University of Washington February 21, 2019
Department of Computer Science and Engineering
CSE 421, Winter 2019

Homework 6, Due Wednesday, February 27, 2019

On all problems provide justification of your answers. For the algorithms problems, provide an clear explanation of why your algorithm solves the problem, as well as a justification of the run time. Since this assignment is from the Divide and Conquer section, express your algorithms in a recursive manner.

Problem 1 (10 points):

Solve the following recurrences:

a)

$$T(n) = \begin{cases} T(\frac{n}{2}) * T(\frac{n}{2}) & \\ 2 & \text{if } n \leq 1 \end{cases}$$

b)

$$T(n) = \begin{cases} T(n-1) * T(n-1) & \\ 2 & \text{if } n \leq 1 \end{cases}$$

Problem 2 (10 points):

Given an array of elements $A[1, ..., n]$, give an $O(n \log n)$ time algorithm to find a majority element, namely an element that is stored in more than $n/2$ locations, if one exists. Note that the elements of the array are not necessarily integers, so you can only check whether two elements are equal or not, and not whether one is larger than the other. HINT: Observe that if there is a majority element in the whole array, then it must also be a majority element in either the first half of the array or the second half of the array.

Problem 3 (10 points):

Suppose A is an array of n integers that is a strictly decreasing sequence, followed by a strictly increase sequence such as $[12, 9, 8, 6, 3, 4, 7, 9, 11]$. Give an $O(\log n)$ algorithm to find the minimum element of the array.

Problem 4 (10 points):

Let A and B be two sorted arrays of integers, each of length n. Show how you can find the median of the combined set of elements in $O(\log n)$ comparisons. (As in the Median algorithm discussed in lecture, you will need to solve the Select the k-th largest problem.)