CO322 - Lab2

Q2. What is the runtime complexity of your implementation.

This implementation goes through every path from 0 to N-1 while going through recursive trees (N being the number of stations) .Thus, the time complexity of the above implementation is exponential.

Q3. Argue that dynamic programming can be used to improve the runtime.

In this implementation it calculate solutions for the same sub problems multiple times. This could increase the runtime. To avoid this situation we can use dynamic programming so that we won't have to find solutions for the same sub problem more than once. A method to store the solutions of sub problems can be used so that we can use them again and again without calculating always. If we use a 2D array to store the solutions of sub problems in the current implementation we can increase the run time of our implementation easily.

Q5.Calculate the runtime of your implementation in part 4 above. Assume, hashin g is O(1).

Runtime complexity of this implementation is $O(N^3)$. Here, in this implementation, the minimum cost for the same path is not calculated more than once as the minimum cost for each path is stored in an array after calculating once. Hence, although the recursive function is implementing the runtime complexity would be $O(N^3)$

Example recursive tree -:

minCost(0,4)

↓

minCost(0,1) + minCost(1,4)

↓

minCost(0,2) + minCost(2,4);

↓ ↓

minCost(0,1) + minCost(1,2) minCost(2,3) +
minCost(3,4)