East West University

Department of Computer Science & Engineering
A/2, Jahurul Islam Avenue, Jahurul Islam City, Aftabnagar, Dhaka-1212

_____

Lab Manual   : 08

Course Code   : CSE207

Course Title   : Data Structure

Instructor   : Tanni Mittra, Lecturer, CSE

Objective:

The objective of this lab is to provide basic concept of Graph. At the end of the lab, students are able:

  ➢ To learn how to create Graph
  ➢ To learn how to perform Insertion, Deletion of nodes in Graph
  ➢ To learn how to traverse nodes of a graph

Graph:

Graph is a data structure that consists of following two components:

1. a finite set of vertices also called as nodes.

2. a finite set of ordered pair of the form (u, v) called as edge. The pair is ordered because (u, v) is not same as (v, u) in case of directed graph (di-graph). The pair of form (u, v) indicates that there is an edge from vertex u to vertex v. The edges may contain weight.

Graphs are used to represent many real life applications: Graphs are used to represent networks. The networks may include paths in a city or telephone network or circuit network. Graphs are also used in social networks like linkedIn, facebook. For example, in facebook, each person is represented with a vertex (or node). Each node is a structure and contains information like person id, name, gender and locale. Following is an example undirected graph with 5 vertices.
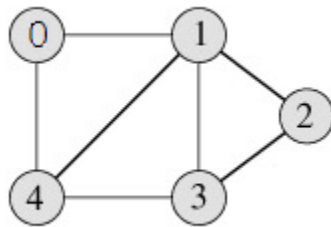
Exercise 1:
Create and Display Graph
Here you have to take number of nodes as input and the source and destination of connecting edge. After taking input display the full graph with number of edges and nodes

Exercise 2:
Add new vertex and edge
Add a new vertex to the graph and add an edge between the two vertices of the graph.

Exercise 3:
Delete vertex
Delete a vertex and its edges from the graph.

Exercise 4:
Traversal of graph (BFT and DFT)

Breadth First Traversal (or Search) for a graph is similar to Breadth First Traversal of a tree (See method 2 of this post). The only catch here is, unlike trees, graphs may contain cycles, so we may come to the same node again. To avoid processing a node more than once, we use a boolean visited array. For simplicity, it is assumed that all vertices are reachable from the starting vertex.

Depth First Traversal (or Search) for a graph is similar to Depth First Traversal of a tree. The only catch here is, unlike trees, graphs may contain cycles, so we may come to the same node again. To avoid processing a node more than once, we use a boolean visited array.