

## 1. Đồ thị (ADDEGE.\*)

Người ta khởi tạo một đồ thị có hướng gồm  $10^9$  đỉnh, các đỉnh được đánh số từ 1 đến  $10^9$ . Ban đầu đồ thị không có cung nào. Người ta lần lượt thêm các cung vào đồ thị bởi  $m$  lệnh dạng  $Add(u,v)$ : thêm một cung nối từ đỉnh  $u$  đến đỉnh  $v$  trên đồ thị.

Cho trước hai đỉnh  $s$  và  $t$ . Hãy cho biết số thứ tự của lệnh  $Add$  đầu tiên mà sau thời điểm thực hiện lệnh  $Add$  đó, ta có thể đi từ  $s$  đến  $t$  theo các cung của đồ thị

*Input:*

- Dòng 1 chứa ba số nguyên dương  $m, s, t$  ( $m \leq 10^5; s \neq t$ )
- $m$  dòng tiếp theo, mỗi dòng ghi hai số nguyên  $u, v$  tương ứng là một lệnh  $Add(u,v)$

*Output:* Một số duy nhất là số thứ tự lệnh  $Add$  tìm được, trong trường hợp không thể đi từ  $s$  đến  $t$  cho dù thực hiện tất cả các lệnh  $Add$  thì ghi số 0.

*Example:*

input	output
51 5	4
1 2	
3 5	
3 1	
2 3	
2 4	

## 2. Điệp viên (SPY.\*)

Địa bàn hoạt động của một điệp viên là một khu phố mà ở đó chỉ có các đường phố ngang, dọc tạo thành một lưới ô vuông. Với mục đích bảo mật, thay vì tên đường phố, điệp viên đánh số các phố ngang từ 0 đến  $m$  và các phố dọc từ 0 đến  $n$ . ở một số ngã ba hoặc ngã tư có các trạm kiểm soát. Anh ta đang đứng ở nút giao của hai đường  $(i_1, j_1)$  ( $j_1$  - đường ngang;  $i_1$  - đường dọc) và cần tới điểm hẹn ở giao của hai đường  $(i_2, j_2)$ . Để tránh bị theo dõi, đường đi phải không qua các trạm kiểm soát và cứ tới chỗ rẽ thì nhất thiết phải đổi hướng đi, thậm chí có thể sang đường và đi ngược trở lại. Việc đổi hướng chỉ được thực hiện ở ngã ba hoặc ngã tư. Hãy xác định đường đi ngắn nhất tới điểm hẹn hoặc cho biết không có đường đi đáp ứng được yêu cầu đã nêu.

*Input:*

- Dòng đầu:  $m \ n \ i_1 \ j_1 \ i_2 \ j_2$  ( $0 \leq m, n \leq 100$ )
- Các dòng sau: mỗi dòng 2 số  $i, j$  (tọa độ trạm kiểm soát).

*Output:*

- Dòng đầu: độ dài đường đi ngắn nhất hoặc thông báo NO nếu không có đường đi.
- Các dòng sau: mỗi dòng 2 số  $i, j$  chỉ nút tiếp theo cần tới theo đường đi tìm được, bắt đầu là  $i_1 \ j_1$  và kết thúc là  $i_2 \ j_2$ .

*Example:*

Input	Output
4 5 0 0 5 4	13
0 1	0 0
0 4	1 0
2 2	1 1
2 3	1 0

4 0	2 0
5 2	2 1
5 3	3 1
	3 2
	4 2
	4 3
	3 3
	4 3
	4 4
	5 4

### 3. Di chuyển robot (RMOVE.\*)

Cho một đồ thị có hướng gồm  $n$  đỉnh và  $m$  cung, hai con robot đứng ở đỉnh 1 và  $n$ . Hãy tìm cách di chuyển nhanh nhất hai con robot đến gặp nhau tại một đỉnh của đồ thị, biết rằng cả hai con robot chỉ được chạy theo các cung định hướng và không được dừng lại cho tới lúc gặp nhau tại một đỉnh nào đó. Thời gian robot đi qua một cung bất kỳ luôn là 1 đơn vị thời gian

*Input:*

- Dòng 1: Chứa hai số nguyên dương  $n \leq 1000; m \leq 5000$
- $m$  dòng tiếp theo mỗi dòng chứa hai số nguyên  $u, v$  tương ứng với cung  $(u, v)$  của đồ thị.

