Instructions:

1. The test is closed book, closed notes.

2. For most of the problems, I am interested in testing whether you understand the techniques and concepts more than I am interested in the solution to the particular problem. For example, if I ask you to prove that a problem is NP-hard, I am more interested in learning if you know how to prove that a problem is NP-hard, than I am in the specifics of the problem. If I ask you to prove that a greedy algorithm is correct using an exchange argument, I am more interested in learning if you know how an exchange argument works, than I am in the specifics of the problem. I ask these questions in the context of specific problems to allow you to demonstrate your understanding in a concrete setting. Of course you have to take into account the specifics of the problem, but make sure to explain the general method/technique/concept that you are using as well.

3. 25% partial credit is given for the answer “I don’t know.” A blank answer will be interpreted as “I don’t know.” An answer that displays a major conceptual error will likely receive a grade of zero. It is perfectly fine to give an incomplete answer, e.g. “Here is how one proves a problem NP-hard, but I don’t know how to prove this problem NP-hard.” I will make a judgement call on how much credit such answers should receive.

4. I will assume that if you write something, that you are asserting that you have good confidence in the correctness of what you write. It is a bad strategy to give an answer that you do not have good confidence in.

5. Answer up to 4 of the problems. Clearly write the number of the problem that you are answering. If you answer more than 4, an arbitrary 4 answers will be graded.

6. If you are uncertain about anything, ask a question!
1. Prove using an adversarial argument that every comparison based algorithm for solving the Element Uniqueness problem requires $\Omega(n \log n)$ comparisons for $n$ numbers.

2. Give a polynomial time algorithm to find the shortest bitonic tour of a collection of points in the Euclidean plane (under the assumption that distances between pairs of points can be computed exactly). Recall that a bitonic tour is one that starts at the leftmost point, goes strictly rightward to the rightmost point, and then goes strictly leftward back to the starting point.

3. Let $x_1, \ldots, x_n$ be Bernoulli trials, where with probability $1/2$ a trial is a success. Let $y_i$ be the number of trials one has to observe, starting from $x_i$, until one encounters a failure. That is, $y_i$ is the minimum positive integers such that $x_{i+y_i-1}$ is a failure. Show $E[\max(y_1, y_2, \ldots, y_n)] = O(\log n)$.

4. Give a polynomial sized linear program for the minimum weight spanning tree problem. Make sure to explain why your linear program is correct.

5. Suppose that we are given $n$ currencies $c_1, c_2, \ldots, c_n$ and an $n$ by $n$ table $R$ of exchange rates such that one unit of currency $c_i$ buys $R[i, j]$ units of currency $c_j$. Give a polynomial time algorithm for arbitrage problem, that is, to determine one can make a profit by exchanging currencies. Completely describe the algorithm, and explain why it is correct.

6. Give a polynomial time algorithm for the Weighted Vertex Cover Problem, and prove that it has approximation ratio 2. Recall that the input for the Weighted Vertex Cover problem is a graph with positive integer weights on the vertices, the feasible solutions are vertex covers, and the objective is to minimize the aggregate weight of the selected vertices.

7. Prove that if there is a polynomial time approximation algorithm for the maximum clique problem that has approximation ratio 1000 then there is a polynomial time approximation algorithm with approximation ratio 1.000000001.

8. Consider the following online problem. There are two taxis on a line that initially start at the origin. At positive integer time $t$, a request point $h_t$ on the line arrives. In response, each taxi can move to a different location on the line, or stay put at its current point. The path traveled by at least one of the two taxis must cross $h_t$. The objective is to minimize the total movement of the taxis. Consider the following algorithm $A$. If both taxis are to the left of $h_t$, then a rightmost taxi moves to $h_t$. If both taxis are to the right of $h_t$, then a leftmost taxi moves to $h_t$. If $h_t$ is between the two taxis, then both taxis move toward $h_t$ at the same rate until one of the taxis reaches $h_t$, at which point both taxis stop moving. Show that this algorithm is 2-competitive using the following potential function: $\Phi = \text{twice}$
the distance between the leftmost taxi for $A$ and the leftmost taxi for optimal plus twice the distance between the rightmost taxi for $A$ and the rightmost taxi for optimal plus the distance between the leftmost and the rightmost taxis for $A$.

9. Show using Yao’s technique that the expected competitive ratio for every randomized paging algorithm is $\Omega(\log k)$ with respect to the objective minimizing the number of page faults. Start by stating the version of Yao’s technique that you will use.

10. (a) Explain the setup for the online learning from experts problem that we discussed in class. In particular, explain what the input is that the algorithm sees and when it sees it this input, what the algorithm outputs and when it produces this output, and what the objective is.

(b) Describe the Weighted Majority algorithm.

(c) Give the analysis for the weighted majority algorithm that we discussed in class.