CS 3343 (Spring 2008) Assignment 5

Due: March 5 before class starts

1. (40 points) Quick sort.

   a. (10 points) Study the pseudocode of the Partition algorithm on Slide #12 (Lecture 11). Use Slide #13 as a model, illustrate the operation of Partition on array A = [13 19 9 5 12 8 7 4 21 2 6 11]. Importantly, indicate where is the pivot element when the algorithm terminates. There is no need to use colors.

   b. (5 points) Redo 1(a) on array B = [1 2 3 4 5 6 7 8 9].

   c. (5 points) Redo 1(a) on array C = [9 8 7 6 5 4 3 2 1].

   d. (5 points) The partition algorithm may result in extremely unbalanced partition when all elements are equal. Illustrate this by redoing 1(a) on array D = [5 5 5 5 5 5 5 5 5]. Can you find an easy fix so that the partition will be more balanced for this type of input? You can achieve this by modifying two lines in the pseudocode (without adding any lines).

   e. (5 points) In the pseudocode for Quicksort on Slide #7, Quicksort was recursively applied to two subarrays: one subarray from index p to $q-1$, and the other from index $q+1$ to r. This is correct because element $A[q]$ is already in the correct position. Now consider the following modified pseudocode for Quicksort:

```
QUICKSORT(A, p, r)
   if (p < r)
   then q <- PARTITION(A, p, r)
   QUICKSORT(A, p, q-1)
   QUICKSORT(A, q, r)
```

   Assume that the function Partition is the same as the one on Slide #12. The only difference between the pseudocode above and the original one is the last line. In the above pseudocode, the second subarray starts from q instead of $q+1$. Besides having to sort one extra element, this modified algorithm will sometimes fail to terminate. Give an example array for which the modified algorithm will fail to terminate.

   f. (10 points) Use Slide #42 as a model, illustrate the operation of Quick Sort (the viersion in lecture slides #7 and #12) on array A.

2. (20 points) Assuming $T(0) = \Theta(1)$, use **recursion tree method** to solve $T(n) = 3T(n-4) + 1$.

3. (20 points) Assuming $T(0) = \Theta(1)$, use **recursion tree method** to solve $T(n) = T(n/2) + T(n/3) + n$. Prove that your solution is correct using the **substitution method** (you can just prove the Big-Oh part).

Hint: you will need the fact that $(a+b)^k = \sum_{i=0}^{k} \binom{k}{i} a^{k-i} b^k$, where $\binom{k}{i} = \frac{k!}{i!(k-i)!}$. For example, $(\frac{1}{2} + \frac{1}{3})^3 = \frac{1}{2^3} + 3 \cdot \frac{1}{2^2} \cdot \frac{1}{3} + 3 \cdot \frac{1}{2} \cdot \frac{1}{3^2} + \frac{1}{3^3}$.

4. **Bonus** (5 points) How much time did you spend on this homework? Who did you discuss with and what was the discussion about? What do you think about the difficulty level of the homework (harder than

expected? just all right? easy?) What is the most difficult part? Do you have any comments/suggestions about the lecture, recitation, and homework?

Note: Nobody will get more than 80 points for this homework, which means if you have answered all the other questions correctly, answering these bonus questions will NOT help you earn any extra points.