CMSC 501

Programming assignment

Date due: 4/24/2014 5pm

A team of cavers are planning a training session in the Great Cave of Byte Mountains. The Great Cave is a system of *n* chambers linked by corridors. Each chamber is at a different level, starting with the Top Chamber at level 1, and reaching down to the Bottom Chamber at level *n*. The team of cavers decided that each of the cavers will explore a different route from Top Chamber to Bottom Chamber. That is, each caver will leave the Top Chamber by a different corridor, and each will enter the Bottom Chamber by a different corridor. The remaining corridors may be traversed by more than one caver. Also, the caver will always move down, i.e. the level of every consecutive chamber on a route should be lower than the previous one.

In order to finalize the plan, the team has one question left: Given the conditions above, how many cavers can participate in the training session?

Task

Write a program which:

- reads the cave system description from the standard input (System.in),
- computes the maximal number of cavers that may participate in training,
- writes the result to the standard output (System.out).

Input

In the first line there is one integer *n* (*2<=n<=200*), equal to the number of chambers in the cave system. The chambers are numbered with integers from *1* to *n* in descending level order - the chamber of greater number is at a lower level than the chamber of the lower number. (Top Chamber has number *1*, and Bottom Chamber has number *n*). In the following *n-1* lines (i.e. lines *2,3,...,n*) the descriptions of corridors leading from particular chambers are given. The (*i+1*)-th line contains numbers of chambers that are accessible by a corridor from the *i*-th chamber. (only lower chambers, i.e. chambers with numbers greater then *i* are mentioned). The first number in a line, *m*, *0<=m<=(n-i+1)*, is a number of corridors exiting the chamber being described. Then the following *m* integers are the numbers of the chambers the corridors are leading to.

Output

Your program should write one integer - the maximal number of cavers that can participate in the training session.

Example

Sample input:

```
12
4 3 4 2 5
1 8
2 9 7
2 6 11
1 8
2 9 10
2 10 11
1 12
2 10 12
1 12
1 12
```

Sample output:

```
3
```

Graphical view of the cave system described by the sample input
(Top Chamber: 1; Bottom Chamber: 12):