Data Structures and Algorithms (EE 220): Homework 2

Submit to Ms. Spanner before 10am on Feb 13

Email your programs to PA before the class on 13th Feb

Posted 02/04/2003

Problem 1: (5 pts) Rank the following 5 functions by order of growth; that is, find an arrangement $f_1(n)$, $f_2(n)$, $f_3(n)$, $f_4(n)$ and $f_5(n)$ satisfying $f_1(n) = O(f_2(n))$, $f_2(n) = O(f_3(n)), \ldots, f_4(n) = O(f_5(n))$. The functions are as follows:

1. $\log(n!)$

2. $\log^2(n)$

3. $\log(\log n)$

4. $(\log(n))^{\log(n)}$

5. $e^{\log(n)}$

Problem 2: (5 pts) $f(n)$ and $g(n)$ are asymptotically positive functions.

1. Prove that the following relation is *not* true.

$$f(n) + g(n) = \Theta(\min\{f(n), g(n)\}).$$

2. Prove or disprove the following relations.

(a) $f(n) = O(g(n))$ implies $\log(f(n)) = O(\log(g(n)))$, where $\log(g(n)) > 0$ and $f(n) \geq 1$ for all sufficiently large n.

(b) $f(n) = \Theta(f(\frac{n}{2}))$.

Problem 3: (5 pts) Use iteration to solve the following recurrence.

$$T(n) = T(n - a) + T(a) + n,$$

where $a \geq 1$ is a constant.

Note: Please do not use Master Theorem.

Problem 4: (10 pts) This Problem is a variation of subsequence sum problem discussed in the class. Consider two lists (stored in the arrays) $LIST_1$ and $LIST_2$ sorted in the ascending order, each of length n and m, respectively. Now, append $LIST_1$ by $LIST_2$, which will give us a list of length $n + m$. Aim is to find the maximum subsequence sum of the newly obtained list for the following case. Total number of positive numbers in the lists $LIST_1$ and $LIST_2$ are $O(\sqrt{n})$ and $O(\sqrt{m})$, respectively and m is much smaller than n.

   Prove the correctness of your solution and analyze the complexity. Grades will depend on the correctness and the efficiency of the solution.

Programming Assignment (15 pts) The objective of this assignment is to test the running time of your algorithm for the large input size. Fix $m = 100$ and vary n from 1000 to 100,000 in the step of 1000. Each time for $LIST_1$ make the last $\sqrt{n}$ elements equal to 1 and remaining elements equal to -1. Similarly, for $LIST_2$ make last $\sqrt{m}$ elements 10 and the remaining -1. Plot the runtime of your algorithm as a function of parameter n. Does your plot resemble the function you would expect? (If you wish, also plot the function that you would expect, so that the comparison becomes easy.)