CS570

Analysis of Algorithms

Fall 2010

Final Exam

Name: ______________________

Student ID: __________________

____Monday Section ____Friday Section ____DEN Section

	Maximum	Received
Problem 1	20	
Problem 2	20	
Problem 3	20	
Problem 4	20	
Problem 5	20	
Total	100	

2 hr exam
Close book and notes

If a description to an algorithm is required please limit your description to within 200 words, anything beyond 200 words will not be considered.

1) 20 pts
Mark the following statements as **TRUE** or **FALSE** (or **UNKNOWN** if given as an option). No need to provide any justification.

[TRUE/FALSE]
For a given problem with input size n, there must be some N that when n>N, a $\Theta(n\log n)$ algorithm that solves this problem runs faster than a $\Theta(n^2)$ algorithm that also solves it.

[TRUE/FALSE]
For a given problem with input size n, there must be some N' that when n>N', a $\Theta(n\log n)$ algorithm that solves this problem runs slower than a $\Theta(n^2)$ algorithm that also solves it.

[TRUE/FALSE]
A problem may require memory space which is not polynomial with respect to the input size but may still have polynomial run time.

[TRUE/FALSE]
If two minimum spanning trees on the same graph only have 2 edges in common, then those two edges must be the lowest costs edges in the graph.

[TRUE/FALSE/UNKNOWN]
Any problem in NP is polynomial time reducible to integer programming.

[TRUE/FALSE/UNKNOWN]
Any problem in P is polynomial time reducible to linear programming.

[TRUE/FALSE]
Given a flow network, if one needs to delete one edge to reduce the original maximum flow by the most, this edge must be crossing a minimum cut in the original graph.

[TRUE/FALSE/UNKNOWN]
If there is a polynomial time algorithm for some problem in NP, then all problems in NP can be solved in polynomial time.

[TRUE/FALSE]
To prove a problem is NP-hard, you have to give a polynomial time certificate and certifier.

[TRUE/FALSE]
If you are given a maximum s-t flow in a graph then you can find a minimum s-t cut in time O(m) where m is the number of the edges in the graph.

2) 20 pts

In scheduling courses for a semester, one has to take a lot of things into account. One of them is avoiding overlap: it would be a pity if many students wanted to take CS570 and CS551 in the same semester and could not because they are at the same time. One way to address this would be to query the students ahead of time on which courses they want to take, and to schedule accordingly.

Formally, there are n courses and m students. For each student j, there is a set S_j of courses the student would like to take. We assume that there is only one time slot, and we want to schedule as many courses as possible in this time slot, but such that no student is planning to take two courses scheduled in this slot. Formulate this problem as a decision problem, and prove that it is NP-complete.

3) 20 pts

A spammer is located at one node q in an undirected communication network G and peaceful email users are located at nodes denoted by the set S. Let c_{uv} denote the effort required to install a spam filter for the network edge (u, v). The problem is to determine the minimal effort required to isolate the spammer from the email users using the spam filters. Find an efficient polynomial time algorithm to solve this problem.

4) 20 pts

To get in shape, you have decided to start running to the university. You want a route that goes entirely uphill and then downhill so that you can work up a sweat going uphill and then get a nice breeze at the end of your run as you run faster downhill. Your run will start at home and end at the university and you have a map detailing the roads with m road segments (any existing road between two intersections) and n intersections. Each road segment has a positive length, and each intersection has a distinct elevation.

Assuming that every road segment is either uphill or downhill, give an efficient algorithm to find the shortest route that meets you specifications.

5) 20 pts

A furniture manufacturer makes two types of furniture – chairs and sofas. The production of the sofas and chairs requires three operations – carpentry, finishing, and upholstery. Manufacturing a chair requires 3 hours of carpentry, 9 hours of finishing, and 2 hours of upholstery. Manufacturing a sofa requires 2 hours of carpentry, 4 hours of finishing, and 10 hours of upholstery. The factory has allocated at most 66 labor hours for carpentry, 180 labor hours for finishing, and 200 labor hours for upholstery. The profit per chair is $90 and the profit per sofa is $75.

A-) Formulate the problem as a linear programming problem. The goal is to determine how many chairs and how many sofas should be produced each day to maximize the profit.

B-) Show constraints graphically.

C-) Solve the problem in order to maximize the profit.