

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) $T(n) = 3T(n/4) + n$

(b) $T(n) = 2T(n/4) + \sqrt{n} \lg(n)$

(c) $T(n) = 5T(n/2) + n^2$

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 .