*Output:*

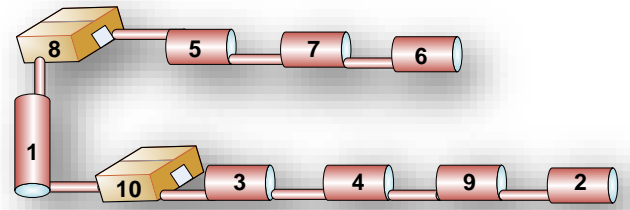
- Ghi thời gian tính từ lúc bắt đầu di chuyển đến khi hai robot gặp nhau (nếu không có phương án thì ghi -1)
- Trong trường hợp có phương án thực hiện thì dòng thứ hai ghi hành trình của robot thứ nhất theo thứ tự từ đỉnh 1 đến đỉnh gặp nhau và dòng thứ ba ghi hành trình của robot thứ hai với qui cách tương tự.

*Example:*

Input	Output
2 2 1 2 2 1	-1
4 4 1 3 3 2 4 1 1 2	2 1 3 2 4 1 2

#### 4. Sơ tán (EVA.\*)

Một Trung tâm nghiên cứu tuyệt mật (mà chúng ta không có quyền nói rõ tên ở đây) có  $n$  phòng thí nghiệm đặt ngầm trong lòng đất. Các phòng thí nghiệm được đánh số từ 1 đến  $n$  ( $1 \leq n \leq 10^5$ ). Giữa một số phòng có đường hầm nối với nhau, sao cho từ một phòng bất kỳ có thể đi đến phòng bất kỳ khác (có thể phải đi qua một số phòng nào đó). Độ dài mỗi đường hầm là như nhau và thời gian đi hết một đường hầm là 1. Không có đường hầm nào nối một phòng với chính nó, nhưng có thể có nhiều đường hầm cùng nối 2 phòng với nhau và tổng cộng trong Trung tâm có tất cả  $m$  đường hầm ( $1 \leq m \leq 10^5$ ). Đường hầm cho phép đi lại theo cả hai chiều. Có  $k$  phòng có lối thoát hiểm lên mặt đất ( $1 \leq k \leq n$ ). Trong trường hợp sơ tán khẩn cấp, tất cả các nhân viên phải tập trung ở những phòng có lối thoát hiểm.



**Yêu cầu:** Hãy xác định thời gian tối thiểu để nhân viên mỗi phòng tập trung về phòng có lối thoát hiểm trong trường hợp phải sơ tán khẩn cấp.

**Input:**

- Dòng đầu tiên chứa 2 số nguyên  $n$  và  $k$ ,
- Dòng thứ 2 chứa  $k$  số nguyên khác nhau cho biết các phòng có cửa thoát hiểm,
- Dòng thứ 3 chứa số nguyên  $m$ ,
- Mỗi dòng trong  $m$  dòng tiếp theo chứa 2 số nguyên xác định cặp phòng có đường hầm nối trực tiếp.

**Output:** Một dòng chứa  $n$  số nguyên, số thứ  $i$  xác định thời gian tối thiểu để nhân viên phòng  $i$  đi được tới phòng có lối thoát hiểm.

**Example:**

Input	Output
10	1 4 1 2 1 3 2 0 3 0
2	
10 8	
9	
6 7	
7 5	
5 8	
8 1	
1 10	
10 3	
3 4	
4 9	
9 2	

#### 5. Tìm khớp và cầu (DFSGRAPH.\*)

Xét đơn đồ thị vô hướng  $G = (V, E)$  có  $n$  ( $1 \leq n \leq 3 \cdot 10^5$ ) đỉnh, các đỉnh đánh số từ 1 đến  $n$  và  $m$  ( $1 \leq m \leq 5 \cdot 10^5$ ) cạnh. Người ta định nghĩa một đỉnh là khớp nếu như bỏ đỉnh đó

sẽ làm tăng số thành phần liên thông của đồ thị. Tương tự, một cạnh được gọi là cạnh cầu nếu xóa cạnh đó sẽ làm tăng số thành phần liên thông của đồ thị. Vấn đề đặt ra là đếm tất cả các đỉnh khớp và cạnh cầu của đồ thị.

