Homework 9, CPSC 4100-01, Winter 2017

I have not received unauthorized aid on this assignment. I understand the answers that I have submitted. The answers submitted have not been directly copied from another source, but instead are written in my own words.

1) How can we use the output of the Floyd-Warshall algorithm to detect the presence of a negative-weight cycle?

Floyd-Warshall returns a matrix of the shortest hops between all the vertices in a graph. If we see a very large number in the matrix, then we know that a negative weight cycle exists because the algorithm was counting the hops through the cycle to try and find the shortest path between the two vertexes.

2) Diameter of a graph is the longest distance (in terms of number of hops) between any two vertices. Propose an efficient algorithm to determine the diameter of a given graph.

Floyd-Warshall will return all pairs shortest paths in a matrix. If we use a max-heap and insert all the path lengths, then we can find the two vertexes that are the farthest apart and that will be the diameter. Floyd Warshall runs in $O(n^3)$ and it would take O(logn) to insert the path lengths at worst case.

3) Professor Gaedel has written a program that he claims implements Dijkstra's algorithm. The program produces v.d and $v.\pi$ for each vertex $v \in V$. Give an $O(|V|+|E|)$-time algorithm to check the output of the professor's program. It should determine whether the d and π attributes match those of some shortest-paths tree. You may assume that all edge weights are nonnegative.

We can run a DFS in $O(|V| + |E|)$ to check the edges from the professors program by following the π value for each vertex and then if we relax all the edges in the graph and check to see if all the v.d values match then if at any point they do not match then we know that the professors results were incorrect.