Bahria University,

Karachi Campus

LAB EXPERIMENT NO. _13_

LIST OF TASKS

TASK NO	OBJECTIVE
Task 1	Create a program to implement Graphs with Adjacency matrix.
Task 2	Create a program to implement Graphs with Adjacency list.

Submitted On:

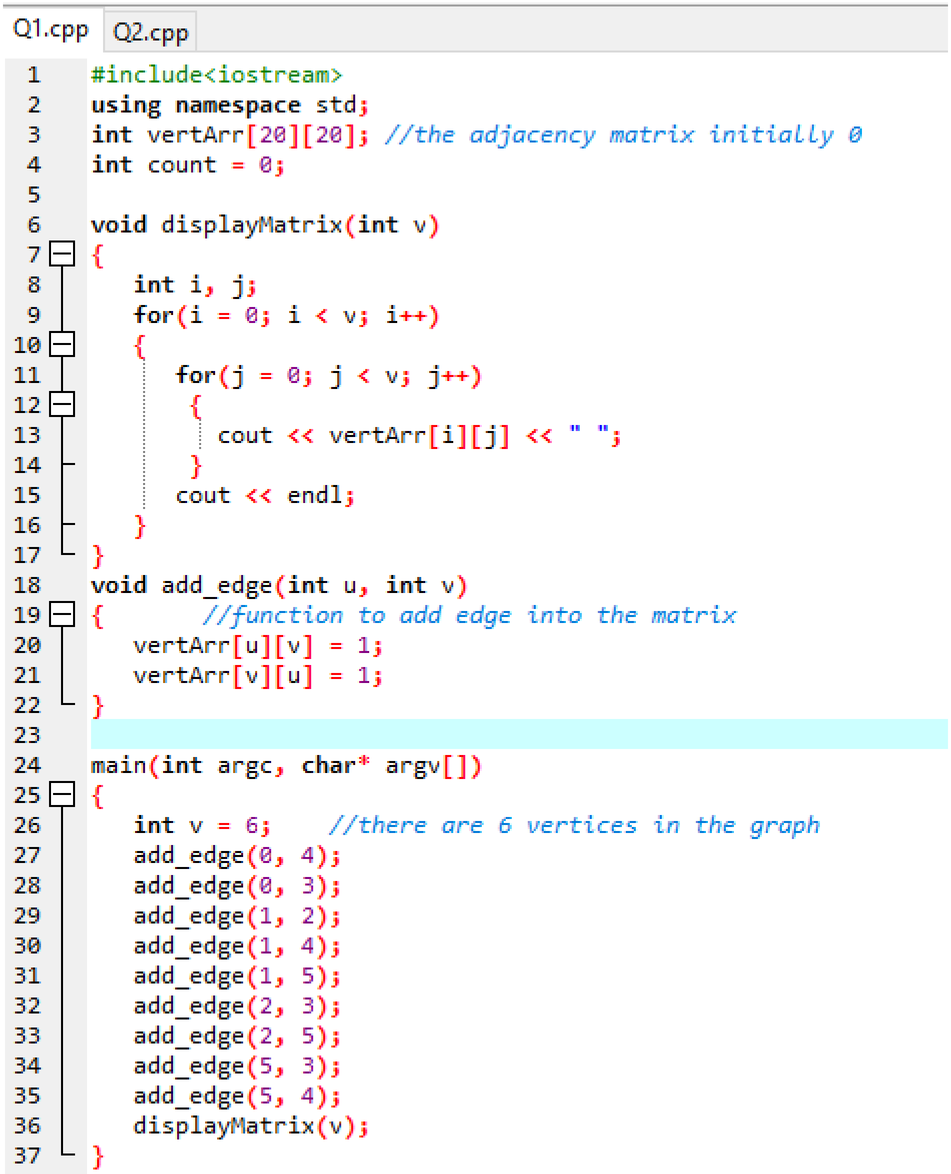
__25/06/2020__

(Date: DD/MM/YYY)

Task No. 1:

Create a program to implement Graphs with Adjacency matrix.

