Problem K. Graph coloring

Source file name:	`K.c, K.cpp, K.java`
Input:	`Standard`
Output:	`Standard`
Author(s):	

As you learned in your Algorithms class(es), the 4 colors theorem only applies to planar graphs (that is, graphs that can be drawn without edges intersecting each other, except on their vertices).

However, you can always color a graph according to the rules of the theorem (no adjacent nodes can have the same color) with a number of colors $C \leq N$, the number of nodes (ultimately you can color each node with a different color).

On this problem, you will strive to find the minimum number of colors (min C) that you need to color all nodes from a directed graph following those same rules. Remember that in a directed graph an edge has direction, so if an edge connects A to B, that does not imply that B connects to A.

Input

Each test case input will be separated by a blank line and the end of the input is indicated by the EOF.

Each test case will be formatted as follows: On the first line a single integer $0 \leq N \leq 10$ indicates the number of nodes in the graph. The following N lines describe the information of the edges. The first integer $0 \leq E \leq N$ on line x indicate the number of edges connecting node x to another node. The rest E integers indicate the node number that each edge connects to. Nodes are enumerated from 1 to N.

Output

For each test case all you have to do is print a line with the minimum number of colors required to color the graph with the rules indicated above.

Example

Input	Output
4 2 2 3 2 1 3 3 1 2 4 1 3 5 4 2 3 4 5 4 1 3 4 5 4 1 2 4 5 4 1 2 3 5 4 1 2 3 4	3 5