

თამაში

n ცალი პლანეტის აღმოჩენის შემდეგ, რომლებიც გადანომრილია 0-დან $(n - 1)$ -მდე, ფარაონებმა დაიწყეს მათ შორის სატრანსპორტო სისტემის აგება **ცალმხრივი ტელეპორტერების** საშუალებით. ყოველ ტელეპორტერს აქვს საწყისი და საბოლოო პლანეტა. როცა ტურისტი იყენებს ტელეპორტერს საწყის პლანეტაზე, ის ტელეპორტირდება საბოლოო პლანეტაზე. შევნიშნოთ, რომ საწყისი და საბოლოო პლანეტები ტელეპორტის დროს შეიძლება ემთხვეოდნენ ერთმანეთს. ტელეპორტერი საწყისი u პლანეტით და საბოლოო v პლანეტით აღინიშნება როგორც (u, v) .

ფარაონებს სურთ, რომ წაახალისონ ტელეპორტაციის გამოყენება, ამიტომ მათ შექმნეს თამაში, რომელსაც ითამაშებენ ტურისტები სატრანსპორტო სისტემაში მოგზაურობის დროს. ტურისტს შეუძლია დაიწყოს ნებისმიერი პლანეტიდან. პლანეტები $0, 1, \dots, k - 1$ ($k \leq n$) წარმოადგენენ **სპეციალურ პლანეტებს** და თუ ტურისტი მოხვდება ასეთ პლანეტაზე, ის ჯილდოდ საფოსტო მარკას მიიღებს.

მოცემულ მომენტში ყოველი i -სთვის ($0 \leq i \leq k - 2$), არსებობს $(i, i + 1)$ ტელეპორტერი. ამ $k - 1$ ტელეპორტერს უწოდებენ **საწყის ტელეპორტერებს**.

ახალი ტელეპორტერები ემატება რიგრიგობით. რომელიმე მათგანის დამატების შემდეგ ტურისტმა შეიძლება მიიღოს მარკების უსასრულო რაოდენობის მიღების შესაძლებლობა. უფრო ზუსტად რომ ვთქვათ, ეს ხდება მაშინ, როცა პლანეტების $w[0], w[1], \dots, w[t]$ თანმიმდევრობა აკმაყოფილებს შემდეგ პირობებს:

- $1 \leq t$
- $0 \leq w[0] \leq k - 1$
- $w[t] = w[0]$
- ყოველი i ($0 \leq i \leq t - 1$), არსებობს ტელეპორტერი $(w[i], w[i + 1])$.

მიაქციეთ ყურადღება, რომ ტურისტს შეუძლია გამოიყენოს სასტარტო ტელეპორტერები და **ნებისმიერი** ტელეპორტერი, რომელიც მოცემულ მომენტამდე იქნა დამატებული. თქვენი ამოცანაა, ყოველი ტელეპორტერის დამატების შემდეგ დაეხმაროთ ფარაონებს იმის შემოწმებაში, ხომ არ შეუძლია ტურისტს მიიღოს უსასრულო რაოდენობის მარკა.

იმპლემენტაციის დეტალები

თქვენ უნდა მოახდინოთ შემდეგი პროცედურის იმპლემენტაცია:

```
init(int n, int k)
```

- n : პლანეტების რაოდენობა.
- k : სპეციალური პლანეტების რაოდენობა.
- ეს პროცედურა გამოცხადებული იქნება მხოლოდ ერთხელ, პროცედურა `add_teleporter`-ის ნებისმიერ გამოცხადებამდე.

```
int add_teleporter(int u, int v)
```

- u და v : დამატებული ტელეპორტერის სანყისი და საბოლოო პლანეტები.
- ეს ფუნქცია გამოცხადებული იქნება არაუმეტეს m -ჯერ (m -ის მნიშვნელობები იხილეთ შეზღუდვებში).
- ამ ფუნქციამ უნდა დააბრუნოს 1, თუ (u, v) ტელეპორტერის დამატების შემდეგ ტურისტს შეუძლია უსასრულო რაოდენობის მარკის მიღება. სხვა შემთხვევაში ფუნქციამ უნდა დააბრუნოს 0.
- თუ ფუნქცია დააბრუნებს 1-ს, თქვენმა პროგრამა დაასრულებს მუშაობას.

მაგალითები

მაგალითი 1

განვიხილოთ შემდეგი გამოცხადება:

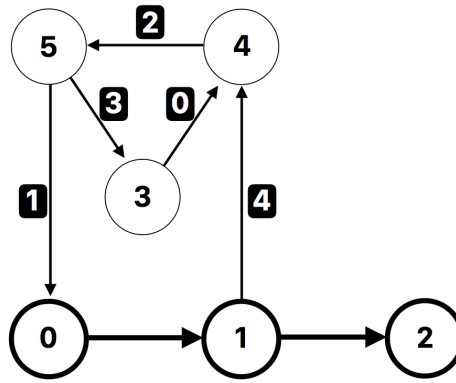
```
init(6, 3)
```

ამ მაგალითში მოცემულია 6 პლანეტა და 3 სპეციალური პლანეტა. 0, 1, და 2 სპეციალური პლანეტებია. სანყისი ტელეპორტერები არის (0, 1) და (1, 2).

