**REPORT LAB 8**

Task 1: read

Task 2: Implement Graph data structure in Java

**Graph class:**

Manages the graph structure with methods to add vertices, edges, and perform operations like Depth-First Search (DFS), path finding, and Dijkstra's algorithm.(in task 5)

DFS:

A computer screen shot of code

Description automatically generated

Path finding:

A screen shot of a computer code

Description automatically generated

**Vertex class (Vertex.java):**

Represents each node in the graph and maintains its neighbors and edge weights.

A computer screen shot of a program code

Description automatically generated

**Edge presentation (Edge.java):**

While edges are implicit in the adjacency list maintained in the Graph class, their weights are stored in the neighbors map of each vertex.

A screen shot of a computer program

Description automatically generated

Task 3: Form the Graph

The graph from Figure 1 was implemented with vertices A to L and edges with corresponding weights. Data was added in the MapApp.java

A screen shot of a computer

Description automatically generated

The graph's adjacency list representation ensures efficient management and traversal.

Task 4: Find the path

Print out the number of paths to go from A to K



Print out all the paths with the smallest and largest number of nodes from A to K and the corresponding cost (total weight)



Task 5: Dijkstra's algorithm

A screen shot of a computer code

Description automatically generated

It is in the Graph.java, which

 Begin with the starting vertex and explore all neighbors.

 Update paths dynamically, ensuring that the shortest path is always prioritized.

* The shortest path from the start to the end is printed, along with its cost.

Output:

