1. Suppose we perform a sequence of \( n \) operations on a data structure in which the \( i \)-th operation costs \( i \) if \( i \) is an exact power of 2, and 1 otherwise. Use the accounting method to determine the amortized cost per operation and use this to get an upper bound on the actual cost of the sequence of \( n \) operations. (15 points)

2. Consider the same set-up as in the previous problem but now use the potential method to determine the amortized cost per operation and use this to get an upper bound on the actual cost of the sequence of \( n \) operations. (15 points)

3. Exercise 19.4-1 on page 526. (10 points)

4. Consider a connected weighted graph where the edge weights are all distinct. Show that for every cycle of the graph, the edge of maximum weight on the cycle does not belong to any minimum spanning tree of the graph. (15 points)

5. Again consider a connected weighted graph where the edge weights are all distinct. Show that the minimum spanning tree must be unique but the second-best minimum spanning tree need not be unique. Is a shortest path between two vertices unique? (15 points)

6. Exercise 24.3-5 on page 663. (15 points)

7. In all the shortest paths algorithms we’ve learned in class we break ties arbitrarily. Discuss how to modify these algorithms such that, if there are several different paths of the same length, then the one with the minimum number of edges will be chosen. (15 points)