**Exercises** (strongly recommended, do not turn in)

1. DPV 5.2 (Modified). Suppose we want to find the minimum spanning tree of the following graph.

   ![Graph Image]

   (a) Run Prim’s algorithm; whenever there is a choice of nodes, always use alphabetic ordering (e.g.,
   start from node A). Draw a table showing the intermediate values of the cost array. Show the
   final spanning tree.

   (b) Suppose Kruskal’s algorithm is run on this graph. In what order are the edges added to the MST?
   Show the final spanning tree.

2. Prove or disprove: Let $G = (V, E)$ be a weighted graph and let $T$ be a minimum spanning tree of $G$.
   The path in $T$ between any pair of vertices $u$ and $v$ must be a shortest path in $G$.

3. For each of the following statements, decide whether it is true or false. If it is true, give a short
   explanation. If it is false, give a counterexample.

   (a) Suppose we are given an instance of the MST Problem on a graph $G$, with edge costs that are
   positive and distinct. Let $T$ be a minimum spanning tree for this instance. Now suppose we
   replace each edge cost $c_e$ by its square $c_e^2$, thereby creating a new instance of the problem with
   the same graph but with different costs.
   True or False? $T$ must be a minimum spanning tree for this new instance.

   (b) Suppose we are given an instance of the Shortest Path Problem on a directed graph $G$. We assume
   that all edge costs are positive and distinct. Let $P$ be a minimum cost $s - t$ path for this instance.
   Now suppose we replace each edge cost $c_e$ by its square $c_e^2$, thereby creating a new instance of the
   problem with the same graph but with different costs.
   True or False? $P$ must still be a minimum-cost $s - t$ path for this new instance.

4. DPV 5.14. Suppose the symbols $a, b, c, d, e$ occur with frequencies 1/2, 1/4, 1/8, 1/16, 1/16 respectively.

   (a) What is the Huffman encoding of the alphabet?

   (b) If this encoding is applied to a file consisting of 1,000,000 characters with the given frequencies,
   what is the length of the encoded file in bits?

**Problems** (must be written up and turned in)

1. Given an edge-weighted undirected connected chain-graph $G = (V, E)$, all vertices having degree 2,
   except two endpoints which have degree 1 (there is no cycle). Design an algorithm that preprocesses
   the graph in linear time and can return the distance of the shortest path between any two vertices in
   constant time (i.e., the $O(|V|)$ preprocessing enables return of the shortest-path distance between any
   two vertices in $O(1)$ time). Give an English description of your algorithm along with pseudocode, and
   give an analysis of runtime complexity.
2. Recall interval scheduling problem from class (the interval scheduling problem is to find a largest compatible set - a set of non-overlapping intervals of maximum size). Suppose that instead of always selecting the first activity to finish, we select the last activity to start that is compatible with all previously selected activities. Describe how this approach is a greedy algorithm, and prove that it yields an optimal solution.

3. DPV 5.10. Let \( T \) be an MST of a weighted, undirected graph \( G \). Given a connected subgraph \( H \) of \( G \), show that \( T \cap H \) is contained in some MST of \( H \).

4. Given an edge-weighted undirected graph \( G = (V, E) \), such that \(|E| > |V|\) and all edge-weights are distinct.
   We define a second-best minimum spanning tree as follows. Let \( T \) be the set of all spanning trees of \( G \), and let \( T' \) be a MST of \( G \). Then a second-best minimum spanning tree is a spanning tree \( T \) such that \( W(T) = \min_{T'' \in T - \{T'\}}(W(T'')) \). \( W(\cdot) \) denotes weight of spanning tree.
   
   (a) Show that the minimum spanning tree is unique, but that the second-best minimum spanning tree need not be unique.
   
   (b) Let \( T' \) be the minimum spanning tree of \( G \). Prove that \( G \) contains edges \((u, v) \in T' \) and \((x, y) \notin T' \) such that \( T' - \{(u, v)\} \cup \{(x, y)\} \) is a second-best minimum spanning tree of \( G \).
   
   (c) Let \( T \) be a spanning tree of \( G \) and for any two vertices \( u, v \in V \), let \( \max(u, v) \) denote an edge of maximum weight on the unique simple path between \( u \) and \( v \) in \( T \). Design an \( O(|V|^2) \) time algorithm that, given \( T \), computes \( \max(u, v) \) for all \( u, v \in V \). Give an English description of your algorithm along with pseudocode, and give an analysis of runtime complexity.
   
   (d) Design an efficient algorithm to compute the second-best minimum spanning tree of \( G \). Give an English description of your algorithm along with pseudocode, and give an analysis of runtime complexity.

5. Given a graph \( G = (V, E) \), a subset \( S \subseteq V \) of vertices is said to be a vertex cover of \( G \) if for every edge \((u, v) \in E\), at least one of \( u, v \) belongs to the subset \( S \). A minimum vertex cover of \( G \) is a vertex cover with minimum cardinality (i.e., smallest vertex cover).
   Design an algorithm to find a minimum vertex cover of a given tree \( T = (V, E) \). Justify why your algorithm works, give pseudocode, and give an analysis of runtime complexity.

6. Given an edge-weighted directed graph \( G = (V, E) \) such that all the edge-weights are positive. Let \( s \) and \( t \) be two vertices in \( G \) and \( k \leq |V| \) be an integer. Design an algorithm to find the shortest path from \( s \) to \( t \) that contains exactly \( k \) edges. Give an English description of your algorithm along with pseudocode, and give an analysis of runtime complexity.
   Note that the path need not be simple, and is permitted to visit vertices and edges multiple times.