



Oyun (Game)

Gənc oğlan Jian-Jia tapmacanı xoşlayır. Ondan nə isə soruşduqda, o, birbaşa cavab verməkdənsə oyun oynamağa üstünlük verir. Jian-Jia dostu Mei-Yu ilə görüşür və onunla Tayvanın uçuş xətlərinin şəbəkəsi haqqında söhbət edir. Tayvanda n sayda şəhər var (numbered $0, \dots, n - 1$ kimi nömrələnib) və onların bəziləri bir-birilə uçuş xətləri ilə birləşib. Hər bir uçuş (reys) iki şəhəri birləşdirir və hər iki istiqamətdə ola bilər.

Mei-Yu istənilən iki şəhər arasında təyyarə reysinin olub-olmaması (birbaşa, yaxud dolay) haqda Jian-Jiadan soruşur. Jian-Jia suala açıq cavab vermək istəmir və əvəzində oyun oynamağı təklif edir. Mei-Yu ona belə bir sual verə bilər: “ x və y şəhərləri arasında *birbaşa* təyyarə reysi varmı?” və Jian-Jia belə suallara dərhal cavab verəcək. Mei-Yu cəmi $r = n(n - 1)/2$ sual verməklə hər bir şəhər cütünü haqqında soruşa bilər. Əgər Mei-Yu verə biləcəyi r sualın ilk i saydasına $i < r$ cavab aldıqdan sonra istənilən iki şəhər arasında təyyarə ilə (birbaşa, yaxud dolay) səyahət edə bilməsinin mümkünlüyü haqqında nəticə çıxara bilirsə, o qalib gəlir. Əks halda, yəni o, r sualın hamısını verirsə, qalib Jian-Jia olur.

Qaydalarına görə, oyunun Jian-Jia üçün maraqlı olması üçün dostlar razılaşırlar ki, Jian-Jia Tayvanın mövcud uçuş xətləri şəbəkəsini unuda bilər və o, suallara istədiyi kimi cavab verə bilər. Siz Jian-Jiaya suallara necə cavab verməli olduğunu bildirməklə oyunu udmaqda ona yardım etməlisiniz.

Örnəklər (Examples)

Biz oyun qaydalarını üç nümunə üzərində araşdıracağıq. Hər bir nümunədə $n = 4$ şəhər və $r = 6$ sual-cavab raundu var.

Birinci nümunədə (aşağıdakı cədvəl), Jian-Jia *uduzur*, çünki Mei-Yu 4-cü raunddan sonra, Jian-Jiannın 5 və ya 6-cı suallara necə cavab verəcəyindən asılı olmayaraq, istənilən iki şəhər arasında təyyarə ilə səyahət edə biləcəyinə əmin olur.

raund	sual	cavab
1	0, 1	yes
2	3, 0	yes
3	1, 2	no
4	0, 2	yes
-----	-----	-----
5	3, 1	no
6	2, 3	no

İNövbəti nümunədə Mei-Yu 3-cü raunddan sonra, Jian-Jiannın 4, 5 və ya 6-cı suala necə cavab verəcəyindən asılı olmayaraq, sübut edə bilər ki, kimsə 0 və 1 şəhərləri arasında təyyarə ilə uça *bilməz*, deməli, Jian-Jia yenə də uduzur.

round	question	answer
1	0, 3	no
2	2, 0	no
3	0, 1	no
-----	-----	-----
4	1, 2	yes
5	1, 3	yes
6	2, 3	yes

Sonuncu nümunədə Mei-Yu altı sualın hamısına cavab aldıqdan sonra istənilən iki şəhər arasında təyyarə ilə səyahət etməyin mümkün olub-olmamasını müəyyən edə bilmir, deməli, oyunu Jian-Jia *udur*. Çünki, əgər Jian-Jia sonuncu suala *yes* cavabını verərsə (aşağıdakı cədvəldə), onda istənilən iki şəhər arasında uçuş mümkündür. Fəqət, əgər Jian-Jia sonuncu suala *no* cavabını verərsə, uçuş mümkün olmayacaq.

round	question	answer
1	0, 3	no
2	1, 0	yes
3	0, 2	no
4	3, 1	yes
5	1, 2	no
6	2, 3	yes

Task

Please write a program that helps Jian-Jia win the game. Note that neither Mei-Yu nor Jian-Jia knows the strategy of each other. Mei-Yu can ask about pairs of cities in any order, and Jian-Jia must answer them immediately without knowing the future questions. You need to implement the following two functions.

- `initialize(n)` -- We will call your `initialize` first. The parameter n is the number of cities.
- `hasEdge(u, v)` -- Then we will call `hasEdge` for $r = n(n - 1)/2$ times. These calls represent Mei-Yu's questions, in the order that she asks them. You must answer whether there is a direct flight between cities u and v . Specifically, the return value should be 1 if there is a direct flight, or 0 otherwise.

Subtasks

subtask	points	n
1	15	$n = 4$

subtask	points	n
2	27	$4 \leq n \leq 80$
3	58	$4 \leq n \leq 1500$

Implementation details

You have to submit exactly one file, called `game.c`, `game.cpp` or `game.pas`. This file implements the subprograms described above using the following signatures.

C/C++ programs

```
void initialize(int n);  
int hasEdge(int u, int v);
```

Pascal programs

```
procedure initialize(n: longint);  
function hasEdge(u, v: longint): longint;
```

Sample grader

The sample grader reads the input in the following format:

- line 1: n
- the following r lines: each line contains two integers u and v that describe a question regarding cities u and v .