UCF "Practice" Local Contest — Aug 27, 2016

Interstellar Love

filename: `stars`
(*Difficulty Level:* `Medium`)

After two years of sharing a bedroom with you in a college dorm, Jeff finally has his own room. Excited about inviting girls over to his room, he ponders over what decorations the fairer sex will enjoy.[1] He decides upon setting up a "fake" planetarium with a black ceiling and glow in the dark stars that form constellations. Unfortunately, in his haste, he has made several "errors" in setting up his constellations. See, everyone knows that constellations don't have cycles in them. Instead, whenever we visually connect the stars together with lines, a tree is always formed. (A cycle is formed when you can start at a star and, using connections, go to one or more other stars and then end up at the original star.)

Since you were Jeff's roommate for two years, you figure you'll help him fix his constellations. Your job will be twofold: to count the number of constellations Jeff has, and to report how many of them have cycles and need to be fixed. A constellation consists of multiple stars that are all connected to one another (directly or indirectly). A constellation that needs fixing is simply one that has a cycle.

The Problem:

Given several configurations of stars and connections between stars, determine how many constellations are defined in each configuration and how many need fixing.

The Input:

The first input line contains a positive integer, *n (n ≤ 100)*, indicating the number of night skies to consider. The first line of each night sky contains two positive integers, *s (s ≤ 1000)*, representing the number of stars for this night sky, and *c (c ≤ 10000)*, representing the total number of connections between pairs of stars for this night sky. Each of the following *c* input lines contains two distinct positive integers representing a single connection between two stars. The stars in each test case will be numbered 1 through *s*, inclusive. A connection is considered bidirectional, thus, if *a* is connected to *b*, *b* is connected to *a*. Assume that all connections in a data set are distinct, i.e., no duplicates. Also assume that there will never be a connection from a star to itself.

[1] This is based on a true story. The person who is the inspiration for this story did, in fact, major in Planetary Science and make his room's ceiling a relatively accurate depiction of the night sky, as seen in Boston circa 1995.

The Output:

For each test case, just output a line with the following format:

`Night sky #`*`k`*`: `*`X`*` constellations, of which `*`Y`*` need to be fixed.`

where *k* is the number of the night sky, starting at 1, *X* is the total number of constellations described in that night sky, and *Y* is how many of those constellations contain a cycle.

Leave a blank line after the output for each test case.

Sample Input:

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

Sample Output:

```
Night sky #1: 2 constellations, of which 1 need to be fixed.

Night sky #2: 3 constellations, of which 1 need to be fixed.

Night sky #3: 1 constellations, of which 0 need to be fixed.
```

Note: In the second example, star number 5 is not connected to any other stars. This star on its own is NOT counted as a constellation, leaving only {1,2}, {3,4} and {6,7,8} as constellations, of which only the last one needs to be fixed.