CS 1510 Midterm 3
Fall 1995

1. Show that the problem of squaring an $n$ by $n$ matrix can't be solved any faster (ignoring multiplicative constants) than the problem multiplying two arbitrary $n$ by $n$ matrices. Write your explanation for a fellow computer science student who has had the benefit of taking this course.

2. (a) Give an example of a randomized algorithm that was discussed in the book or in class.
   (b) Define what is meant by the term "randomized algorithm". Specifically explain how it is different than a deterministic algorithm.
   (c) Define what is meant by "the average case running time of a deterministic algorithm".
   (d) Define what is meant by "the worst-case expected case running time of a randomized algorithm".
   (e) From a practical point of view which would you prefer, a deterministic algorithm $A$ with an average case run time of $\Theta(n^2)$, or a randomized algorithm $A$ with worst case expected run time of $\Theta(n^2)$. Explain.

3. The input to this problem is $n$ intervals $(a_1, b_1), \ldots, (a_n, b_n)$ on the real line. The problem is to find a maximum cardinality nonintersecting subcollection of these intervals. Give a greedy algorithm for this problem, and prove that it is correct. Note: Please emphasize the main ideas in the proof and be as brief as possible.

4. Consider the following problem. The input is $n$ points in the plane. The output is the shortest path that visits all of the points starting from the origin. Show that this problem has an $\Omega(n \log n)$ information theoretic lower bound.

5. Assume that you are given a list $P$ of $n$ people $p_1, \ldots, p_n$, and a list of $m$ jobs $j_1, \ldots, j_m$. Each person $p_a$ has a limit $L_a$ on the number of hours that they can work. Each job $j_b$ has a requirement $R_b$ of the number of man hours that must be devoted to this job in order for it to be finished. For each person $p_a$ and each job $j_b$ you are given an hourly salary $c_{a,b}$ that you must $p_a$ to work at $j_b$. Assume that all variables are real numbers. The goal is to find the cheapest way to assign people to tasks so that all tasks are completed, and all the above constraints are satisfied. The cost of an assignment is just the total amount of money you have to pay all of the people. You need not worry about scheduling when people work on tasks, you need only be concerned about the total time devoted to each tasks. Show that an efficient algorithm exists for this problem by making use of the fact that an efficient algorithm exists for linear programming.