

Final Exam

December 18, 2002

READ THIS PAGE. DO NOT TURN TO THE NEXT PAGE UNTIL YOU ARE TOLD YOU CAN START.

- NAME: _____
- This is the CS 441T exam. If you are a CS 539T student please raise your hand so I can give you the CS 539T exam.
- If any question is not clear to you, come up to the front and I will answer any questions you have. There is no point cost (unlike with the hints) if you want to ask for clarification about the problem itself.
- Please write up all solutions clearly and legibly in the space provided. If needed use the back of the last page of the exam for extra space to write-up your solution for any problem. If you need scratch paper please come up front.
- When you are asked to give a lower bound, spend at most 15 minutes determining the best bound you can prove and then write-it up. Your write-up should have the following steps:
 - State the lower bound which you will prove.
 - Clearly describe your adversary strategy.
 - Prove the stated lower bound.
 - (OPTIONAL) Describe an alternate adversary strategy that you think is better but for which you can't prove any better bound. Also feel free to give a few sentences of thoughts about how you might prove a better bound with it.
- For any problem in which you want to prove a problem is NP-hard, you must use one of the following for your reduction: CIRCUIT-SAT, SAT, 3-CNF-SAT, CLIQUE, VERTEX-COVER, PARTITION, SUBSET-SUM, HAM-CYCLE, TSP. If you would like me to write the definition for any of these problems, I will (for no point deduction). Just come to the front.
- For any problem if you feel that you are stuck and need a hint, please come to the front and for an appropriate reduction in the maximum score you can get on that problem a hint will be provided.
- Relax and begin working when instructed to start.

1. (10 points) Both Randomized Quicksort and the Miller-Rabin primality testing algorithm are randomized algorithms. Describe the important distinction between them in terms of the guarantees of correctness and the time complexity for these two different styles of randomized algorithms. Be very explicit in your answer.

2. (20 points) Consider the following on-line problem. You are given a sequence of prices $\langle p_1, p_2, p_3, \dots, p_m \rangle$ all in the range, $\ell \leq p_i \leq h$. When presented with p_i you can either buy the item for that price, or pass in which case you have committed to buying that item at price among p_{i+1}, \dots, p_m . Your goal is to buy the item for the cheapest price. Here is a deterministic on-line algorithm A for this problem. Accept the first p_i such that $p_i \leq \sqrt{\ell h}$. (If all $p_i > \sqrt{\ell h}$ then A accepts p_m .)

What is the competitive ratio of this on-line algorithm? Be sure to prove that the on-line algorithm has the competitive ratio that you give.

Hint: Consider two cases based on whether the optimal solution pays a price $\geq \sqrt{\ell h}$.

3. (20 points) Consider the following problem. For input you are given an undirected graph $G = (V, E)$ with two specified vertices v_s and v_e . The question is whether or not there is a Hamilton path in G that begins with v_s and ends with v_e . That is, the path must begin with v_s , visit each vertex in G exactly once ending with v_e .

(a) Give a graph G that does not have a Hamilton cycle, but for which there is a Hamilton path.

(b) Prove that given problem (determining if there is a Hamilton path starting at v_s and ending at v_e) is NP-complete.

More Space for Problem 3

4. (25 points) Consider the following problem. You are given a sequence $\langle s_1, s_2, \dots, s_n \rangle$ of integers (possibly negative). You are to give a polynomial-time algorithm to find a subset S' such that $\sum_{i \in S'} s_i$ is maximized under the constraint that S' cannot contain s_j and s_{j+1} for $j = 1, \dots, n - 1$. You should clearly describe your algorithm, prove it is correct, and analyze the time complexity.

5. (25 points) Give the best lower bound for the following problem. There is a bit string of n bits that is hidden from the algorithm and the algorithm can only learn about the bit string by asking a question of the form “What is the value of the i th bit?” Give the best lower bound you can on the number of such queries that must be made for the problem of determining if a bit string has 2 consecutive 1s.

Additional Work for Problem _____

Problem	Points Possible	Points Received
1	10	
2	20	
3	20	
4	25	
5	25	
total	100	