REPORT

IT-427

PROGRAMMING ASSIGNMENT 2

Implementing Dijkstra Algorithm

Assignment- A java code to find the shortest path of a given graph(file) using Dijkstra’s algorithm.

The commands to run the code for the given file

wdGraphs.txt

Compile – javac dijkstra.java

Run – java dijkstra wdGraphs.txt

**Code Explanation**

The code consists of multiple methods and classes

**Class vertex**

A class named vertex has been created, which has destination and weight fields.

I am using this to store the graph Data.

**Class Dijkstra**

This class has the main method which takes graph file path via command line arguments and I am using Scanner to read the data and storing it in List. And after

Parsing is done I am calculating the shortest path

Methods-

**findShortestPath() –** return type **double**

I am using a priority queue and the order of storage is the nodes are compared based on their weights and are stored in ascending order. I am returning the shortest path to end vertex, If the end vertex distance is still set to INFINITY

It means there is no shortest path

**printPath()** void -return type

I am using this method to print the output format based on the given requirement.

**Data Structures Used-**

* List
* Queue

**Algorithm**- Dijkstra’s algorithm was used as it uses greedy approach.

I am initializing the value as infinity to all the nodes as their distance, and starting my calculation form source vertex that is 0, and the weight is 0. Adding that value in priority queue , after calculating I am adding the neighboring nodes into the queue and so on, until the queue is empty.

Time complexity of my code-

Using priority queue the complexity of the algorithm will be **O(|E|+|V|log|V|)** where ,

**E** is the number of edges

**V**  is the number of nodes.

Space complexity of my code-

O(V^2) -It’s the worst case , as I am checking each node and its neighboring vertices.

Where V is the number of vertices of the graph