

ECE 3020 Homework 11

Due Date: Friday, November 11, 2:00 PM

Using the high-level programming language of your choice, write a program that implements *both* Prim's Algorithm to determine a minimum-weight spanning tree and Dijkstra's Shortest Path Algorithm for an arbitrary graph. Your program should implement both algorithms within a single loop, i.e. do not simply run Prim's Algorithm all the way through followed by Dijkstra's Algorithm all the way through. You will be graded on how closely integrated the two algorithms are. Your implementation should *not* use a heap to find the minimum weight (distance) for Prim's (Dijkstra's) Algorithm, rather it should simply iterate through the nodes to find the minimum at each step. You may use either the adjacency matrix or adjacency lists representation for the graph – you will not be graded on the running time of your program.

Notes: The pseudocode for both algorithms is available on the course Web site. For both algorithms, you will need to add an additional data structure to track the edge that corresponds to the current minimum weight (distance) for each node.

Test your program on the graph below and a few others. Turn in your source code and output from the below graph with two different starting nodes: v_2 and v_6 . Each output should show the minimum-weight spanning tree from Prim's Algorithm and the shortest-path tree from Dijkstra's Algorithm, the total weight (the sum of the edge weights) for each tree, and the final values of the `min_weight` and `distance` arrays. A tree can be output by either showing its adjacency matrix or simply listing all of its edges.

