CECS 528, Learning Outcome Assessment 5, Yellow, Fall 2023, Dr. Ebert

NO NOTES, BOOKS, ELECTRONIC DEVICES, OR INTERPERSONAL COMMUNICATION ALLOWED. Submit each solution on a separate sheet of paper.

Problem

LO1. Solve the following problems.

(a) Show that $4^{1536} + 9^{4824}$ is divisible by 7.

(b) For the Strassen-Solovay primality test, is $a = 3$ an accomplice or witness to the fact that $n = 5$ is not prime? Show all work.

LO2. Solve the following problems.

(a) Use the Master Theorem to determine the growth of $T(n)$ if it satisfies the recurrence $T(n) = 12T(n/3) + n^{\log_{12} 3}$. Defend your answer.

(b) Use the substitution method to prove that, if $T(n)$ satisfies

$$T(n) = 2T(n/2) + 6n \log n$$

then $T(n) = \Omega(n \log^2 n)$.

LO3. Solve each of the following problems.

(a) Recall that, for the `Randomized Quicksort Algorithm`, $T(n)$ denotes the expected running time of the algorithm when applied to an array a of distinct integers where $\text{size}(a) = n$. Provide an expression for $T(n)$ conditioned on the event that the pivot M selected for the root level of recursion is such that there are 25 members of a that are less than M. Explain.

(b) Demonstrate the partitioning step of Hoare's version of `Quicksort` for the array

$$a = 7, 10, 1, 2, 8, 9, 11, 4, 3, 5, 6$$

where we assume that the pivot equals the median of the first, last, and middle integers of a.

LO4. Solve each of the following problems.

(a) When using the FFT algorithm to compute $\text{DFT}_8(7, -10, 1, 2, -8, 9, -11, 4)$, provide a list of all the subproblem instances that must be computed. Hint: there are 15 of them (including the original problem instance) and $\text{DFT}_4(7, 1, -8, -11)$ is one of them.

(b) Compute $\mathrm{DFT}_4^{-1}(2, 1, -3, 4)$ using the IFFT method. Show the solution to each of the seven subproblem instances and, for each one, clearly represent it using DFT^{-1} notation and apply the formula for computing it. Show all work.

LO5. The following pertains to a correctness-proof outline for the Unit Task Scheduling (UTS) algorithm. Let $S = (a_1, t_1), \ldots, (a_m, t_m)$ represent the tasks that were selected by the algorithm for scheduling, where a_i is the task, and t_i is the time that it is scheduled to be completed, $i = 1, \ldots, m$. Moreover, assume that these tasks are ordered in the same order for which they appear in the sorted order. Let S_{opt} be an optimal schedule which also consists of task-time pairs. Let k be the first integer for which $(a_1, t_1), \ldots, (a_{k-1}, t_{k-1})$ are in S_{opt}, but $(a_k, t_k) \notin S_{\mathrm{opt}}$ because a_k is scheduled by S_{opt}, but at time $t \neq t_k$.

(a) Explain why $t < t_k$. Assume that $t > t_k$ and explain why this creates a contradiction.

(b) Assume that S_{opt} has scheduled some task a at time t_k. Explain why

$$\hat{S}_{\mathrm{opt}} = S_{\mathrm{opt}} - \{(a_k, t), (a, t_k)\} + \{(a_k, t_k), (a, t)\}$$

is a valid schedule. In words, the new schedule swaps schedule times for a_k and a. Explain why this does not create a scheduling problem for either task.

(c) Continuing in this manner we eventually arrive at an optimal schedule S_{opt} for which $S \subseteq S_{\mathrm{opt}}$. Moreover, explain why it is not possible for S_{opt} to possess a task-time pair (a, t) that is *not* a member of S. Assuming it did have such a pair, what contradiction arises?