრაქციონი. ამის გამო ფატიმა აპირებს გაჰყოს n ატრაქციონი სამ A , B და C სიმრავლედ ისე, რომ მათი ზომები შესაბამისად a , b და c იყოს. ყოველი ატრაქციონი უნდა შედიოდეს ზუსტად ერთ სიმრავლეში. მაშასადამე, $a + b + c = n$.

ფატიმას სურს მოძებნოს ისეთი A , B და C სიმრავლეები, რომ სამი სიმრავლიდან **მინიმუმ ორი** იყოს **ბმული**. S სიმრავლე ითვლება ბმულად, თუ მასში შემავალი ატრაქციონების ნებისმიერი წყვილისათვის შესაძლებელია ამ წყვილს შორის მგზავრობა ისე, რომ გამოყენებული იყოს მხოლოდ S -ში შემავალი ატრაქციონები და არ იყოს გამოყენებული ატრაქციონები, რომლებიც არ შედიან S -ში. ატრაქციონების დაყოფას A , B და C სიმრავლეებად უწოდებენ **ვარგისს**, თუ ის აკმაყოფილებს ზემოთ აღწერილ პირობებს.

დაეხმარეთ ფატიმას მოძებნოს ატრაქციონთა ვარგისი გაყოფა (მოცემული a , b და c სიმრავლეებისათვის), ან განსაზღვროს, რომ ვარგისი გაყოფა არ არსებობს.

იმპლემენტაცია

თქვენ უნდა მოახდინოთ შემდეგი ფუნქციის იმპლემენტაცია:

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

- n : ატრაქციონთა რაოდენობა.
- a , b და c : A , B და C სიმრავლეების სასურველი ზომები.
- p და q : m სივრცის მასივები, რომლებიც შეიცავენ გზების ბოლო წერტილების ჩამონათვალს. ყოველი i -სათვის ($0 \leq i \leq m - 1$), $p[i]$ და $q[i]$ წარმოადგენენ ორ ატრაქციონს, რომელიც შეერთებულია i -ური გზით.
- ამ ფუნქციამ უნდა დააბრუნოს n სივრცის მასივი. აღვნიშნოთ ეს მასივი s -ით. თუ ვარგისი დაყოფა არ არსებობს, s უნდა შეიცავდეს n ცალ ნულს. სხვა შემთხვევაში, ნებისმიერი ($0 \leq i \leq n - 1$)-სათვის, $s[i]$ ტოლი უნდა იყოს ერთ-ერთის რიცხვებიდან 1, 2 ან 3, რაც აღნიშნავს იმას, რომ i -ური ატრაქციონი

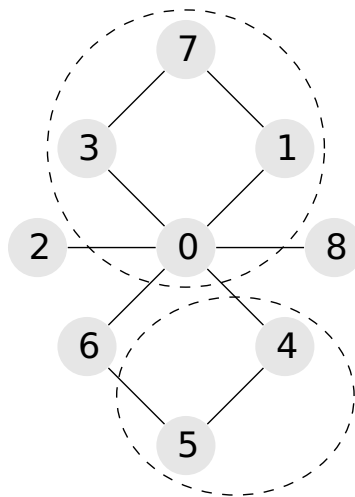
მიკუთვნება A -ს, B -ს ან C -ს შესაბამისად.

მაგალითები

მაგალითი 1

განვიხილოთ შემდეგი გამოცახება:

```
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])
```

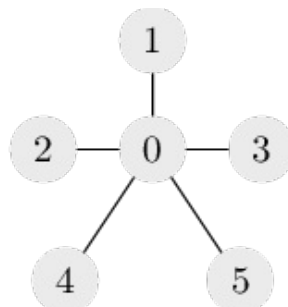


შესაძლო კორექტული ამოხსნაა $[1, 1, 3, 1, 2, 2, 3, 1, 3]$. ეს ამოხსნა აღწერს შემდეგ დაყოფას: $A = 0, 1, 3, 7$, $B = 4, 5$ და $C = 2, 6, 8$. A და B სიმრავლეები ბმულია.

მაგალითი 2

განვიხილოთ შემდეგი გამოცახება:

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



ვარგისი დაყოფა არ არსებობს. მაშასადამე ერთადერთი კორექტული პასუხია $[0, 0, 0, 0, 0]$.

შეზღუდვები

- $3 \leq n \leq 100\,000$
- $2 \leq m \leq 200\,000$
- $1 \leq a, b, c \leq n$
- $a + b + c = n$
- ატრაქციონთა ყოველი წყვილისათვის არაუმეტეს ერთი გზისა მათ შორის.
- ატრაქციონთა ყოველი წყვილისათვის შესაძლებელია მათ შორის მგზავრობა.
- $0 \leq p[i], q[i] \leq n - 1$ და $p[i] \neq q[i]$ ($0 \leq i \leq m - 1$)

ქვეამოცანები

1. (7 ქულა) ყოველი ატრაქციონი წარმოაგენს მაქსიმუმ ორი გზის ბოლო წერტილს.
2. (11 ქულა) $a = 1$
3. (22 ქულა) $m = n - 1$
4. (24 ქულა) $n \leq 2500, m \leq 5000$
5. (36 ქულა) დამატებითი შეზღუდვების გარეშე.

სანიმუშო გრაფერი

სანიმუშო გრაფერი კითხულობს შესატან მონაცემებს შემდეგი ფორმატით:

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

სანიმუშო გრაფერს გამოაქვს ერთადერთი სტრიქონი, რომელიც შეიცავს `find_split-` ის მიერ დაბრუნებულ მასივს.