# Ladder game

Có N người chơi ở N cột, có một số đường ngang. Mỗi ng đi từ trên xuống dưới, **nếu gặp một đầu mút của đường ngang thì đi sang đầu mút bên kia, nếu gặp ô trống hoặc điểm giữa đường ngang thì vẫn đi thẳng xuống**.

A graph with a bomb and a line

Description automatically generated

Ví dụ như trên hình có 8 người chơi tương ứng với 8 cột, các đường màu xanh là đường ngang.

Có thể tưởng tượng mỗi cột dọc như con đường dưới đất còn đường ngang như kiểu cầu vượt, nghĩa là khi đi trên đường ngang, dù có đi qua bao nhiêu đường dọc thì cũng k rẽ, chỉ khi đi hết cầu, đến đầu bên kia thì mới rẽ. Lúc đi đường dọc cũng thế, chỉ khi nào gặp 1 trong 2 đầu mút thì mới "lên cầu", còn gặp bất cứ điểm nào ở giữa đường ngang thì vẫn đi thẳng xuống tiếp, như kiểu chui dưới gầm cầu.

A screenshot of a graph

Description automatically generated

Ví dụ đường màu xanh của người chơi số 1, khi kết thúc sẽ ở vị trí số 4, đường màu đỏ của người chơi số 2 sẽ kết thúc ở vị trí 5, và BÙM!

**Trong input đã đảm bảo:**

- Hàng dưới cùng k có đường ngang.

- Quả bom luôn nằm ở hàng dưới cùng

- Trên 1 hàng có thể có nhiều đường ngang, nhưng k bao giờ có chung điểm đầu điểm cuối, vì nếu có chung 1 đầu mút thì sẽ k biết phải rẽ vào đường ngang nào.

- Các chỉ số được đánh số từ 1.

Đề bài sẽ cho số người, vị trí (cột) quả bom và M là số đường ngang, sau đó là M cặp điểm, mỗi cặp điểm là tọa độ điểm đầu và điểm cuối của đường ngang. Đảm bảo chỉ số hàng của 2 đầu mút luôn bằng nhau, ví dụ cặp điểm 1 3 1 5 biểu diễn một đường ngang nối từ hàng 1 cột 3 sang hàng 1 cột 5.

Có cố định 10 input, mỗi input đc viết trên 1 dòng.

Mỗi input bắt đầu bởi 3 số:

- N: số người chơi, cũng là số hàng và số cột của ma trận.

- B: Vị trí quả bom.

- M: Số đường ngang

Sau đó là 4 \* M số nguyên dương chứa tọa độ điểm đầu và điểm cuối của các đường ngang.

Tìm và in ra vị trí của người mà sẽ đi đến quả bom, mỗi test case in trên 1 dòng theo định dạng **“#” + số thứ tự test case + “ “ + vị trí.**

**Ví dụ:**

*Input*:

4 3 4 1 1 1 3 2 1 2 2 2 3 2 4 3 2 3 3

8 5 11 1 3 1 5 2 2 2 3 2 5 2 8 3 4 3 5 3 6 3 7 4 3 4 6 5 1 5 6 6 1 6 2 6 4 6 5 7 2 7 5 7 7 7 8

*Output:*

#1 3

#2 2

# Bài 2: Ladder 1

Cho ma trận NxN như hình dưới:

A yellow and white grid with numbers

Description automatically generated

Các ô số 0 (màu trắng) là **ô trống**, các ô có số 1 (màu vàng) là **đường đi**. Có **duy nhất một ô số 2** tượng trưng cho quả bom. Các cột có toàn số 1 là các đường dọc trong trò chơi, có bao nhiêu cột 1 sẽ tương ứng với từng đấy người chơi.

Giữa các cột có các hàng ngang toàn số 1 biểu thị các đường ngang. Khi đi từ trên xuống theo đường dọc, nếu gặp một hàng ngang thì nhất định phải rẽ ngang, đi đến khi nào hết đường ngang thì lại đi dọc xuống.

Ví dụ: khi đi từ cột 5 hoặc khi đi từ cột 0

A yellow and orange grid with numbers

Description automatically generated A screenshot of a grid

Description automatically generated

Trên hình là đường đi của người ở cột 5 và người ở cột 0, người ở cột 0 sẽ gặp bom.

Tìm và in ra số cột của người mà có con đường dẫn đến quả bom.

*Input:* Có cố định 10 test case.

