

Coding Question

8. Implement depth first search in either C, C++, C#, Java, or Python. Be efficient and implement it in $O(n + m)$ time, where n is the number of graph nodes, and m is the number of graph edges. Remember to submit a makefile along with your code, just as with week 1's coding question.

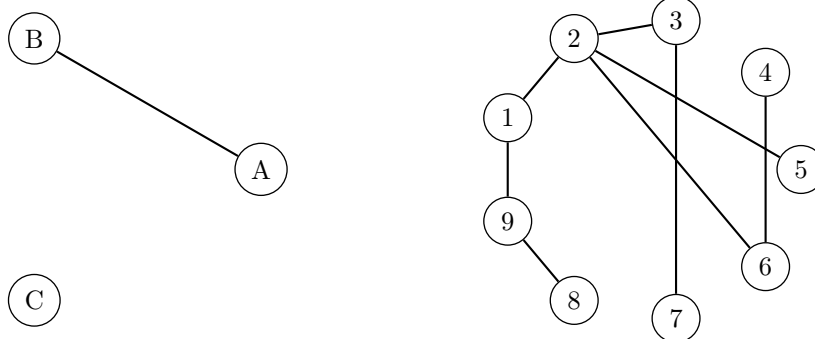
Input format: the input will start with a positive integer, giving the number of instances that follow. For each instance, there will be a positive integer, giving the number of graph nodes. For each node there will be a line of space-delimited input. The first string in a line will be the node name. All following strings in a line will be the names of the adjacent nodes to the first node in the line.

You can assume the order the nodes will be listed is in increasing lexicographic order (0-9, then A-Z, then a-z), both in each list of adjacent nodes, as well as the order nodes' lines are listed.

A sample input is the following:

```
2
3
A B
B A
C
9
1 2 9
2 3 5 6
3 2 7
4 6
5 2
6 2 4
7 3
8 9
9 1 8
```

The sample input has two instances. The first instance corresponds to the graph below on the left. The second instance corresponds to the graph below on the right.



Target output: for each instance, your program should output the names of nodes visited in depth first traversal of the graph, *with ties between nodes visiting the first node in lexicographic ordering*. Start your traversal with the first node in lexicographic order. Each instance's traversal should be on a separate line. Each output line should be terminated by a newline. The correct output to the sample input would be:

```
A B C
1 2 3 7 5 6 4 9 8
```