

# Fake tournament

We consider only special type of tournaments. Each tournament consists of a series of matches. We have  $n$  competitors at the beginning of a competition and after each match the loser is moved out of the competition and the winner stays in (there are no draws). The tournament ends when there is only one participant left - the winner. It is a task of National Sports Federation to schedule the matches. Members of this committee can pick the contestants for the first match. Then, after they know the result, they say which of the remaining contestants meet in the second match, and so on until there is only one participant left.

It is easy to see that not only skill and training decides about the win, but also "luck" - i.e. the schedule. The members of NSF know it as well.

The committee used the training time to look carefully on the performance of each probable contestant. It is clear now, at the start of the season, that some of the results between the competitors are 100% predictable. Having this information NSF considers if it is possible to schedule the matches in such a way that the given contestant  $x$  wins. That is to plan the matches for  $x$  only with those who will lose with him (then he wins the whole tournament of course). If it is possible then we say that **the tournament can be set for  $x$** .

## Task

Your task is to write a program which determines the number of contestants of a given tournament for which it is possible to set it.

## Input

$t$  [number of tests to solve].

In the first line of each test:  $n$  ( $1 \leq n \leq 1000$ ) - the number of participants of the tournament. We number the participants with numbers 1, 2 ...  $n$ . The following line contains a list of participants who will inevitably win with participant 1. This list begins with a number  $m$  (the number of contestants "better" than 1) and numbers  $n_1, n_2 \dots n_m$  delimited by single spaces.

Next  $n-1$  lines contain analogous lists for participants 2, 3 ...  $n$ .

Remark 1. The fact that participant **a** would lose with **b** and **b** would lose with **c** doesn't necessarily mean that **a** would lose with **c** in a direct match.

Remark 2. It is not possible that **a** is on the list of contestants better than **b** and **b** is on the list of **a** at the same time.

## Output

For each test your program should output a single integer - the number of participants, for which it is possible to set the tournament.

## Example

Input:

1

3  
2 3 2  
1 3  
0

Output:  
1