Analysis of Algorithms

Midterm Exam : 26 April 2012

1. Prove that $(2n+1)^3 = \Theta(n^3)$ by definition of the Θ notation.

2. Solve the following recurrence by any method

$$T(n) = 3T(n/2) + n.$$

(Assume that n is a power of 2.)

3. Describe the divide-and-conquer algorithm for the maximum subarray problem.

4. Design a divide-and-conquer algorithm to find the minimum and maximum elements in a set of n numbers. Prove that your algorithm uses at most $3n/2$ comparisons. (You may assume that n is a power of 2 in the proof.)

5. Show how quicksort can be made to run in $O(n \log n)$ time in the worst case by using the idea of the linear-time selection algorithm.

6. Show how to maintain a dynamic set Q of numbers that supports the operation Min-Gap, which gives the magnitude of the difference of the two closest numbers in Q. For example, if $Q = \{1, 5, 9, 15, 18, 22\}$, then Min-Gap($Q$) returns $18 - 15 = 3$, since 15 and 18 are the two closest numbers in Q. Make the operations Insert, Delete, Search, and Min-Gap as efficient as possible, and analyze their running times.