Algorithms and Data Structures: Final Exam

Name:

1. (5 points) What is the smallest value of n such that an algorithm whose running time is $10 * n^2$ runs faster than an algorithm whose running time is $5 * 2^{n-1}$?

2. (10 points) Use the merge sort algorithm to arrange the list (23, 27, 50, 2, 10, 73, 10) into increasing order. Write down all intermediate lists.

3. (5 points) Find the asymptotic complexity (the best "big-O" class) for the following functions:

   (a) $f(n) = 4n + 4$

   (b) $f(n) = n^2 + 3n^5 + 5$

   (c) $f(n) = (n + 1)(n - 3) + (n - 1)^2$

   (d) $f(n) = \frac{n^3}{n+1}$

   (e) $f(n) = \sum_{i=1}^{n} 3i$

4. (10 points) Find the running time (the number of computational steps) needed to execute the following fragments of pseudo-code.

   (a)
   ```
   for k = 1 to n do
     for j = 1 to k do
       x = x + 1
     end for
   end for
   ```

   (b)
   ```
   for i = 1 to n do
     y = y + 1
     for h = 1 to p do
       z = z + 3
     end for
     for k = 1 to q do
       x = x + 2
     end for
   end for
   ```

   (c)
   ```
   for k = 1 to n do
     for j = 1 to n^2 do
       x = x + x
     end for
   end for
   ```

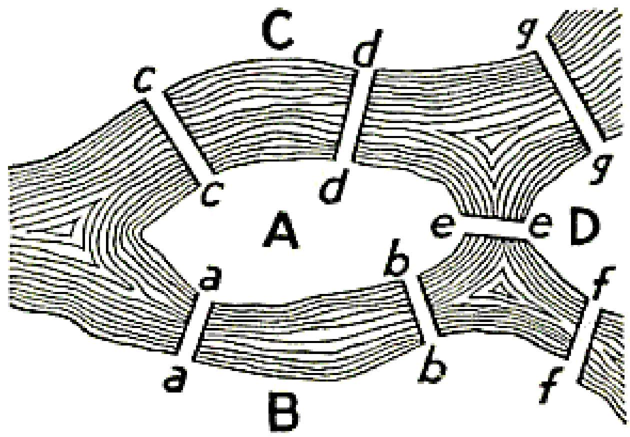

Figure 1: The Königsberg bridges a, b, c, d, e, f, and g.

5. (10 points) Use mathematical induction to prove:

$$\sum_{i=0}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$$

6. (20 points) An Eulerian path is a path in a graph which visits each edge exactly once. An Eulerian circuit is an Eulerian path which starts and ends on the same vertex. They were first discussed by Leonhard Euler while solving the famous "Seven Bridges of Königsberg" problem in 1736. The city of Königsberg in Prussia (now Kaliningrad, Russia) was set on both sides of the Pregel River, and included two large islands which were connected to each other and the mainland by seven bridges. The problem was to find a walk through the city that would cross each bridge once and only once.

   (a) Euler discovered that a graph has an Eulerian circuit whenever all vertices in the graph have an even degree. Provide a pseudo-code algorithm that checks for any given graph if the degree of all its nodes is even. Your algorithm should represent a graph with an adjecency list. It should return **true** if the degree of each vertex in the graph is even and **false** otherwise.

   (b) Given the depiction of the bridges of Königsberg in Figure 1 draw a graph G that represents the situation. The graph will have 4 vertices (A, B, C, D) and 7 edges (modeling the bridges a, b, c, d, e, f, and g).

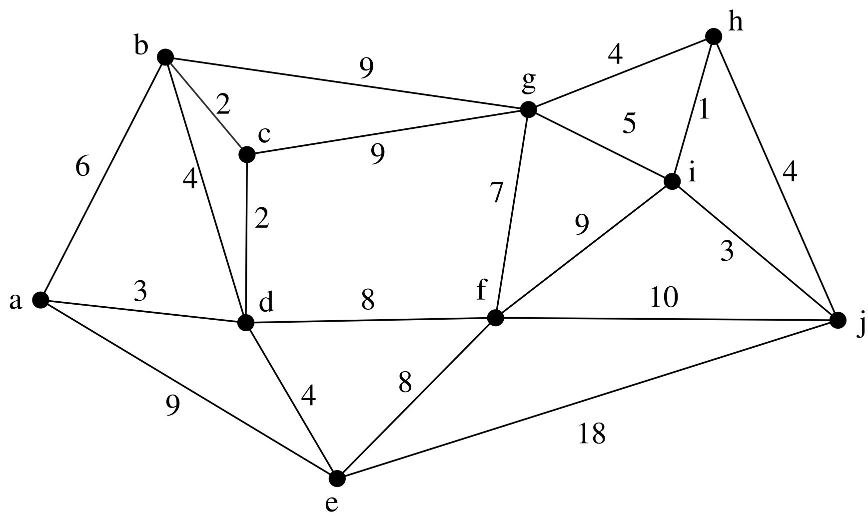

Figure 2: The graph for the minimum spanning tree problem.

(c) Given the graph G you constructed in the previous step the "Seven Bridges of Königsberg" problem can be stated like this: "Is it possible to construct a cycle in graph G which visits each edge exactly once?" Apply your algorithm from subquestion (a) to solve the "Seven Bridges of Königsberg" problem.

7. (20 points) Recall that a minimum spanning tree (MST) or minimum weight spanning tree is a spanning tree with weight less than or equal to the weight of every other spanning tree.

   (a) Kruskal's algorithm computes for a given connected undirected edge-weighted graph a minimum spanning tree. Provide the pseudo-code for Kruskal's algorithm.

   (b) Apply Kruskal's algorithm to the graph in Figure 2. Illustrate every step and write down the state of all the sets that are maintained by Kruskal's algorithm.

8. (20 points) Consider inserting the keys 10, 22, 31, 4, 15, 28, 17, 88, and 59 into a hash table of length $m = 11$ using open addressing with the auxiliary hash function $h'(k) = k \bmod m$. Illustrate the result of inserting these keys using

   (a) linear probing;

   (b) quadratic probing with $c_1 = 1$ and $c_2 = 3$; and

   (c) double hashing with $h_2(k) = 1 + (k \bmod (m - 1))$.