Problem A: The mysterious X network

One of the reasons for which École polytechnique (nicknamed "X" for reasons to be explained during the debriefing talk) is so deeply rooted in French society is its famous network of *camarades* – former students of the same school. When one camarade wants something (money, job, etc.), he can ask this network for help and support. In practice, this means that when he/she wants to reach some other camarade, not always of the same year, then surely he can find intermediate camarades to get to her/him. Note that the "camarade" relationship is symmetric. Due to the magic of the X network, there is always a means to reach anybody.

The program you have to write is supposed to help to minimize the number of these intermediate camarades.

Details

The huge file of all living camarades is simplified so as to obey the following format. The first line in the file is the number of camarades, say N, an integer $1 \leq N \leq 10^5$. Camarades are labeled from 0 to $N-1$. Follow N lines. Each line starts with the camarade label c, followed by the number of other camarades he/she knows, say n_c, followed by the labels of those n_c camarades. All these integers are separated by a single blank. It is assumed that n_c is always less than 100. The last line in the file is the label of the camarade seeking help (say c_1) followed by the label of the camarade he wants help from, say c_2 ($c_2 \neq c_1$).

Your program should output three integers separated by a blank: c_1, c_2 and the minimal number of *intermediate* camarades to reach c_2.

Examples

For example the input file is:

```
4
0 3 1 2 3
1 1 0
2 2 0 3
3 2 0 2
1 2
```

and the answer is:

```
1 2 1
```