CS 3100 – Algorithms

Assignment 1
Kishore Kothapalli

Due: 6-DEC-2007 in class.

Problem 1. Prove or disprove the following statements.

- $f(n) = O(g(n))$ implies $2^{f(n)} = O(2^{g(n)}$.
- $f(n) + g(n) = \Theta(\min\{f(n), g(n)\}$.
- $f(n) \neq O(g(n))$ implies $g(n) = O(f(n))$.
- $f(n) + g(n) = O(\min\{f(n) + g(n)\})$.

(4 Points)

Problem 2. Solve the following recurrence relations.

- $T(n) = 3T(n/2) + n \log n$
- $T(n) = T(\sqrt{n}) + 1$
- $T(n) = 8T(n/3) + n^2$
- $T(n) = T(n) = T(n-2) + n$
- $T(n) = T(n-2) + 2 \log n$
- $T(n) = \sqrt{n}T(\sqrt{n}) + n$
- $T(n) = 5T(n/3) + n^{4/3}$
- $T(n) = T(8n/9) + n$
- $T(n) = T(n-1) + \log n$
- $T(n) = 4T(n/2) + n^2 \log n$

(5 Points)

Problem 3. Given an array A design an efficient algorithm to find two elements x, y of A so that $|x - y|$ is minimized. **(5 Points)**

Problem 4. In an array A define a pair (i, j) to be an inversion if $i < j$ and $A[i] > A[j]$. Answer the following questions.

- Given $A = (3, 5, 2, 9, 6, 1)$ find the array B consisting of elements of A but with the maximum number of inversions.
- What is the relationship between the number of inversions in an array A and the runtime of insertion sort on A. Justify your answer.

(2+3=5 Points)

Problem 5. Consider using insertion sort to sort an array A. Suppose we use binary search to find the position of the element that we are trying to insert into the already sorted sequence. How does this affect the worst-case runtime of insertion sort? Justify. **(5 Points)**