

## Problem F. Forever and Always

Input file: *standard input*  
 Output file: *standard output*  
 Time limit: 2 seconds  
 Memory limit: 512 mebibytes

Consider an abstract voting procedure. For example, this might be voting for the best song of “Bullet for My Valentine” in 2019.

There are  $n$  persons taking part in voting, and there are  $m$  options to vote for. Every person has formed their own *preference list* which includes some of the options, in order from most preferred to least preferred. Note that some options might be missing from the preference list — such options are not just little preferred, but unacceptable.

Voting is conducted in *iterations*.

In the first iteration, every person votes for the first option on their preference list. The number of votes for every option is counted and announced to everyone.

In every subsequent iteration, every person intends to vote for the option on their preference list which has received the most number of votes in the previous iteration. If there are several such options, the one that comes earlier on the preference list is chosen.

Before every iteration, it’s asked whether anyone is going to vote differently from the previous iteration. If this is not the case, the iteration is not conducted, and the results of the last iteration are declared to be the final vote results. Otherwise, voting takes place, and similarly to the first iteration, the number of votes for every option is counted and announced to everyone again. Note that votes of the previous iterations become ignored.

This sort of voting procedure looks very cumbersome to you, and, most importantly, it looks like it may take forever to find out the results! To prove your point, propose values of  $n$ ,  $m$  and preference lists such that at least 100 iterations of voting will be conducted.

### Input

The only line of the input contains a single integer  $p$  — the required number of iterations.

There are two test cases. In test case 1,  $p = 2$ . In test case 2,  $p = 100$ .

### Output

Output two integers  $n$  and  $m$  ( $1 \leq n \leq 10^5$ ;  $1 \leq m \leq 2 \cdot 10^5$ ) — the number of persons and the number of options, respectively, followed by  $n$  preference lists.

Each preference list must be described by  $k_i$  ( $1 \leq k_i \leq m$ ) — the number of options on the list, followed by  $k_i$  distinct integers  $a_{i,j}$  ( $1 \leq a_{i,j} \leq m$ ) — option identifiers on the list, in order from most preferred to least preferred.

The sum of all values of  $k_i$  must not exceed  $2 \cdot 10^5$ .

### Example

standard input	standard output
2	4 5 2 1 2 1 2 3 5 1 3 2 2 3

### Note

Consider the example test case.

In the first iteration everyone votes for the first option on their list. Thus, the first person votes for option 1, the second and the fourth persons vote for option 2, and the third person votes for option 5.

In the second iteration, seeing that option 2 is now prevailing, the first person will change their vote from option 1 to option 2. Everyone else will keep their vote as is. In particular, the third person will keep his vote for option 5 since both option 5 and option 1 have had one vote in the first iteration, but option 5 is earlier on their list.

Finally, the third iteration is not conducted since nobody is willing to change their vote anymore. Two iterations have been conducted, which satisfies  $p = 2$ .