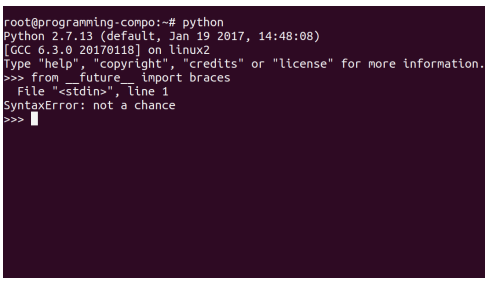
Problem I
Import Spaghetti
Problem ID: importspaghetti
Time limit: 4 seconds



cc-by NCPC 2017

You just graduated from programming school and nailed a Python programming job. The first day at work you realize that you have inherited a mess. The *spaghetti* design pattern was chosen by the previous maintainer, who recently fled the country. You try to make sense of the code, but immediately discover that different files depend cyclically on each other. Testing the code, in fact running the code, has not yet been attempted.

As you sit down and think, you decide that the first thing to do is to eliminate the cycles in the dependency graph. So you start by finding a shortest dependency cycle.

Input

The first line of input contains a number n, $1 \leq n \leq 500$, the number of files. Then follows one line with n names of files. Each name is a string with at least 1 and at most 8 lower case letters ‘a’ to ‘z’. Then follow n sections, one section per file name, in the order they were given on the second line. Each section starts with one line containing the name of the file and an integer k, followed by k lines, each starting with “`import`”.

Each “`import`” line is a comma-space separated line of dependencies. No file imports the same file more than once, and every file imported is listed in the second line of the input. Comma-space separated means that every line will start with “`import`”, then have a list of class names separated by `", "` (see sample inputs for examples).

Output

If the code base has no cyclic dependencies, output “`SHIP IT`”. Otherwise, output a line containing the names of files in a shortest cycle, in the order of the cycle. If there are many shortest cycles, any one will be accepted.

Sample Input 1	Sample Output 1
4 a b c d a 1 import d, b, c b 2 import d import c c 1 import c d 0	c

Sample Input 2

```
5
classa classb myfilec execd libe
classa 2
import classb
import myfilec, libe
classb 1
import execd
myfilec 1
import libe
execd 1
import libe
libe 0
```

Sample Output 2

```
SHIP IT
```

Sample Input 3

```
5
classa classb myfilec execd libe
classa 2
import classb
import myfilec, libe
classb 1
import execd
myfilec 1
import libe
execd 1
import libe, classa
libe 0
```

Sample Output 3

```
classa classb execd
```