IT251 Assignment-5

NAME: SUYASH CHINTAWAR

ROLL NO.: 191IT109

TOPIC: DIJKSTRA’S ALGORITHM

VARIABLES:

- **n** – number of vertices in the graph
- **edges** – number of edges in the graph
- **edge[edges][3]** – array to store each pair of edges with their weights
- **adj_list[n]** – vector of vector of pairs consisting of adjacency list of each vertex with the edge weights
- **type** – type of graph (directed/undirected)
- **src** – source vertex, assumed as vertex 1
- **d** – vector of distances of all vertices, initialized to infinite
- **parent** – vector containing parent vertices of all vertices
- **found** – vector to store shortest distances of vertices which are finalized
- **q** – priority queue to implement min-heap
- **path** – vector to store final path from source to a vertex in reverse order

README and assumptions:

This program implements the Dijkstra's Algorithm on both, directed as well as undirected graph, considering edges as 1-based

- Adjacency list is generated using the edge set given by the user by calling **adjacency_list().**

- **dijkstra()** is then called with source vertex as 1, and is implemented as given below
- Initially, all the vertices will have distances as infinite except for the source vertex which will have distance as zero.
- The source vertex is then pushed in a min-heap which is implemented using a priority queue by using **greater<pair<int,int>>** as its default is max-heap
- We run a while loop until the queue is empty by popping off the top most element in queue which has the least distance from source vertex and then its adjacent vertices are visited and their distances are updated if there is a shorter path via this vertex.
- Parents of the vertices are stored so as to retrieve the shortest path from the source vertex.
- Finally, the shortest distances are stored in **found** and it is given as output along with the path reachable for every vertex from the source.

Time complexity of this algorithm is **O(E*logV)** where E is number of edges and V, number of vertices as adjacency list and priority queue is used.

THANK YOU