

Choose Your Own Adventure

You've been reading some Choose Your Own Adventure books and always seem to be getting to read the same few pages, no matter what choices you make. This makes you wonder - are all the pages in the book even reachable? Or, are some of the pages in the book, but if you follow the directions, there's no way to ever get to those pages?

You've decided to write a computer program to find out!

The Problem

Given the number of pages in a Choose Your Own Adventure book, as well as which pages you can go to from each page, determine the number of reachable pages in the book. Assume that you always start reading on page 1.

The Input (from file "adventure.in")

The first line of input will contain a single positive integer, n ($n \leq 100$), representing the number of input cases. The input cases follow. The first line of each input case contain a single positive integer p ($p \leq 1000$), representing the number of pages in the book for that input case. This is followed by p lines. The i^{th} of these lines will contain a space separate list, in increasing order, of the pages one could choose to go to from page i . If the line only contains the integer i , that means that page i is a possible ending to the story. Otherwise, the line will contain one or more integers, none of which are i .

The Output

For each input case, on a line by itself, output the number of reachable pages in the book.

Sample Input

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

Sample Output

```
3
1
```