Practice Problem Set: SSSP-Dijkstra's Algorithm

Problem 1. Make a small example of directed weighted graph (3 vertices), where Dijkstra algorithm does not produce correct shortest paths, when some weights are negative.

Problem 2. Make a small example of directed weighted graph (3 vertices), where Dijkstra algorithm does produce correct shortest paths, even though there are negative weights.

Problem 3. Dijkstra algorithm always add the closest vertex to s in the unknown region $(\bar{R})$, prove that such a vertex must be one edge away from R.

Problem 4. Prove the following loop invariant about Dijkstra algorithm. At the end of each iteration, there is a value $d \geq 0$, such that all vertices in R are at distance at most d from s and all vertices outside R at distance at least d from s.

Problem 5. Prove the following loop invariant about Dijkstra algorithm. At the end of each iteration, for every vertex v the $d[v]$ is either ∞ or it is equal to the weight of a path from s to v, such that all intermediate nodes on this path are in R.

Problem 6. Suppose at iteration i we added v to R because (u, v) was the minimum score crossing edge $u \in R, v \notin R$. By the algorithm we would set $d[v]$ to $d[u] + w(uv)$. Prove that $d[v]$ is the length of a shortest path from s to v, i.e. prove that the length of any other path from s to v is at least as large as $d[v]$.

Problem 7. Suppose we keep vertices of $V \setminus R$ in a min-heap. Prove that through out the execution of Dijkstra's algorithm, we do at most $|V| - 1$ ExtractMin operation and at most $|E|$ DecreaseKey operations

Problem 8. Prove that the array $prev[.]$ computed by Dijkstra's algorithm, the edges $(v, prev[v])$ for all $v \in V$, form a tree.

Problem 9. Suppose we have a weighted graph where some weights could be negative. We decide to multiply each weight with -1 so as we can apply Dijkstra algorithm. Make a small example to show that this strategy will not work.

Problem 10. Another strategy to deal with negatives could be to shift them all. We find the smallest weight in the graph say it is M and add M to each edge weight. It is easy to see that all weights are now non-negative. Make a small example of graph, where even after doing this shifting, Dijkstra's algorithm doesn't output correct shortest paths.

Problem 11. Given a weighted digraph $G = (V, E, w)$ and a source vertex $s \in V$ such each vertex in $V \setminus \{s\}$ is reachable from s. We want to find directed paths $P(s, v)$ from s to v for all $v \in V \setminus \{s\}$ that minimizes the heaviest edge on P. Modify the Dijkstra's algorithm to solve this problem. Prove correctness of your algorithm. Does your algorithm work if some weights are negative?

Problem 12. Given a weighted digraph $G = (V, E, w)$ and a source vertex $s \in V$ such each vertex in $V \setminus \{s\}$ is reachable from s. We want to find directed paths $P(s, v)$ from s to v for all $v \in V \setminus \{s\}$. Define capacity of a path P to be $c(P) := \min_{(i,j)\in P} w(ij)$ (the smallest capacity that appears along P). We want to find paths from s to all v that are of maximum capacity. Modify the Dijkstra's algorithm to solve this problem, you may use Max-Heap and result from above problem.