B1 Algorithms (25 points)

Do any 3 of the 4 parts. If you answer all 4 parts, only the first 3 will be graded. Be sure to address all issues raised in each question.

1) First, explain the main idea behind Strassen's algorithm. Second, state the number of scalar multiplications used by Strassen's algorithm when multiplying two 16 by 16 matrices. Your answer should be a single number (not a number raised to a power).

2) Sketch the main ideas in an algorithm that takes as input a string S and a pattern P, and returns the index of the first occurrence of P in the string SP. The algorithm that you sketch should have a linear worst case running time.

3) First, carefully describe Prim's algorithm, including how you implement the main steps. Also explain the running time and how it relates to the implementation. Second, suppose there was a data structure SUPERHEAP for which the operations Insert, Delete_min, Union, and Decrease_Key all ran in time O(1). What would be the running time of Prim's algorithm using the SUPERHEAP data structure?

4) First, very briefly state the two main principles of dynamic programming. Second, sketch a dynamic programming algorithm to solve a problem of your choosing. Be sure to give the main idea and the running time.

B2 Algorithms (25 points)

Do all three parts.

a) [5 points] Suppose that Dr. Einstein develops a polynomial time algorithm for the problem 3SAT. Carefully explain why this would also mean that there is a polynomial time algorithm for the Traveling Salesperson Problem. Be sure to briefly define any technical terms that you use.

b) [10 points] For each of the following problems, state whether or not it is NP-complete. If it is not NP-complete, then sketch (one or two lines) a linear time algorithm for solving the problem. If it is NP-complete, do nothing more.

i) Given an undirected graph G, does G contain a vertex cover of size 3?
Recall: a vertex cover of G is a set S of vertices such that every edge of G is incident to at least one of the vertices in S.

ii) Given an undirected graph G, does there exist a 2-coloring of G?

iii) Given a set of positive numbers $M = \{x_1, \ldots, x_n\}$, an integer k, and a positive number B, does there exist a subset Q of M such that the sum of the numbers in Q is at least B and such that $|Q| <= k$?

iv) Given a graph G and an integer k, does G contain a clique of k vertices?

c) [10 points] It is well known that DIRECTED HAMILTONIAN CIRCUIT (DHC) is NP-complete. In DHC the instance is a directed graph G', and the question is "Does G' contain a directed circuit that contains each vertex of G' exactly once?".

Give a complete proof that the following problem is NP-complete. You may assume that DHC is NP-complete.

DHP: DIRECTED HAMILTONIAN PATH

INSTANCE: A directed graph G=(V,E).

QUESTION: Does G contain a directed ***path*** that contains each vertex of G exactly once?

Just to be clear, a directed Hamiltonian path cannot be a circuit.

B3 Algorithms (25 points)

Do all four parts.

a. [10 points] For each of the following algorithms, describe the main idea/ideas, and state the running time, if it is not given.

   i) Heapsort.
   ii) Selection in worst-case linear time.

b. [6 points] Give an **in-place linear time** algorithm to rearrange an array of n elements so that all the negative keys come before all the zero keys, and all the zero keys come before all the positive keys. It may happen that there are no elements in one or two of the three classes.

c. [6 points] Give an algorithm to determine if the n keys in an array are all distinct. Assume three-way comparisons; that is, the results of a comparison of two keys is <, =, or >. How many key comparisons does your algorithms use?

d. [3 points] Give a lower bound on the number of three-way key comparisons needed in part 3. Explain your reasoning.

B4 Algorithms (25 points)

Do both part a and part b.

a. [6 points] Select any 2 of the following 3 items (do only 2 – if you do all 3, only the first 2 will be graded). For each tree that you select you should describe: the structure of the tree when used as a dictionary; the order associated with the items sorted in the tree; and how an Insert operation is performed on that tree (if the Insert involves a rotation, you need only say that there is a rotation AND what the rotation is designed to accomplish. You do not have to give the details of the rotation). Do not describe how to perform any operation other than Insert.

- 2-3 tree
- Red-Black tree
- AVL tree

b. Fibonacci Heaps

i) [3 points] What is a Fibonacci Heap? (i.e., explain the data structure)

ii) [3 points] What is meant by an amortized running time?

iii) [4 points] In a Fibonacci Heap, state the worst case and amortized running times for these operations: Insert, Min, Delete_min, Union, Decrease_Key.

iv) [3 points] Explain the implementation of a decrease key operation in a Fibonacci Heap.

v) [6 points] Give an example of an algorithm that uses Fibonacci Heaps – explain how Fibonacci Heaps are used there, and how they figure in the running