Coding:

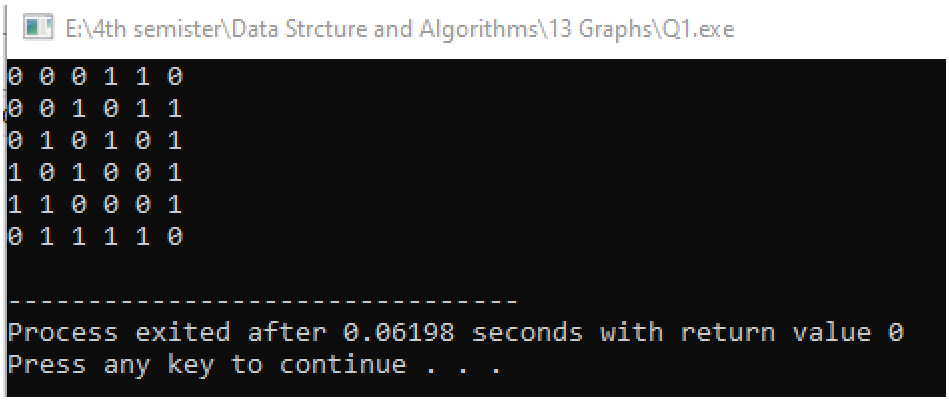


```cpp
#include<iostream>
using namespace std;
int vertArr[20][20]; //the adjacency matrix initially 0
int count = 0;

void displayMatrix(int v)
{
   int i, j;
   for(i = 0; i < v; i++)
   {
      for(j = 0; j < v; j++)
      {
         cout << vertArr[i][j] << " ";
      }
      cout << endl;
   }
}
void add_edge(int u, int v)
{       //function to add edge into the matrix
   vertArr[u][v] = 1;
   vertArr[v][u] = 1;
}

main(int argc, char* argv[])
{
   int v = 6;    //there are 6 vertices in the graph
   add_edge(0, 4);
   add_edge(0, 3);
   add_edge(1, 2);
   add_edge(1, 4);
   add_edge(1, 5);
   add_edge(2, 3);
   add_edge(2, 5);
   add_edge(5, 3);
   add_edge(5, 4);
   displayMatrix(v);
}
```

Output:



E:\4th semister\Data Strcture and Algorithms\13 Graphs\Q1.exe

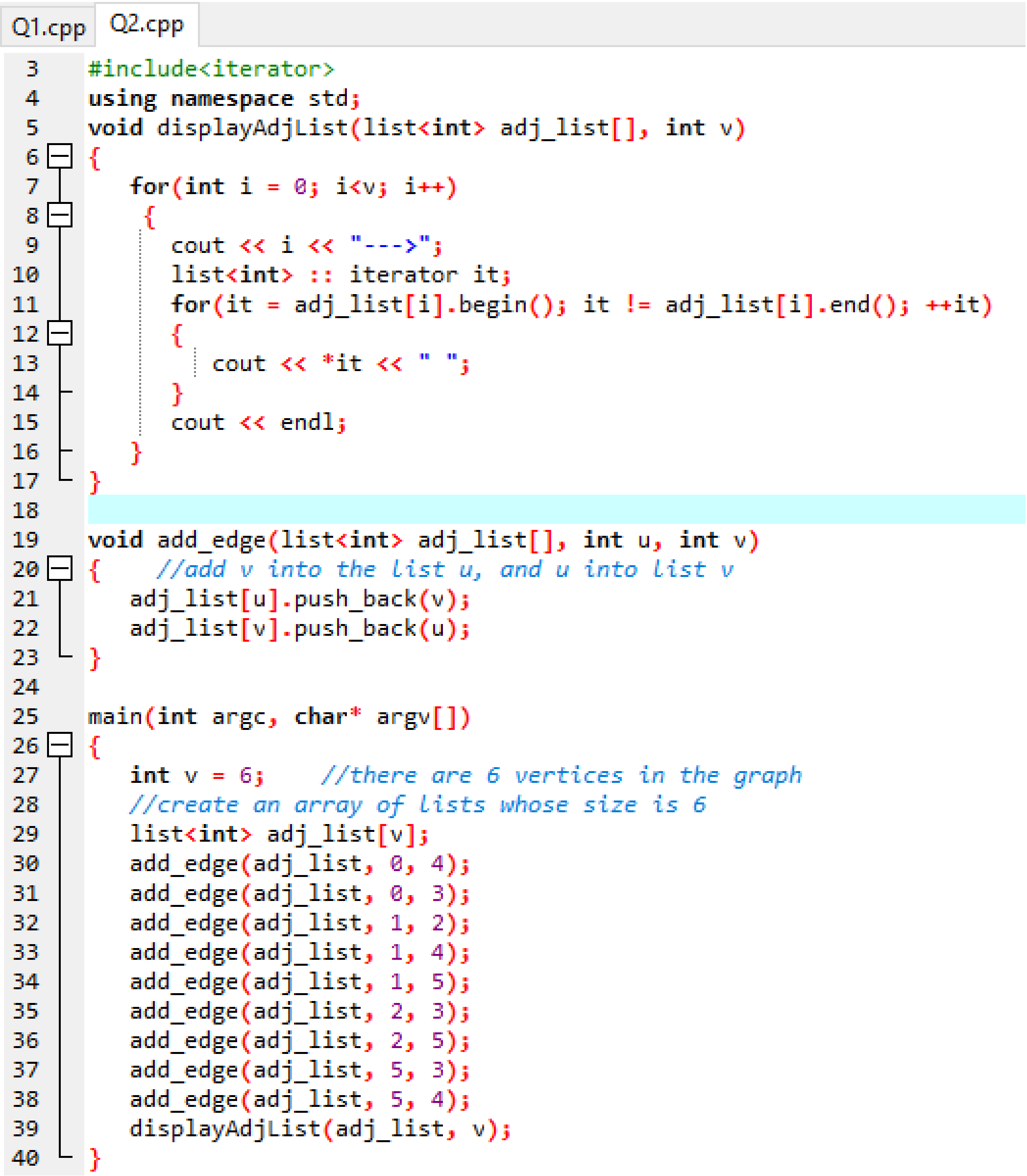
```
0 0 0 1 1 0
0 0 1 0 1 1
0 1 0 1 0 1
1 0 1 0 0 1
1 1 0 0 0 1
0 1 1 1 1 0

--------------------------------
Process exited after 0.06198 seconds with return value 0
Press any key to continue . . .
```

