Algorithms Lab 11

In lab

1. In class we talked about Bellman-Ford algorithm for computing SSSP on directed graphs and proved that $|V| - 1$ edge relaxation phases always suffice for computing any shortest path in G, if shortest paths exist. In order for shortest paths to exist (to be well-defined) the graph can have negative edges but no negative cycles.

2. What happens if we run an additional relaxation phase?

3. We know that if the graph has negative cycles SP do not exist. Suppose that we run Bellman-Ford SSSP(s) algorithm on a graph that has negative cycles. What will happen? Consider two cases:

   (a) A negative cycles is reachable from source vertex s.

   (b) No negative cycle is reachable from s.

4. Bases on your observation above, can you extend BF algorithm to determine if G has a negative cycle *reachable from* s?

5. Can you think of an approach to determine if a graph G has a negative cycle? Hint: augment the graph in some way (add vertices and dges) to get a graph G' and run BF on the modified graph from a specific vertex s. You want to argue that G' has a negative cycle if and only of G has a negative cycle, and if G has a negative cycle, it will be reachable from the vertex s in G'.

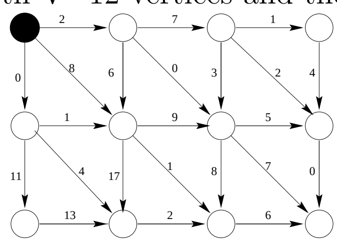
Homework

1. (CLRS 24.3-6) We are given a directed graph $G = (V, E)$ on which each edge (u, v) has an associated value $r(u, v)$, which is a real number in the range $[0, 1]$ that represents the reliability of a communication channel from vertex u to vertex v. We interpret $r(u, v)$ as the probability tht the channel from u to v will not fail, and we assume that these probabilities are independent. Give an efficient algorithm to find the most reliable path between two given vertices.

2. (GT C-7.7) Suppose you are given a diagram of a telephone network, which is a graph G whose vertices represent switching centers, and whose edges represent communication links between the two centers. The edges are marked by their bandwidth. The bandwidth of a path is the *minimum* bandwidth along the path. Give an algorithm

that, given two switching centers a and b, will output a maximum bandwidth path between a and b.

3. Consider a directed weighted graph with non-negative weights and V vertices arranged on a rectangular grid. Each vertex has an edge to its southern, eastern and southeastern neighbours (if existing). The northwest-most vertex is called the root. The figure below shows an example graph with V=12 vertices and the root drawn in black:



Assume that the graph is represented such that each vertex can access **all** its neighbours in constant time.

   (a) How long would it take Dijkstra's algorithm to find the length of the shortest path from the root to all other vertices?

   (b) Describe an algorithm that finds the length of the shortest paths from the root to all other vertices in $O(V)$ time.

   (c) Describe an efficient algorithm for solving the all-pair-shortest-paths problem on the graph (it is enough to find the length of each shortest path).