Computer Science 2300: Homework 5

Due: March 30, 2009

Note: Please use rigorous, formal arguments. If you are asked to provide an algorithm then you may either write pseudocode similar to the pseudocode in the DPV text, or provide a clear description in English. You **must** also provide an argument for why the algorithm is correct, and an analysis of the running time. We encourage you to collaborate with other students, while respecting the collaboration policy. Please write the names of all the other students you collaborated with on the homework. **Hardcopies are required by submission time. E-mailed versions will not be accepted.**

1. (15 points) CLRS Problem 17.3-3 (page 416). [Hint: Consider using a potential function of the form $kn \ln n$].

2. (10 points) CLRS Problem 19.2-8 (pages 472-473).

3. (10 points) DPV Problem 1.29 (page 40)

4. (10 points) Consider the recurrence relation for the minimum number of nodes in an AVL tree: $T(n) = T(n-1) + T(n-2) + 1$ for $n > 1, T(0) = 1, T(1) = 2$. Show that $T(n) = \Theta(F(n))$ where $F(n)$ denotes the nth Fibonacci number.

5. (15 points) Consider a simplified version of the substring matching problem. Given a length n string (called the *target* T) composed exclusively of decimal digits 0 through 9, and a (smaller) length m string (called the *pattern* P) also composed only of decimal digits, we want to find out whether or not P occurs as a substring of T. If we find P as a substring of T we can immediately return "Yes" and terminate.

   (a) Write down a naive $O(mn)$ algorithm for this problem.

   (b) Assuming that arithmetic operations on m-digit numbers take constant time, design an $O(n)$ algorithm for this problem.

   (c) Now consider what happens as m gets larger. Can you use the idea of universal hashing to come up with an algorithm that works in expected time $O(n)$ for this problem? (Hint: work modulo some prime q so that $10q$ fits within one computer word – you can assume that arithmetic operations on numbers that fit within one computer word can be performed in constant time. You may also assume that the probability that a random number is congruent to some given p modulo q is $1/q$). Give an analysis of the expected running time if the pattern does not exist in the target. What is the worst case running time of this algorithm?