Midterm

Note

This is a closed-book exam. Each problem accounts for 10 points, unless otherwise marked.

Problems

1. Given a set of $n+1$ numbers out of the first $2n$ (starting from 1) natural numbers 1, 2, 3, ..., $2n$, prove *by induction* that there are two numbers in the set, one of which divides the other.

2. Below is a theorem from Manber's book:

   For all constants $c > 0$ and $a > 1$, and for all monotonically increasing functions $f(n)$, we have $(f(n))^c = O(a^{f(n)})$.

   Prove, by using the above theorem, that $n(\log n)^2 = O(n^{1.5})$.

3. Find the asymptotic behavior of the function $T(n)$ defined by the recurrence relation

   $T(n) = T(n/2) + \sqrt{n}$, $T(1) = 1$.

   You may consider only values of n that are powers of 2. (5 points)

4. In the towers of Hanoi puzzle, there are three pegs A, B, and C, with n (generalizing the original eight) disks of different sizes stacked in decreasing order on peg A. The objective is to transfer all the disks on peg A to peg B, moving one disk at a time (from one peg to one of the other two) and never having a larger disk stacked upon a smaller one.
   (a) Give an algorithm to solve the puzzle. (5 points)
   (b) Suppose that there is an additional fourth peg D. Modify your algorithm to take advantage of the additional peg. (10 points)

5. Show all intermediate and the final AVL trees formed by inserting the numbers 9, 8, 7, 6, 5, 0, 1, 2, 3, and 4 (in this order).

6. Suppose that you are given an algorithm as a *black box* (you cannot see how it is designed) that has the following properties: If you input any sequence of real numbers and an integer k, the algorithm will answer "yes" or "no," indicating whether there is a subset of the numbers whose sum is exactly k. Show how to use this black box to find the subset whose sum is k, if it exists. You should use the black box $O(n)$ times (where n is the size of the sequence).

7. Apply the quicksort algorithm to the following array. Show the result after each partition operation.

8	1	5	11	14	12	2	16	7	3	13	4	10	9	15	6

8. (a) Give an algorithm for building a (max) heap using the bottom-up approach.

   (b) Rearrange the following array into a heap using the algorithm obtained in (a).

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
6	2	5	11	9	14	3	10	8	1	13	4	12	15	7

   Show the result after each element is added to the part of array that already satisfies the heap property.

9. Draw a Huffman tree for a text that contains eight characters A, B, C, D, E, F, G, and H with frequencies 7, 2, 3, 5, 16, 10, 4, and 2, respectively.

10. Compute the *next* table as in the KMP algorithm for the string *aababaabaa*. Show the details of your calculation.