Problem 1 Present full pseudocode of a variant of Prim’s algorithm that runs in time $O(|V|^2)$ for a graph $G = (V, E)$ given by an adjacency matrix $A$. Analyze the running time.

Note: do not use any procedure or data structures operation.

Problem 2 Give an example of a weighted directed graph $\vec{G}$ with negative-weight edges, but no negative-weight cycle, such that Dijkstra’s algorithm incorrectly computes the shortest-path distances from some start vertex $v$. Use the algorithm version from the handout.

A four-vertex example is possible. Draw the graph, mention the start vertex, show the result of Dijkstra’s algorithm, and point out for which vertex the result is incorrect.

(This was on a previous final exam)

Problem 3 A complete digraph has exactly one directed edge (also called arc) from every vertex $u$ to every vertex $v$ other than itself. Let $G$ be a complete digraph with non-negative arc weights. Let the capacity of a path be the minimum arc weight along it, and let the capacity of a pair of nodes $(u, v)$ be the maximum capacity of a path from $u$ to $v$. Find a Dijkstra-like algorithm to find, for all $v \neq s$, the capacity of $(s, v)$. (Node $s$ is a fixed source.)

Present the pseudocode, analyze the running time, and prove correctness.

Problem 4 Assume that the weights on the edges of directed graph $G$ are integers in the range from 1 to $K$. Give a new method for implementing EXTRACT-MIN($Q$) and DECREASE-KEY($Q, v, d[v]$) so that Dijkstra’s algorithm’s running time becomes $O(|E| + K|V|)$.

Describe the data structure and present the pseudocode for the two operations used by Dijkstra’s algorithm, together with correctness and running-time analysis.