```cpp
#include <iostream>
#include <vector>
#include <queue>

using namespace std;

#define MAX 10000

bool visited[MAX];

void dfs(vector<vector<int>> n, int node)
{

    for (int neighbuor : n[node])
    {
        if (!visited[neighbuor])
        {
            dfs(n, neighbuor);
        }
    }
    visited[node] = true;
    cout << node << ' ';
}

void bfs(vector<vector<int>> v)
{

    queue<int> q;
    int start = 0;
    q.push(start);
    visited[start] = true;

    while (!q.empty())
    {
        int first = q.front();
        cout << first << ' ';
        for (int n : v[first])
        {
            if (!visited[n])
            {
                q.push(n);
                visited[n] = 1;
            }
        }
        q.pop();
    }
}

int main()
{
    int n;
    cout << "Enter how many node you have : ";

    cin >> n;
    vector<vector<int>> v;

    // vector input
    for (int i = 0; i < n; i++)
    {
        vector<int> n;
        cout << "Enter how many neighbour for node " << i << " : ";

        int neighbour;
        cin >> neighbour;

        for (int j = 0; j < neighbour; j++)
        {
            int val;
            cin >> val;
            n.push_back(val);
        }
        v.push_back(n);
    }

    dfs(v, 0);

    bfs(v);
    return 0;
}
```