Task No. 2:

Create a program to implement Graphs with Adjacency list.

Coding:

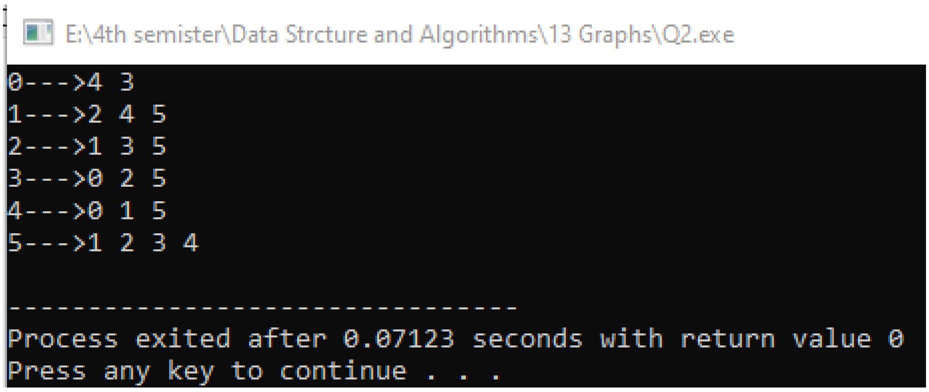


```cpp
#include<iterator>
using namespace std;
void displayAdjList(list<int> adj_list[], int v)
{
    for(int i = 0; i<v; i++)
     {
      cout << i << "--->";
      list<int> :: iterator it;
      for(it = adj_list[i].begin(); it != adj_list[i].end(); ++it)
      {
         cout << *it << " ";
      }
      cout << endl;
    }
}

void add_edge(list<int> adj_list[], int u, int v)
{    //add v into the list u, and u into list v
    adj_list[u].push_back(v);
    adj_list[v].push_back(u);
}

main(int argc, char* argv[])
{
    int v = 6;    //there are 6 vertices in the graph
    //create an array of lists whose size is 6
    list<int> adj_list[v];
    add_edge(adj_list, 0, 4);
    add_edge(adj_list, 0, 3);
    add_edge(adj_list, 1, 2);
    add_edge(adj_list, 1, 4);
    add_edge(adj_list, 1, 5);
    add_edge(adj_list, 2, 3);
    add_edge(adj_list, 2, 5);
    add_edge(adj_list, 5, 3);
    add_edge(adj_list, 5, 4);
    displayAdjList(adj_list, v);
}
```

Output:


E:\4th semister\Data Strcture and Algorithms\13 Graphs\Q2.exe

```
0--->4 3
1--->2 4 5
2--->1 3 5
3--->0 2 5
4--->0 1 5
5--->1 2 3 4

--------------------------------
Process exited after 0.07123 seconds with return value 0
Press any key to continue . . .
```