Suppose that the grader calls:

- (0) `add_teleporter(3, 4)`: თქვენ უნდა დააბრუნოთ 0.
- (1) `add_teleporter(5, 0)`: თქვენ უნდა დააბრუნოთ 0.
- (2) `add_teleporter(4, 5)`: თქვენ უნდა დააბრუნოთ 0.
- (3) `add_teleporter(5, 3)`: თქვენ უნდა დააბრუნოთ 0.
- (4) `add_teleporter(1, 4)`: ამ მომენტიდან შესაძლებელია მივიღოთ მარკების უსასრულო რაოდენობა. მაგალითად, თუ ტურისტი დაიწყებს 0 პლანეტიდან და შემოივლის პლანეტებს 1, 4, 5, 0, 1, 4, 5, 0, ... თანმიმდევრობით. შესაბამისად, თქვენ უნდა დააბრუნოთ 1, და თქვენი პროგრამა შეწყვეტს მუშაობას.

ქვემოთ მოცემული ნახატი აღწერს ამ მაგალითს. სპეციალური პლანეტები და სასტარტო ტელეპორტერები გამუქებულია. `add_teleporter`-ის მიერ დამატებული ტელეპორტერები მონიშნულია 0-დან 4-მდე, თანმიმდევრულად.



მაგალითი 2

განვიხილოთ შემდეგი გამოცხება:

```
init(4, 2)
```

ამ მაგალითში მოცემულია 4 პლანეტა და 2 სპეციალური პლანეტა. 0 და 1 სპეციალური პლანეტებია. საწყისი ტელეპორტერია (0, 1).

დავუშვათ, გრადერმა მოახდინა შემდეგი გამოცხება:

- `add_teleporter(1, 1)`: (1,1) ტელეპორტერის დამატების შემდეგ შესაძლებელია მარკების უსასრულო რაოდენობის მიღება. ტურისტი სტარტს აიღებს 1 პლანეტიდან და უსასრულო რაოდენობით შევა 1 პლანეტაზე (1,1) ტელეპორტერის საშუალებით. შესაბამისად, თქვენ უნდა დააბრუნოთ 1 და თქვენმა პროგრამამ უნდა დაასრულოს მუშაობა.

თანდართულ ფაილში არის შეტანა-გამოტანის სხვა მაგალითიც.

შეზღუდვები

- $1 \leq n \leq 300\,000$
- $1 \leq m \leq 500\,000$
- $1 \leq k \leq n$

`add_teleporter` პროცედურის ყოველი გამოცხებისათვის:

- $0 \leq u \leq n - 1$ და $0 \leq v \leq n - 1$
- არ არსებობს ტელეპორტერი u პლანეტიდან v პლანეტამდე (u, v) ტელეპორტერის დამატებამდე.

ქვეამოცანები

1. (2 ქულა) $n = k, n \leq 100, m \leq 300$
2. (10 ქულა) $n \leq 100, m \leq 300$
3. (18 ქულა) $n \leq 1\,000, m \leq 5\,000$

4. (30 ქულა) $n \leq 30\,000$, $m \leq 50\,000$, $k \leq 1\,000$

5. (40 ქულა) დამატებითი შეზღუდვის გარეშე

სანიმუშო გრადერი

სანიმუშო გრადერი კითხულობს შესატან მონაცემებს შემდეგი ფორმატით:

- სტრიქონი 1: $n\ m\ k$
- სტრიქონი $2 + i$ ($0 \leq i \leq m - 1$): $u[i]\ v[i]$

სანიმუშო გრადერი ჯერ იძახებს `init`-ს და შემდეგ `add_teleporter`-ს, for $u = u[i]$ და $v = v[i]$ $i = 0, 1, \dots, m - 1$ -თვის თანმიმდევრულად.

მას გამოაქვს `add_teleporter`-ის პირველი გამოძახების ინდექსი, რომელიც აბრუნებს 1-ს (მოთავსებულია 0-სა და $(m - 1)$ -ს შორის მათი ჩათვლით), ან m -ს, თუ `add_teleporter`-ის ყველა გამოძახებამ დააბრუნა 0.

თუ `add_teleporter`-მა დააბრუნა რაიმე მთელი რიცხვი 0-ის ან 1-ის გარდა, სანიმუშო გრადერი დაბეჭდავს -1 -ს და თქვენი პროგრამა შეწყვეტს მუშაობას.