Practice Graph Problems

Problem 1: Weighted Shortest Paths

Let G be an undirected weighted graph with weight function $w : E \to \{1, 2, 3, ..., k\}$, where k is a small constant. Let u be an arbitrary node in G. Provide an algorithm which computes the length of the shortest paths between u and any other node in the graph in time $O(k(n + m))$.

Problem 2: Multi-Source Shortest Paths

How do you compute shortest paths when the source is not a single vertex s, but instead a set S of vertices? More precisely, suppose you are given a weighted directed graph $G = (V, E, w)$ with non-negative edge weights, you are given a source set $S \subset V$, and you are given a single target node $t \epsilon V$. Your goal is to find the weight of the shortest path from some vertex $s \epsilon S$ to t. (In other words, among all choices $s \epsilon S$, you want to find the choice that results in the shortest shortest path from s to t.) Describe an efficient algorithm for computing this multi-source shortest path distance. Your algorithm should run in O(V lg V + E).

Problem 3: Festival Traffic Planning

During events in Greensboro, road closures and delays are caused by festival traffic and events. This makes it very difficult to figure out the best route to drive from one point to another.

To aid drivers, the city has implemented a system that will give the (correct) travel time on a given road segment, given a specific starting time within the festival. That is, given a road segment (e.g., the 600 block of N. Congress Avenue) and a time (e.g., 7pm on the Tuesday of the festival), the system returns a travel time for that road segment (e.g., 5 minutes). Travel times in the city obey two rules: (1) travel times are always positive and (2) for $t_1 < t_2$, $t_1 + f_e(t_1) \leq t_2 + f_e(t_2)$ (that is, you cannot arrive earlier by waiting to start later).

Given a graph that represents the city's street network, this system can be represented as a function $f_e(t)$ that, for a given edge e, returns the travel time for road segment e, assuming the driver enters the segment at time t.

Give an efficient algorithm to determine the earliest possible arrival time at a given destination d, given a starting location s and a departure time t_d. State and briefly justify the running time of your algorithm.