CS 141, Fall 2018 | Assignment 2

Posted: October 12th, 2018 | Due: October 26th, 2018, 11:59pm

Problem 1. Solve the following recurrences:

1. $T(n) = 4T(n/2) + \frac{n^2}{\log n}$
2. $T(n) = T(n/3) + T(2n/3) + n$
3. $T(n) = 4T(n/3) + n^{\log_3 4}$

Problem 2. Given an array of numbers $X = \{x_1, x_2, \ldots, x_n\}$, an *exchanged pair* in X is a pair (x_i, x_j) such that $i < j$ and $x_i > x_j$. Note that an element x_i can be part of up to $n-1$ exchanged pairs, and that the maximal possible number of exchanged pairs in X is $n(n-1)/2$, which is achieved if the array is sorted in descending order. Give a divide-and-conquer algorithm that counts the number of exchanged pairs in X in $O(n \log n)$ time.

Problem 3.

For an n that is a power of 2, the $n \times n$ Weirdo matrix W_n is defined as follows. For $n = 1$, $W_1 = [1]$. For $n > 1$, W_n is defined inductively by

$$W_n = \begin{bmatrix} W_{n/2} & -W_{n/2} \\ I_{n/2} & W_{n/2} \end{bmatrix},$$

where I_k denotes the $k \times k$ identity matrix. For example,

$$W_2 = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \quad W_4 = \begin{bmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & 0 & 1 & -1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \quad W_8 = \begin{bmatrix} 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 0 & 1 & -1 & -1 & 0 & -1 & 1 \\ 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & -1 & -1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Give $O(n \log n)$-time algorithm that computes the product $W_n \cdot \bar{x}$, where $\bar{x}$ is a vector of length n and n is a power of 2.

Problem 4. You are given two sorted arrays of integers A and B of size m and n, respectively. Describe a divide and conquer that takes $O(\log k)$ time for computing the k-th smallest element in the union of the two arrays. Assume that integers in the both arrays are distinct, and no integers in A appear in B (and vice versa). Explain carefully why your algorithm takes $O(\log k)$ time. Hint: The k-th smallest element in the union of the two arrays has to be contained in A[1...k] and B[1...k].