Binary search trees, Greedy algorithms

November 3, 2009

Homework 4

Due Date: Tuesday, 10 November 2009, 2:05pm

Problem 4-1. 20pts
Suppose that a search for key k in a binary search tree ends up in a leaf. Consider three sets: A, the keys to the left of the search path; B, the keys on the search path; and C, the keys to the right of the search path. Provide a small counterexample that disproves the following claim: *Any three keys* $a \in A, b \in B, c \in C$ *must satisfy* $a \leq b \leq c$.

Problem 4-2. 15pts
An alternative method of performing an inorder tree walk of an n-node binary search tree finds the minimum element in the tree and then finding its $n-1$ successors, one at a time. In other words, if minimum is a, then first find a, then find b =successor of a, then c=successor of b, etc. Prove that this algorithm runs in $\Theta(n)$ time.

Problem 4-3. 20pts
Suppose that we have a set of activities to schedule among a a large number of lecture halls, where any activity can take place in any lecture hall. Each activity is defined by “start time” and “end time” and cannot be rescheduled. One one activity can be scheduled in a lecture hall at a time. We wish to schedule all of the activities *using as few lecture halls as possible*. Give an efficient algorithm to determine which activity should use which lecture hall. Do not forget to prove that your algorithm indeed produces optimum solution and analyze its running time.

Problem 4-4. 10pts
During the lecture we have considered the ”activity selection problem”, where the goal is to schedule as many non-overlapping activities as possible. Now we will modify the problem, by adding ”revenue” value r_i to activity i, for all i. The modified goal is to find a set of non-overlapping activities S that maximizes ”total revenue”, defined as $\sum_{i \in S} r_i$. Is the algorithm presented in class correctly solves this problem ? Prove if yes and provide a counterexample if no.

Problem 4-5. 20pts
Consider a highway with gas stations located at distances $a_1, a_2, \ldots, a_n$ along the highway. In other words, the first gas station is a_1 miles from your current location, the next gas station is a_2 miles from your current location, and so on. You can travel R miles on a single gas tank. The goal is to design an algorithm that allows you to travel to the last gas station a_n while minimizing the number of times you have to refuel along the way. As usual, analyze running time and provide proof of correctness. [Proof of correctness is the main point of this exercise.]

Problem 4-6. 15pts
Binary tree is called ”full” if every non-leaf node has two children. Prove that non-full binary tree cannot correspond to an optimum prefix code.