

# National University of Computer and Emerging Sciences



## **Lab Manual** *for* **Data Structure**

Course Instructor	Ms. <a href="#">arooj Khalil</a>
Lab Instructor(s)	Mr. Sohaib Ahmad Mr. <a href="#">Dilawar Shabbir</a>
Section	BSE 3A
Semester	FALL 2022

Department of Computer Science  
FAST-NU, Lahore, Pakistan

## Lab Manual 11

### Objectives:

After performing this lab, students shall be able to revise:

- ✓ Graphs

### **Problem 1**

Implement a class Graphs using adjacency matrix for **directed** and **undirected** graphs (two different implementations) with following functions structure.

```
class Graphs
{
private:
    int** Matrix;
    int vertices;// total number of vertices
    bool isDirected; // 0 for undirected, 1 for directed

public:
    Graphs(int Tvertices, bool dir);
    Graphs(const Graphs& obj);
    Graphs(string fName);
    bool addEdge(int x,int y);
    bool removeEdge(int x,int y);
    bool isConnected(int x, int y);
    void printPathBFS(int src, int dest);
    void printPathDFS(int src, int dest);
    bool areConnected(int src, int dest);
    int shortestPath(int src, int dest);// assume one unit cost for one traversal
    int getIndegree();
    int getOutdegree();
    void printAllAdjc();
    bool isComplete();// is every node connected to any other node
    void printGraph();// print the whole Matrix
    ~Graphs();
};
```

### **Problem 2**

Now use adjacency list to implement the above class.

directedGraph.txt

---

6 0  
0 1 3  
1 2 3 5  
2 1  
3 4 5  
4 1 2 3 5  
5 0 2

undirectedGraph.txt

---

6 1  
0 1 3  
1 0 2 3 4 5  
2 1 5  
3 0 1 5  
4 1  
5 1 2 3