Your solutions must be submitted to Blackboard as a PDF document.

1. (10 pts.) Show a recursion tree for $T(n) = T(n - 1) + T(n - 2) + 1$. Provide upper and lower asymptotic bounds on $T(n)$.

2. Consider the weighted graph $G$ in Figure 23.1. Wherever there is a weight $w(u, v)$, replace that weight with $20 - w(u, v)$.

   (a) (10 pts.) Display this new weighted graph.

   (b) (10 pts.) Trace Kruskal’s Algorithm on the graph. That is, show the order in which edges are added to the minimum-spanning tree.

   (c) (10 pts.) Trace Prim’s Algorithm on the graph for two different roots, $a$ and $e$. That is, show the order in which edges are added to the minimum-spanning tree.

3. (10 pts.) Trace the Bellman-Ford Algorithm on the graph in Figure 24.4 using $z$ as the start vertex.

4. (10 pts.) Trace Dijkstra’s Algorithm on the graph in Figure 24.6 using $z$ as the start vertex.

5. (10 pts.) Suppose $(u, v)$ is the minimum-weight edge incident on $u$ in a graph $G$, where $G$ is undirected, connected, and weighted. Assume all weights are distinct. Show that $(u, v)$ belongs to some minimum spanning tree of $G$. Hint: If $T$ is a spanning tree without $(u, v)$, show to construct a spanning tree $T'$ that includes $(u, v)$ and has a lower total weight.

6. (10 pts.) Using the the Bellman-Ford algorithm as a subroutine, write an algorithm in pseudocode to determine if a directed graph $G$ contains a cycle. What is the running time of your algorithm?

7. (20 pts.) In pseudocode, write an algorithm to count all the simple paths in a graph, including paths of length 0. Don’t worry about creating an efficient algorithm. Hint: Write it recursively with one parameter equal to the vertices not in the current path. What is the running time of your algorithm?