Dòng đầu tiên gồm 2 số T và N, trong đó

* T: số thứ tự của test case (1 <= T <= 10)
* N: kích thước ma trận (10 <= N <= 100)

Theo sau là nội dung ma trận: N hàng, mỗi hàng N số.

*Output:* mỗi test case in trên 1 dòng: **“#” + test case + “ “ + số cột.**

**Các ràng buộc:**

* Cột từ đầu tiên và cột cuối cùng trong input luôn là cột toàn số 1
* Không bao giờ có 2 cột toàn số 1 đứng cạnh nhau
* Hàng trên cùng và hàng dưới cùng không có đường ngang nào.
* Các chỉ số hàng cột được đếm từ 0.

**Ví dụ:** (giả sử có 2 test case, input thật sẽ luôn có 10 test case)

*Input*:

1 10

1 0 0 1 0 1 0 0 0 1

1 0 0 1 0 1 1 1 1 1

1 1 1 1 0 1 0 0 0 1

1 1 1 1 0 1 0 0 0 1

1 0 0 1 1 1 0 0 0 1

1 0 0 1 0 1 0 0 0 1

1 1 1 1 0 1 1 1 1 1

1 0 0 1 1 1 0 0 0 1

1 0 0 1 0 1 1 1 1 1

1 0 0 1 0 1 0 0 0 2

2 10

1 0 0 1 0 0 1 0 0 1

1 0 0 1 0 0 1 1 1 1

1 1 1 1 0 0 1 0 0 1

1 0 0 1 0 0 1 1 1 1

1 0 0 1 1 1 1 0 0 1

1 0 0 1 0 0 1 0 0 1

1 1 1 1 0 0 1 1 1 1

1 0 0 1 1 1 1 0 0 1

1 1 1 1 0 0 1 1 1 1

1 0 0 2 0 0 1 0 0 1

*Output*:

#1 0

#2 6

# Bài 3: Ladder 2

Gần giống bài 2, chỉ khác là không có quả bom nào cả. Thay vào đó, **con đường đi của người nào mà qua ít bước hơn** thì người đó sẽ phải đi mua kem, nếu có hơn 1 người cùng có số bước ít nhất, **chọn người có số cột cao hơn**.

A yellow and white grid with numbers

Description automatically generatedA yellow and white grid with numbers

Description automatically generated

Với ví dụ trên, số bước của từng người là:

Người ở cột số 0: 25

Người ở cột số 3: 26

Người ở cột số 5: 20

Người ở cột số 9: 19

* *Người ở cột số 9 có số bước ít nhất sẽ phải đi mua.*

*Input:* Có cố định 10 test case.

Dòng đầu tiên gồm 2 số T và N, trong đó

* T: số thứ tự của test case (1 <= T <= 10)
* N: kích thước ma trận (10 <= N <= 100)

Theo sau là nội dung ma trận: N hàng, mỗi hàng N số.

*Output:* mỗi test case in trên 1 dòng: **“#” + test case + “ “ + số cột.**

**Các ràng buộc:**

* Cột đầu tiên và cột cuối cùng trong input luôn là cột toàn số 1
* Không bao giờ có 2 cột toàn số 1 đứng cạnh nhau
* Hàng trên cùng và hàng dưới cùng không có đường ngang nào.
* Các chỉ số hàng cột được đếm từ 0.

**Ví dụ:** (giả sử có 1 test case, input thật sẽ luôn có 10 test case)

*Input*:

1 10

1 0 0 1 0 1 0 0 0 1

1 0 0 1 0 1 1 1 1 1

1 1 1 1 0 1 0 0 0 1

1 1 1 1 0 1 0 0 0 1

1 0 0 1 1 1 0 0 0 1

1 0 0 1 0 1 0 0 0 1

1 1 1 1 0 1 1 1 1 1

1 0 0 1 1 1 0 0 0 1

1 0 0 1 0 1 1 1 1 1

1 0 0 1 0 1 0 0 0 1

2 10

1 0 0 1 0 0 1 0 0 1

1 0 0 1 0 0 1 1 1 1

1 1 1 1 0 0 1 0 0 1

1 0 0 1 0 0 1 1 1 1

1 0 0 1 1 1 1 0 0 1

1 0 0 1 0 0 1 0 0 1

1 1 1 1 0 0 1 1 1 1

1 0 0 1 1 1 1 0 0 1

1 1 1 1 0 0 1 1 1 1

1 0 0 1 0 0 1 0 0 1

*Output:*

#1 9

#2 3