Comprehensive Exam: Algorithms and Concrete Mathematics Autumn 2007

This is a one hour closed-book exam and the point total for all questions is 60.

In questions that ask you to provide an algorithm, please explain the algorithm in words and diagrams, no need to write code or pseudo code. Also, for any algorithm, prove correctness and prove its running time. No credit will be given for exponential-time algorithms. Polynomial but slow algorithms will get some partial credit. Amount of credit will depend on how much slower they are compared to what is achievable using the knowledge in the reading list. Correct and fast algorithms with incomplete proof of correctness will get small number of points.

For full credit, the answers should be short and precise. Long and convoluted answers will not get full credit even if they are correct.

1. **[14 pts]** Please answer "true" or "false" to each one of the following questions. Correct answers will give you **(2 pts)** each while wrong answers will reduce your score by **(2 pts)** each.

   (a) There exists $\epsilon > 0$ such that $n^\epsilon = O(\log n)$.

   (b) Suppose $T(n)$ is given by the recurrence $T(n) = T(\lfloor n/\log n \rfloor) + n$; $T(1) = 1$. Then, $T(n) = \Theta(n)$.

   (c) Given a heap data structure (organized with minimum on top), it is possible to find the second-smallest element in constant time.

   (d) There exists a deterministic linear time algorithm that, given two English words with at most n characters each, determines whether they are anagrams of each other.

   (e) In comparison model, the lower bound on finding median of n input numbers is $\Omega(n \log n)$.

   (f) You are given a graph $G = (V, E)$ with positive lengths on edges and a shortest path P from $v \in V$ to $u \in V$. Next, lengths are transformed by computing square of the length of each edge. (If old edge length was w, new one is w^2). Claim: it is guaranteed that P is a shortest path from v to u with respect to new lengths as well.

   (g) Given a weighted graph and two nodes, it is possible to list all shortest paths between these two nodes in polynomial time.

2. **[18 pts]** You have a single classroom and a list of lectures that you would like to use the classroom for. Each lecture is specified by start time and end time. Your goal is to come up with a schedule that maximizes the number of lectures scheduled in the classroom under the constraint that no two scheduled lectures can overlap. For example, if one lecture needs the 1pm to 2pm slot and the other one needs 1:30pm to 3pm slot, at most one of these lectures can be scheduled. Design an efficient algorithm to solve the problem. Prove correctness and running time.

3. **[12 pts]** Prove that if edge weights of a graph are unique (no two edges have the same weight), then there is unique solution to the minimum-cost spanning tree problem.

4. **[16 pts]** Given an acyclic directed graph $G = (V, E)$ and a node $s \in V$, describe an algorithm to find the number of paths from s to each one of the nodes in V.