Algorithms -1
Tutorial 7
November 2, 2018

For all the problems, first solve assuming that the graph is represented using an adjacency list. Then think what the time complexity of your algorithm will be if the graph is represented by an adjacency matrix.

1. Consider a weighted directed graph G. Suppose the weights of all the edges are increased by a fixed positive constant k. Will a shortest path from u to v in edge remain a shortest path from u to v in the modified graph for all u, v?
2. Consider a weighted directed graph G. Suppose the signs of the weights of all edges are reversed keeping the magnitude same (so 2 becomes -2 and -3 becomes 3 and so on). Will a shortest path in this modified graph give a longest path in G always?
3. Design an algorithm to find the second maximum weighted spanning tree of a weighted undirected graph. Try to make your algorithm as efficient as you can.
4. Suppose that the graph G = (V, E) is represented as an adjacency matrix. Give a simple implementation of Prim's algorithm for this case that runs in $O(V^2)$ time.
5. Let G = (V, E) be a weighted undirected graph. Let s and t be two distinct vertices in V. Design an O(ElogV) time algorithm to find all vertices v such that v lies on at least one of the shortest paths between s and t.
6. Can you detect a negative weight cycle in a directed graph using Floyd-Warshall's algorithm?
7. Can you find the transitive closure of a graph in $O(V^3)$ time? (A transitive closure of a graph G = (V, E) is another graph G1 = (V, E1) such that (u,v) € E1 if there is a path from u to v in G).
8. Describe an algorithm to find the length of the shortest cycle in a directed unweighted graph in $O(V^3)$ time.