Examination "Breadth"

Theory

September 17, 2001

SOLVE EACH PROBLEM ON A SEPARATE PAGE. WRITE ONLY ON ONE SIDE OF A PAGE. PUT DOWN YOUR CODE ON EACH PAGE. YOU HAVE THREE HOURS TO ANSWER THE QUESTIONS.

1. (a) Give an example of an NP-complete set.
   (b) What is known about the time and space complexities of NP-complete sets?
   (c) Are there sets in NP for which more is known about their time and space complexities than is known about the time and space complexities of NP-complete problems? Explain.

2. For each of the following, sketch a proof or give a counterexample.
   (a) The class of regular languages is closed under intersection.
   (b) The class of context-free languages is closed under intersection.

3. Let $S = (a_1 \dots a_n)$ be a sequence of n integers, not necessarily distinct . For any i, let $N(i)$ be the number of elements in S that are greater than or equal to a_i. For a fixed k, $0 \leq k \leq n$, let d be an index so that $N(d) = k$. Design an algorithm that lists all numbers a_i from S such that $a_i \geq a_d$. Give a high-level description of the algorithm and pseudocode, and a complexity analysis (in terms of either worst-case or average-case complexity as a function of n and k). The analysis should include both a big -O function and justification. Grading will be based on correctness and on the efficiency of your algorithm.

4. Give a definition of a binary heap for a set of n elements. Provide description and pseudocode for Insert(Element x) operation. In the description you may use standard operations for binary heaps. Show a binary heap (as a tree or as an array) for a priority queue supporting RETURN_MAX() operation for the following set of characters (with the standard alphabetical order): {B, I, N, A, R, Y, H, E, A, P, S }.

5. Let $K(h)$ denote the number of binary rooted trees with height exactly h. State (but do not solve) a recurrence relation for $K(h)$. Explain why this is a correct recurrence relation. Based on your recurrence, what is $K(3)$?

SOLVE THE FOLLOWING PROBLEM.

6. Let $G = (V, E)$ be a directed acyclic graph with nonnegative edge weights $w(u, v)$, $u, v \in V$, and a pair of distinguished vertices $s, t \in V$, with s a source and t a sink. Recall that a flow in G is a real valued function on $V \times V$ such that (1) for all $u, v \in V$, $f(u, v) \leq w(u, v)$; (2) for all $u, v \in V$, $f(u, v) = -f(v, u)$; and (3) for all $u \in V - \{s, t\}$, $\sum_{v \in V} f(u, v) = 0$. The total flow of f is $|f| = \sum_{v \in V} f(s, v)$.

   Describe an efficient algorithm that finds the flow with largest possible total flow in a given graph G.