Problem Set 1 for CS 170

Problem 1 [Asymptotics]

Order the functions $\sqrt{n}, \log n, 100n + \log n, \log n^{\log n}, \frac{n^2}{\log n}, n2^n, 3^n, n(\log n)^2, \frac{n}{\log n}, (\log n)^5$ into a list from left to right, so that if $f(n)$ is any function in the list and $g(n)$ is the function to its immediate right, we have $f(n) = O(g(n))$. You do not need to prove your answer.

Problem 2 [More Asymptotics]

Let $f(n)$ and $g(n)$ be asymptotically nonnegative functions. Prove or disprove the following:

(a) $\max(f(n), g(n)) = \Theta(f(n) + g(n))$

(b) $f(n) = O(g(n))$ implies $2^{f(n)} = O(2^{g(n)})$

Problem 3 [Recurrence Equations]

Give asymptotically tight solutions for $T(n)$. Assume throughout that $T(n) = \Theta(1)$ for n sufficiently small.

(a) $T(n) = 7T(n/2) + n^2$

(b) $T(n) = 3T(n-6) + n$

(c) $T(n) = T(\sqrt{n}) + 1$

(d) $T(n) = 2T(n/2) + n \lg^2 n$

(e) $T(n) = 2T(n/2) + n/\lg n$

Problem 4 [A Divide and Conquer Algorithm]

Give a divide and conquer algorithm for the following problem: you are given two sorted lists of size m and n, and are allowed unit time access to the ith element of each list. Give an $O(\log m + \log n)$ time algorithm for computing the kth largest element in the union of the two lists. (For simplicity, you can assume that the elements of the two lists are distinct).

Problem 5 [More Divide and Conquer]

Which is faster: a divide and conquer algorithm that breaks a problem of size n into two problems of size $n/2$ at the cost of n^2 steps or a divide and conquer algorithm that breaks a problem of size n into three problems of size $n/2$ at the cost of n steps. Justify your answer.

Problem 6 [Order Statistics]

Show that the smallest and the second smallest of n distinct elements can be found with $n + \lceil \lg n \rceil - 2$ comparisons in the worst case.