



**THE UNIVERSITY
OF LAHORE
ISLAMABAD
CAMPUS**

Data Structures & Algorithms (CS09203)

Lab Report

Name: Muhammad Umer
Registration #: CSU-F16-104
Lab Report #: 07
Dated: 21-05-2018
Submitted To: Mr. Usman Ahmed

The University of Lahore, Islamabad Campus
Department of Computer Science & Information Technology

Experiment # 1

Create a C++ program to implement the graph and add edges in the adjacency list and display added edges.

Objective

To understand and implement the graph with basic edges insertion in the adjacency list.

Software Tool

1. Sublime Text Editor
2. Dev C++
3. Window 7 (32 Bit)

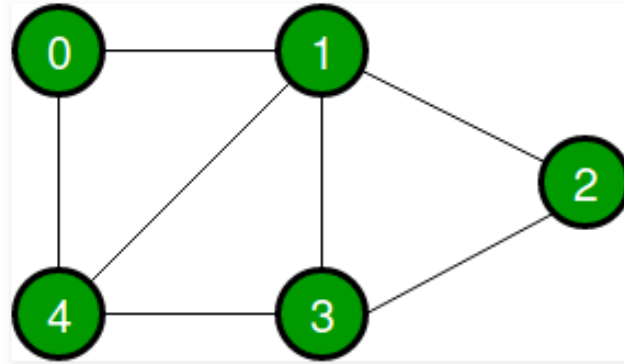
1 Theory

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 (digraph). The pair of form (u, v) indicates that there is an edge from vertex u to vertex v . The edges may contain weight/value/cost.

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.



2 Task

2.1 Procedure: Task 1 Inserting Edges

The following code is the representation of the above graph using STL

```

void addEdge(vector<int> adj[] , int u, int v)
{
    adj[u].push_back(v);
    adj[v].push_back(u);
}

```

2.2 Procedure: Task 2 Displaying the Edges

```

void printGraph(vector<int> adj[] , int V)
{
    for (int v = 0; v < V; ++v)
    {
        cout << "\nAdjacency list of vertex_"
              << v << "\nhead_";
        for (auto x : adj[v])
            cout << " _->_" << x;
        printf("\n");
    }
}

```

2.3 Procedure: Task 3 Main Function

```
#include<bits/stdc++.h>
using namespace std;

int main()
{
    int V = 5;
    vector<int> adj[V];
    addEdge(adj, 0, 1);
    addEdge(adj, 0, 4);
    addEdge(adj, 1, 2);
    addEdge(adj, 1, 3);
    addEdge(adj, 1, 4);
    addEdge(adj, 2, 3);
    addEdge(adj, 3, 4);
    printGraph(adj, V);
    return 0;
}
```

Output :

Consider the Figure 1 (in the end of this document) for the output of the above graph.

Note: My IDE (Dev C++) doesn't support C++11, as the above code is written in C++11, I run this code in the online IDE "<https://ide.geeksforgeeks.org/index.php>", which give me the output showed in Figure 1

Source Code

<https://goo.gl/ccBvqK>

3 Conclusion

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.

(Concerned Teacher/Lab Engineer)

Adjacency list of vertex 0
head -> 1-> 4

Adjacency list of vertex 1
head -> 0-> 2-> 3-> 4

Adjacency list of vertex 2
head -> 1-> 3

Adjacency list of vertex 3
head -> 1-> 2-> 4

Adjacency list of vertex 4
head -> 0-> 1-> 3

Figure 1: Adjacency list representation of the graph