Group Project (Task 1)

Due Date: Saturday September 21, 2019 till 11:59 PM

Write a program which should read any graph from the user at run time by taking the vertices and edges and then display its adjacency matrix on screen as an output.

Program 1: Representing Graph as adjacency matrix

1. Create a package inside the project with the name "GraphPackage"
2. Create a main class inside this package with the name "AddGraph" and add the following code in it.

```
package GraphPackage;

public class AddGraph   //Main Class
{
    public static void main(String[] args)
    {
        //Create an object 'theGraph' of  Graph class
      //Write commands for getting the vertices and edges to be added in the graph ,
      //from the user here. And call the appropriate methods to add the vertices and
      //edges in the graph here
            System.out.println("The graph entered is: ");

            //Call the method displayGraph to display the graph as adjacency matric
            System.out.println();
    } // end main()


} // end class AddGraph

////////////////////////////////////////////////////////////////
```

3. Create another class inside the package "GraphPackage" with the name "Graph" and add the following code in it.

```
package GraphPackage;

public class Graph //Class to implement graph
{
    private final int MAX_VERTS = 20;
    private Vertex vertexList[]; // Array of vertices as objects of class Vertex
    private int adjMat[][]; // adjacency matrix
    private int nVerts; // current number of vertices

// ------------------------------------------------------------
    public Graph() // constructor
    {
         //Write statements here to initialize the above data members
    } // end constructor
// ------------------------------------------------------------



    public void addVertex(char lab)
    {
            //Write statement(s) here to add new vertex


    }
// ------------------------------------------------------------
    public void addEdge(int start, int end)
    {
            //Write statement(s) here to add new edge for undirected graph
```

```
    }
// ------------------------------------------------------------
    public void displayVertex(int v)
    {
            //Write statement(s) here to display the vertex label


    }


// ------------------------------------------------------------
    public void displayGraph(){
        displayGraph(adjMat);
    }

    private void displayGraph(int[][] adjMat)
    {
        //Write statements here to display the adjacency matrix of the given graph like
        //the output given below

     }

    }// End of Graph class



// ------------------------------------------------------------

    class Vertex //Class to implement vertices of the graph
    {
        public char label; // label (e.g. 'A') or "Jeddah" if defined as String
        public boolean wasVisited;
        // ------------------------------------------------------------
        public Vertex(char lab) // constructor
        {
            label = lab;
            wasVisited = false;
        }
        // ------------------------------------------------------------
    } // end of class Vertex

} //End of class Graph
////////////////////////////////////////////////////////////////
```

Output:

```
The graph entered is:

          A     B     C     D     E
    A     0     1     0     1     0
    B     1     0     1     0     0
    C     0     1     0     0     0
    D     1     0     0     0     1
    E     0     0     0     1     0
```

Note:

- Write **Member 1: ID: Name:** for all group members as comments on top of every class.
- Only Member 1 will upload the solution on Blackboard.