CS 375 Homework 1 Anchu A. Lee
September 25, 2017

I have done this assignment completely on my own. I have not copied it, nor have I given my solution to anyone else. I understand that if I am involved in plagiarism or cheating I will have to sign an official form that I have cheated and that this form will be stored in my official university record. I also understand that I will receive a grade of 0 for the involved assignment for my first offense and that I will receive a grade of F for the course for any additional offense.

1. Use the Master theorem to solve the following recurrences.

   (a) $T(n) = 3T(n/4) + n$
   $a = 3$, $b = 4$, $f(n) = n$
   Case 3: $f(n) = \Theta(n^c)$ if $c = 1$.
   $log_4 3 = 0.79248 < c$
   $T(n) = \Theta(f(n)) = \Theta(n)$

   (b) $T(n) = 2T(n/4) + \sqrt{n}\log(n)$
   $a = 2$, $b = 4$, $f(n) = \sqrt{n}\log(n)$
   Case 2: $f(n) = \Theta(n^c \log^k n)$ if $c = \frac{1}{2}$ and $k = 0$
   $\log_4 2 = 0.5$ so $c = \log_b a$
   $T(n) = \Theta(n^{0.5}\log^1 n = \Theta(\sqrt{n}\log(n))$

   (c) $T(n) = 5T(n/2) + n^2$
   $a = 5$, $b = 2$, $f(n) = n^2$
   Case 1: $f(n) = \Theta(n^c)$ if $c = 2$
   $\log_2 5 = 2.3219... > c$
   $T(n) = \Theta(n^{log_2 5}) = \Theta(n^{2.3218...})$

2. Solve the recurrence

$$T(n) = \begin{cases} \Theta(1) & \text{for } n \leq 1 \\ T(n/4) + T(3n/4) + n & \text{otherwise} \end{cases}$$

   using the recursion tree method. Draw the recursion tree and show the aggregate instruction counts for the following levels (0th, 1st, and last levels), and derive the growth class for T(n) with justifications.

3. Use the substitution method to prove that $T(n) = T(n-1) + n \in O(n^2)$

4. Assume that you are given an array of $n(n \geq 1)$ elements sorted in non-descending order. Design a *ternary* search function that searches the array for a given element x by applying the divide and conquer strategy.

5. Develop a divide-and-conquer approach to selection (and hence a solution for the finding median problem). Hint: for any number v, imagine splitting list S into three categories: elements smaller than v, those equal to v (there might be duplicates), and those greater than v.