

Lazy tourist guide

Marie-Chantal works as a tourist guide. As such she proposes tours to tourists in her city that pass by some of the available points of interest. In the city one can reach of course any point of interest from any other point of interest, but some for some pair of points it is unreasonable to walk directly from the first to the second point without stopping on the way at some other intermediate points of interest. The city provided Marie-Chantal a list of pairs of points of interest that is reasonable to reach directly. These pairs are called reasonable. If A, B is a reasonable pair then so is B, A .

A reasonable tour is a list of points of interest that are all different except the first and the last one which are equal, and such that any two successive points of interest on the list form a reasonable pair.

She is quite a lazy tourist guide so she wants to find a reasonable tour that visits as little points of interest as possible. Can you help her?

Input

The first line contains a single integer, the number of test cases.

Each test case starts with a line containing two space separated integers N and M , ($1 \leq N \leq 1000$, $1 \leq M \leq 40000$) where N is the number of points of interest, numbered from 1 to N and M is the number of reasonable pairs. The remaining M lines contain each two space separated integers A, B , ($1 \leq A < B \leq N$) describing a reasonable pair of interest. The pairs will be all distinct.

Output

For each test case, numbered k start at 1, output a single line as follows.

If there is no tour output "Case #k: -1". Otherwise output "Case #k: t u1 u2 ... ut", where t is the length of an arbitrary shortest tour and $u1, u2, \dots, ut$, the identifiers of the points of interest along the tour with $u1$ being equal to ut . The sequence $u1, u2, \dots, ut$ should be lexicographically smallest among all reasonable tours of length t .

Example

Input:

```
2
4 3
1 2
1 3
1 4
4 5
1 2
2 3
3 4
1 4
2 4
```

Output:

Case #1: -1

Case #2: 3 1 2 4 1