F2. Tree of Life (medium)

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Heidi got tired of deciphering the prophecy hidden in the Tree of Life and decided to go back to her headquarters, rest a little and try there. Of course, she cannot uproot the Tree and take it with her, so she made a drawing of the Tree on a piece of paper. On second thought, she made more identical drawings so as to have n in total (where n is the number of vertices of the Tree of Life) – who knows what might happen?

Indeed, on her way back Heidi was ambushed by a group of zombies. While she managed to fend them off, they have damaged her drawings in a peculiar way: from the i-th copy, the vertex numbered i was removed, along with all adjacent edges. In each picture, the zombies have also erased all the vertex numbers and relabeled the remaining $n - 1$ vertices arbitrarily using numbers 1 to n (fortunately, each vertex still has a distinct number). *What's more, the drawings have been arbitrarily shuffled/reordered.*

Now Heidi wants to recover the Tree of Life from her descriptions of all the drawings (as lists of edges).

Input

The first line of the input contains $Z \leq 20$ – the number of test cases. Z descriptions of single test cases follow.

In each test case, the first line of input contains numbers n $(2 \leq n \leq 100)$ and k (where k is the number of drawings; we have $k = n$). In the following lines, the descriptions of the k drawings are given. The description of the i-th drawing is a line containing m_i – the number of edges in this drawing, followed by m_i lines describing edges, each of which contains two space-separated integers –- the numbers of the two vertices connected by the edge.

Output

If Heidi's drawings cannot possibly come from a single tree, you should output the word `NO`. Otherwise, output one line containing the word `YES` and $n - 1$ lines describing any tree that Heidi's drawings could have come from. For every edge you should output the numbers of the vertices that it connects, separated with a single space. If there are many solutions, print any of them.

Example

input

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

output

```
YES
2 5
4 2
```

```
3 2
5 1
```