

E. Restoring Map

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Archaeologists found some information about an ancient land of Treeland. We know for sure that the Treeland consisted of n cities connected by the $n - 1$ road, such that you can get from each city to any other one along the roads. However, the information about the specific design of roads in Treeland has been lost. The only thing that the archaeologists can use is the preserved information about *near* cities.

Two cities of Treeland were called *near*, if it were possible to move from one city to the other one by moving through at most two roads. Also, a city is considered *near* to itself. During the recent excavations archaeologists found a set of n notes, each of them represents a list of cities, *near* to some of the n cities of the country. However, unfortunately, none of the found records lets you understand in what order the cities go in the list and for which city in the list the *near* to it cities were listed.

Help the archaeologists and restore any variant of the map of Treeland that meets the found information.

Input

The first line contains integer n ($2 \leq n \leq 1000$) — the number of cities in the country.

Next n lines describe the found lists of *near* cities. Each list starts from number k ($1 \leq k \leq n$), representing the number of cities in the list followed by k city numbers. All numbers in each list are distinct.

It is guaranteed that the given information determines at least one possible road map.

Output

Print $n - 1$ pairs of numbers representing the roads of the country. The i -th line must contain two integers a_i, b_i ($1 \leq a_i, b_i \leq n, a_i \neq b_i$), showing that there is a road between cities a_i and b_i .

The answer you print must satisfy the description of close cities from the input. You may print the roads of the countries in any order. The cities that are connected by a road may also be printed in any order.

If there are multiple good answers, you may print any of them.

Examples

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

input
6 5 6 1 3 4 2 5 2 1 3 4 6 6 3 6 2 5 4 1 6 6 1 2 5 3 4

3	5	2	4		
5	3	1	2	4	6

output

2	4
1	2
2	3
2	6
4	5