



## اللعبة

جيان جيا شاب يافع يحب اللعب بالألعاب، عندما يوجه إليه سؤال ما، يفضل جيان أن يلعب لعبة على أن يجابوب بشكل مباشر. التقى جيان جيا بصديقه مي-يو وأخبرها عن شبكة الطيران في تايوان. هناك  $n$  مدينة في تايوان (مرقمة من 0 وحتى  $n - 1$ )، بعض هذه المدن مربوطة مع بعضها برحلات طيران. كل رحلة تربط مدينتين ويمكن أن تستخدم الرحلة بالاتجاهين (ذهاباً وإياباً).

سألت مي-يو جيان-جيا فيما إذا كان من الممكن الذهاب بين أي مدينتين بالطائرة (سواءً مباشرة أو بشكل غير مباشر). لم يرغب جيان بالبوح بالجواب، و عوضاً عن ذلك اقترح أن يلعبوا لعبة. يمكن لمي-يو أن تسأله أسئلة من الشكل "هل المدينتان  $x$  و  $y$  مربوطتان بشكل مباشر برحلة طيران"، سيفهم جيان بالإجابة على هذه الأسئلة مباشرة. ستسأل مي-يو عن كل زوج من المدن مرة واحدة لا أكثر ولا أقل. وهذا ما سيعطي ما مجموعه  $r = n(n - 1)/2$  سؤالاً. تريح مي-يو اللعبة إذا استطاعت استنتاج فيما إذا كانت الشبكة متصلة بشكل كلي أم لا بعد الحصول على إجابة  $i$  سؤالاً من إجمالي العدد الكلي للأسئلة حيث قيمة  $i < r$  أي هل من الممكن السفر بين كل زوج من المدن بالركوب برحلات الطيران (سواءً مباشرة أو بشكل غير مباشر). وفي حال كانت مي-يو تحتاج أن تسأل كل الأسئلة  $r$  لتعرف فيما إذا كانت الشبكة متصلة أم لا عندئذ يكون الفائز هو جيان.

حيث يقوم جيان-جيا باختبار أجوبته اعتماداً على أسئلة مي-يو السابقة، مهمتك هي مساعدة جيان على الفوز في اللعبة عن طريق تحديد كيفية إجابته عن الأسئلة.

## مثال

سنقوم بشرح قواعد اللعبة بثلاثة أمثلة، في كل الأمثلة سيكون عدد المدن  $n = 4$  وعدد جولات الأسئلة والأجوبة هو  $r = 6$  من المعادلة اعلاه.

في المثال الأول (الجدول التالي) □□□□ لأنه بعد الجولة الرابعة يمكن لمي-يو معرفة أنه من الممكن السفر بين أي مدينتين بالطائرة مهما كانت إجابة جيان عن السؤالين 5 و 6.

الجولة	السؤال	الجواب
1	0, 1	yes
2	3, 0	yes
3	1, 2	no
4	0, 2	yes
-----	-----	-----
5	3, 1	no
6	2, 3	no

في المثال التالي مي-يو يمكنها أن تثبت أنه تثبت أنه بعد الجولة الثالثة ومهما كانت إجابات جوان عن الأسئلة 4 و 5 و 6 لا يمكن السفر بين المدينتين 0 و 1 باستخدام الرحلات لذلك يخسر جوان أيضاً

الإجابة	السؤال	الجولة
---------	--------	--------

الجولة	السؤال	الإجابة
1	0, 3	no
2	2, 0	no
3	0, 1	no
-----	-----	-----
4	1, 2	yes
5	1, 3	yes
6	2, 3	yes

في المثال الأخير لا يمكن لمي-يو تحديد فيما إذا كان يمكن السفر بين أي مدينتين أم لا إلا بعد انتهاء كل الأسئلة الستة والإجابة عليها، لذلك فإن جيان يربح اللعبة. وخصوصاً أن جوان أجاب بـ "نعم" على آخر سؤال (لاحظ الجدول أدناه) وهذا ما يعني أن الشبكة متصلة ويمكن السفر بين أي مدينتين بالطائرة، أما لو كان أجاب بـ "لا" على آخر سؤال لكانت الشبكة غير متصلة.

الجولة	السؤال	الإجابة
1	0, 3	no
2	1, 0	yes
3	0, 2	no
4	3, 1	yes
5	1, 2	no
6	2, 3	yes

## المهمة Task

قم بكتابة برنامج يساعد جيان جيا على الفوز باللعبة، لاحظ أن كلاً من جيان ومي-يو لا يعرفان الاستراتيجية التي يلعب بها الطرف الآخر، كما أنه من الممكن أن تسأل مي-يو عن أزواج المدن بأي ترتيب، ويجب على جوان الإجابة على الأسئلة فوراً بدون أن يعرف ما هي الأسئلة التالية، يجب عليك تحقيق التابعين التاليين:

■ initialize(n) -- سنقوم باستدعاء الدالة الجاهزة initialize أولاً لكي نتحصل منها على  $n$  وهو عدد المدن.

■ hasEdge(u, v) -- ثم سنقوم باستدعاء hasEdge لعدد  $r = n(n - 1)/2$  مرة. تمثل هذه الاستدعاءات أسئلة مي-يو، بالترتيب التي طرحتهم به. يجب عليك الإجابة فيما إذا كان هناك رحلة مباشرة بين المدينة  $u$  و  $v$ . بشكل أوضح، القيمة المسترجعة من الدالة يجب أن تكون 1 إذا كان هناك رحلة و 0 في حال عكس ذلك..

## المهام الجزئية Subtasks

كل مهمة جزئية تتألف من عدة ألعاب، ستأخذ النقاط على كل مهمة إذا ربح برنامجك بكل الألعاب من أجل جيان-جيا.

المهمة الجزئية	النقاط	$n$
1	15	$n = 4$
2	27	$4 \leq n \leq 80$
3	58	$4 \leq n \leq 1500$

## تفاصيل التنفيذ

يجب عليك إرسال ملف واحد فقط اسمه game.cpp لمنظومة التصحيح

## 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;
```

## منظومة التصحيح

تقرأ منظومة التصحيح البيانات وفق الصيغة التالية:

line 1:  $n$  ■

the following  $r$  lines: each line contains two integers  $u$  and  $v$  that describe a question regarding ■  
 $v$  cities  $u$  and