

National University of Computer and Emerging Sciences



Laboratory Manual *for* **Data Structures Lab 13**

Course Instructor	Dr. Saira Karim
Lab Instructor(s)	Ms. Fariha Maqbool Ms. Muntaha
Section	BCS-3E
Semester	Fall 2022

Department of Computer Science

FAST-NU, Lahore, Pakistan

Objectives:

In this lab, students will practice:

1. Graphs
2. DFS
3. BFS

Q1: You are required to implement the adjacency list class. Assume the graph is directed.

```
#include<iostream>

#include<List> using
namespace std; class
Graph
{
    int v; // No. of vertices list<int>
    *adj; // adjacency lists public:
        Graph(int n); // Constructor
        void TakeInput(int n, int w); // to take input from the user in this sequence: number of
nodes, what are the neighbors of vertex 0, what are the neighbors of vertex 1, ... so on.
        void ExploreFunction(int start); // print the paths from start to every other vertex as
generated by the dfs method. One path per line.
};

int main()
{
    Graph g(4); // Total 5 vertices in graph
    g.TakeInput(0, 1);
    g.TakeInput(0, 2);
    g.TakeInput(1, 2);
    g.TakeInput(2, 0);
```

```

g.TakeInput(2, 3);
g.TakeInput(3, 3); cout << "Following is Depth
    First Traversal\n";
g.ExploreFunction(2);
//Sample output of explore function for starting from vertex 2.
//2 0
//2 0 1
//2 0 1 3
//*****
//Sample output of explore function for starting from vertex 1.
//1 2
// 1 2 0 //1
2 0 3
return 0;
}

```

Q2:

- Implement DFS using a vertex class where a vertex can have name and some attributes
- print DFS spanning tree
- detect cycle using DFS
- Find how many connected component graph