

CSCI 2300 Homework # 6

Q1 [25pnts]. Given an undirected graph G , describe a linear time algorithm to find the number of distinct shortest paths between two given vertices u and v . Note that two shortest paths are distinct if they have at least one edge that is different.

Q2 [25pnts]. Given a weighted directed graph with positive weights, given a $O(|V|^3)$ algorithm to find the length of the shortest cycle or report that the graph is acyclic.

Q3 [25pnts]. Given a directed weighted graph G , with positive weights on the edges, let us also add positive weights on the nodes. Let $l(x,y)$ denote the weight of an edge (x,y) , and let $w(x)$ denote the weight of a vertex x . Define the cost of a path as the sum of the weights of all the edges and vertices on that path. Give an efficient algorithm to find all the smallest cost path (as defined above) from a source vertex to all other vertices. Analyze and report the running time of your algorithm.

Q4 [25pnts]. Given a directed graph G with possibly negative edge weights. Consider the following algorithm to find the shortest paths from a source vertex to all other vertices. Pick some large constant c , and add it to the weight of each edge, so that there are no negative weights. Now, just run Dijkstra's algorithm to find all the shortest paths. Is this method correct? If yes, reason why. If not, give a counterexample.