

COP 5537 - Assignment 1 Pseudocode

File is read in (Data.txt), declare our adjacency matrix as 'matrix,' a 2d int array, initialize all helper variables.

For each row in our matrix, fill with the corresponding row from the Data.txt file.

Run Dijkstra's algorithm with method "runDijkstra(matrix, start, end)" which requires the previously built matrix, the starting vertex (source) and the desired final location that the shortest path will be found for.

runDijkstra():

Determine the total number of vertices in the graph.

Initialize helper variables that will keep vertices creating the shortest path (int[] shortestPath), keeping track of state (boolean[] visitedVertex) and if vertex has an edges before itself (boolean noPrevVertex).

Mark all vertices with infinity and set their visited state to FALSE.

For each vertex:

1. If we haven't marked this vertex as VISITED and there is a newest shortest path: update shortest path distance. closestVertex gets updated and shortestDistance is updated.
2. Mark vertex as VISITED and move on.
3. Check distance from current vertex to closest vertex (denoted with distance variable).
4. If distance > 0 and shortest distance thus far plus possible path is less than the previously thought to be shortest path, then update shortestPath with new vertex and update both the distance edge and the overall shortestDistance variable.

Display results with helper method 'displayData()' and 'displayShortestPath'