

Computer Science 2300: Homework 5

Due: March 25, 2008

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. (10 points) Weiss Problem 6.13, parts a, b, and c (page 253) (this is deletion of the minimum element in a binary heap).
2. (5 points) Weiss Problem 4.45 (page 180), except replace the word “function” with the word “algorithm.” Provide pseudocode or a clear description of the algorithm in English, **not C++ code**. Analyze the running time.
3. (10 points) Weiss Problem 11.8 (page 514).
4. (10 points) Weiss Problem 11.16 (page 515).
5. (10 points) DPV Problem 1.29 (page 40)
6. (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?