CS 170 Homework 6

Due **10/11/2023, at 10:00 pm (grace period until 11:59pm)**

1 Study Group

List the names and SIDs of the members in your study group. If you have no collaborators, you must explicitly write "none".

2 Adding Many Edges At Once

Given an undirected, weighted graph $G(V, E)$, consider the following algorithm to find the minimum spanning tree. This algorithm is similar to Prim's, except rather than growing out a spanning tree from one vertex, it tries to grow out the spanning tree from every vertex at the same time.

procedure FINDMST($G(V, E)$)
  $T \leftarrow \emptyset$
  while T is not a spanning tree **do**
    Let $S_1, S_2 \ldots S_k$ be the connected components of the graph with vertices V and edges T
    For each $i \in \{1, \ldots, k\}$, let e_i be the minimum-weight edge with exactly one endpoint in S_i
    $T \leftarrow T \cup \{e_1, e_2, \ldots e_k\}$
  return T

For example, at the start of the first iteration, every vertex is its own S_i.

For simplicity, in the following parts you may assume that no two edges in G have the same weight.

(a) Show that this algorithm finds a minimum spanning tree.

(b) Give a tight upper bound on the worst-case number of iterations of the while loop in one run of the algorithm. Justify your answer.

(c) Using your answer to the previous part, give an upper bound on the runtime of this algorithm.

3 Minimum ∞-Norm Cut

In the MINIMUM INFINITY-NORM CUT problem, you are given a connected undirected graph $G = (V, E)$ with positive edge weights w_e, and you are asked to find a cut in the graph where the largest edge in the cut is as small as possible (note that there is no notion of source or target; any cut with at least one node on each side is valid).

Solve this problem in $O(|E|\log|V| + |V| + |E|)$ time. **Give a 3-part solution.**

Hint: Minimum Spanning Tree does not require edge weights to be positive.

4 Firefighters

PNPLand is made of N cities that are numbered from $0, 1, \ldots N-1$, which are connected by two-way roads. You are given a matrix D such that, for each pair of cities (a, b), $D[a][b]$ is the distance of the shortest path between a and b.

We want to pick K distinct cities and build fire stations there. For each city without a fire station, the response time for that city is given by distance to the nearest fire station. We define the response time for a city with a fire station to be 0. Let R be the maximum response time among all cities. We want to create an assignment of fire stations to cities such that R is as small as possible.

Suppose the optimal assignment of fire stations to cities produces response time R_{opt}. Given positive integers N, K and the 2D matrix D as input, describe an $O(N^2 \cdot K)$ (or faster) greedy algorithm to output an assignment that achieves a response time of $R_g \leq 2 \cdot R_{\text{opt}}$. **Provide a 3-part solution.** For your proof of correctness, show that your algorithm achieves the desired approximation factor of 2.

Hint: $D[a][b]$ represents shortest (metric) distances. So you can use the triangle inequality: $0 \leq D[a][b] \leq D[a][c] + D[c][b]$ for all a, b, c.