/ Skiers

Contest: 9th Polish Olympiad in Informatics
Task author: Piotr Chrząstowski
Memory: 32 MB
https://oi.edu.pl/en/archive/oi/9/nar

On the south slope of Bytemount there are several ski tracks and one ski lift. All the tracks run from the top station of the ski lift to the bottom station. Every morning a group of ski lift workers examines the condition of the tracks. Together, they take the lift up to the top station, and then each of them skis down along a chosen track to the bottom station. Each worker skis down only once. The tracks of the workers may be partially the same. Each track examined by any of the workers always leads downwards.

The map of ski tracks consists of a network of clearings connected by cuttings in the forest. Each clearing lies at a different height. Any two clearings may be connected directly by at most one cutting. Skiing down from the top to the bottom station, one can choose a track to visit any one clearing (although probably not all of them in a single run). Ski tracks may cross only at clearings, and do not run through tunnels or over bridges.

Task

Write a program that:

- reads from the input the map of ski tracks,
- computes the minimum number of workers needed to examine all the cuttings,
- writes the result to the output.

Input

In the first line of the input there is one integer n equal to the number of clearings ($2 \le n \le 5000$). The clearings are numbered from 1 to n.

Each of the successive $n - 1$ lines contains a sequence of integers separated by single spaces. The integers in the $(i + 1)$-th line specify which clearings the cuttings from clearing number i lead down to. The first integer k specifies the number of such clearings. The successive k integers are the numbers of the clearings, ordered from west to east according to the arrangement of the cuttings leading to them. The top station of the ski lift lies at clearing number 1 and the bottom station at clearing number *n*.

Output

In the first and only line of the output there should be exactly one integer: the minimum number of workers required to examine all the cuttings in the forest.

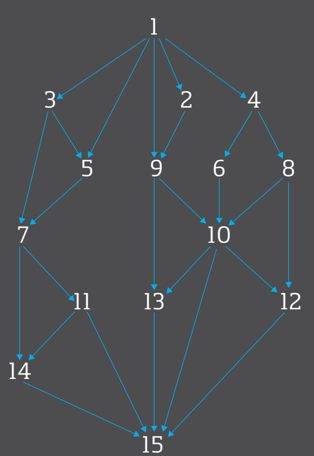
Example

For the input data:

```
15
5 3 5 9 2 4
1 9
2 7 5
2 6 8
1 7
1 10
2 14 11
2 10 12
2 13 10
3 13 15 12
2 14 15
1 15
1 15
1 15
```

the correct result is:

8



/ Solution

Note that we are dealing with a very special kind of graph. Nodes in the graph are the clearings, while edges are forest cuttings—each cutting connects a clearing with a different clearing. This is a planar graph, since it describes ski routes, which can intersect only at clearings: there are no tunnels or overpasses. Additionally, this graph has no cycles. Finally, the graph has two distinguished nodes: the upper and lower ski-lift stations. Each node can be reached from the top station, and from each node the bottom station can be reached. Let us call each such graph a *ski graph*.

For each ski graph we can define a dual graph as follows. The nodes of the dual graph are the edges of the original graph. There is an edge in the dual graph that connects e_1 with e_2 if the cutting e_1 exits at a clearing from which one can enter the cutting e_2. So the neighboring nodes in the dual graph are the edges of the original ski graph that lie one after another on the way from the top to the bottom of the hill. Let us denote the original ski graph by N and the dual graph by N_d.

Consider now the relation $\leq$ in the dual graph N_d, defined between its nodes (cuttings) as follows: $e_1 \leq e_2$ if and only if there is a path in N_d leading from e_1 to e_2. It is therefore a relation expressing the fact that one of the nodes is above the other one in a path from the top to the bottom. This relation has the following properties:

- it is reflexive, i.e. $\forall e \in E : e \leq e$;
- it is antisymmetric, i.e. $\forall e_1, e_2 \in E : e_1 \leq e_2 \wedge e_2 \leq e_1 \Rightarrow e_1 = e_2$;
- it is transitive, i.e. $\forall e_1, e_2, e_3 \in E : e_1 \leq e_2 \wedge e_2 \leq e_3 \Rightarrow e_1 \leq e_3$.

Each relation fulfilling these properties is called a *partial order* relation. Partial order relations play an important role in computing. They allow the classification of data and have been the subject of intensive research for many years.

We say that a set A is partially ordered if a relation of partial order is defined in it. Such an ordered set is denoted as a pair $\langle A, \leq \rangle$. It is always important to know what order we are talking about, because the sets can be ordered in a variety of ways. Our set of forest cuttings has just been ordered by the relation $\leq$, as defined above.