

House of Kittens

Problem

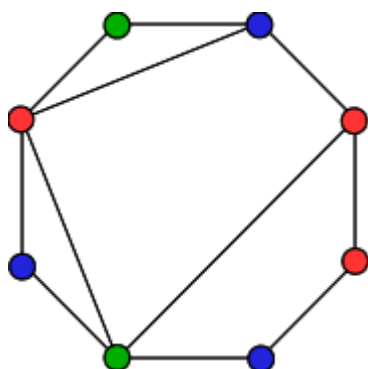
You have recently adopted some kittens, and now you want to make a house for them. On the outside, the house will be shaped like a convex polygon with N vertices. On the inside, it will be divided into several rooms by M interior walls connecting vertices along straight lines. No two walls will ever cross, but there might be multiple walls touching a single vertex.

So why is your house of kittens going to be so special? At every vertex, you are going to build a pillar entirely out of catnip! Kittens will be able to play with any pillar that touches the room they are in, giving them a true luxury home.

To make the house even more exciting, you want to use different flavors of catnip. A single pillar can only use one flavor, but different pillars can use different flavors. There is only one problem. If some room does not have access to *all* the catnip flavors in the house, then the kittens in that room will feel left out and be sad.

Your task is to choose what flavor of catnip to use for each vertex in such a way that (a) every flavor is accessible from every room, and (b) as many flavors as possible are used.

In the following example, three different flavors (represented by red, green, and blue dots) are distributed across an 8-sided house while keeping the kittens in every room happy:



In the image above, starting at the left corner of the top wall and going clockwise, the colors here are: green, blue, red, red, blue, green, blue, red.

Input

The first line of the input gives the number of test cases, T . T test cases follow.

Each test case consists of three lines. The first line gives N and M , the number of vertices and interior walls in your cat house. The second line gives space-separated integers U_1, U_2, \dots, U_M describing where each interior wall begins. The third line gives space-separated integers V_1, V_2, \dots, V_M describing where each interior wall ends.

More precisely, if the vertices of your cat house are labeled $1, 2, \dots, N$ in clockwise order, then the interior walls are between vertices U_1 and V_1 , U_2 and V_2 , etc.

Output

For each test case, output two lines. The first should be "Case #x: C", where x is the case number, and C is the maximum number of catnip flavors that can be used. The second line should contain **N** space-separated integers: " $y_1 y_2 \dots y_N$ ", where y_i is an integer between 1 and C indicating which catnip flavor you assigned to vertex i.

If there are multiple assignments with C flavors, you may output any of them.

Limits

$1 \leq T \leq 100$.

$1 \leq M \leq N - 3$.

$1 \leq U_i < V_i \leq N$ for all i.

Interior walls do not touch each other except at the **N** vertices.

Interior walls do not touch the outside of the house except at the **N** vertices.

Memory limit: 1GB.

Small dataset (Test set 1 - Visible)

$4 \leq N \leq 8$.

Time limit: 30 seconds.

Large dataset (Test set 2 - Hidden)

$4 \leq N \leq 2000$.

Time limit: 60 seconds.

Sample

Sample Input

```
2
4 1
2
4
8 3
1 1 4
3 7 7
```

Sample Output

```
Case #1: 3
1 2 1 3
Case #2: 3
1 2 3 1 1 3 2 3
```