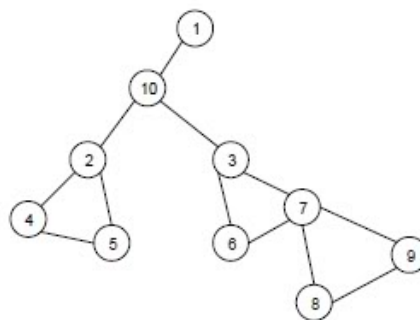
*Input:*

- Dòng đầu chứa hai số tự nhiên  $n, m$
- $m$  dòng tiếp theo, mỗi dòng chứa hai số nguyên  $u, v$  thể hiện có một cạnh nối đỉnh  $u$  với đỉnh  $v$  ( $1 \leq u, v \leq n; u \neq v$ )

*Output:* Gồm một dòng duy nhất chứa hai số nguyên, số thứ nhất - số lượng đỉnh khớp và số thứ hai - số lượng cạnh cầu của đồ thị

*Example:*

Input	Output
1 10 10 2 10 3 2 4 4 5 5 2 3 6 6 7 7 3 7 8 8 9 9 7	4 3



## 6. Thành phố trung tâm (CAPITAL.\*)

Ở đất nước ByteLand có  $n$  thành phố (đánh số thứ tự từ 1 đến  $n$ ) được nối với nhau bằng  $m$  con đường hai chiều). Do yêu cầu về sự thịnh vượng của đất nước, cần thiết phải di chuyển thủ đô hiện thời về một thành phố nào đó trong số các thành phố nói trên. Quốc hội ByteLand muốn chọn một thành phố trung tâm nhất để di chuyển thủ đô đến đó.

Thành phố được gọi là trung tâm nếu như khi cấm các hoạt động giao thông qua thành phố này và trên các con đường nối với nó thì các thành phố còn lại tạo thành nhiều cụm thành phố nhất. Một cụm thành phố là một nhóm các thành phố mà giữa hai thành phố bất kỳ có thể đi đến được với nhau thông qua các thành phố trong cụm. Tất nhiên, nếu thêm một thành phố khác vào cụm thì tính chất trên bị phá vỡ.

Viết chương trình tìm thành phố trung tâm cho Quốc Hội

*Input:*

- Dòng đầu tiên ghi hai số nguyên dương  $n, m$  ( $2 \leq n, m \leq 300000$ )
- $m$  dòng tiếp theo, mỗi dòng ghi hai số  $a_i, b_i$  thể hiện có một đường đi hai chiều nối thành phố  $a_i$  với thành phố  $b_i$ .

*Output:* Ghi hai số  $C_{max}$  và  $C$  thể hiện số cụm lớn nhất tạo ra khi cấm đường giao thông qua thành phố trung tâm và số hiệu của thành phố trung tâm này. Nếu có nhiều thành phố trung tâm thì chỉ in ra thành phố có chỉ số nhỏ nhất

*Example:*

Input	Output
-------	--------

5 5 1 2 3 2 4 2 5 2 1 3	3 2
--	-----

## 7. Mạng máy tính (MNET.\*)

Một hệ thống  $N$  máy tính đánh số từ 1 đến  $N$  được nối mạng bằng  $M$  kênh truyền tin một chiều giữa một số cặp máy. Mạng cho gọi là thông suốt nếu như từ một máy  $u$  bất kỳ luôn có thể truyền tin đến mỗi máy  $v$  trong số các máy còn lại hoặc theo kênh truyền tin từ  $u$  đến  $v$  hoặc thông qua một số kênh trung gian. Ta gọi một mạng con của mạng đã cho là một mạng gồm một số máy và các kênh nối chúng của mạng đã cho. Trong trường hợp mạng là không thông suốt nó sẽ phân rã thành một số mạng con thông suốt. Mạng con thông suốt được gọi là cực đại nếu như không tồn tại một mạng con thông suốt của mạng đã cho chứa nó như một mạng con. Bạn cần xác định số mạng con thông suốt cực đại của mạng đã cho.

*Input:*

- Dòng đầu tiên chứa hai số nguyên dương  $N, M$  ( $1 \leq N, M \leq 5 \cdot 10^5$ )
- Dòng thứ  $i$  trong  $M$  dòng tiếp theo ghi 2 số nguyên dương  $d_i, c_i$  cho biết kênh truyền tin thứ  $i$  cho phép truyền tin từ máy  $d_i$  sang máy  $c_i$ .

*Output:*

- Dòng đầu tiên ghi  $K$  là số mạng con thông suốt cực đại của mạng đã cho
- $K$  dòng tiếp theo, mỗi dòng ghi dãy các đỉnh thuộc cùng một mạng con thông suốt cực đại

*Example:*

Input	Output
9 14 1 2 1 4 1 7 2 3 2 6 3 1 4 5 5 4 5 7 6 7 7 9 9 6 9 8 8 9	3 1 2 3 4 5 6 7 8 9