Data Structures - Assignment no. 7, May 10, 2006

Remarks:

- **Please write your exercises in pen, or in clearly visible pencil. Please write very clearly.**
- **For every question where you are required to write pseudo-code, also explain your solution in words.**

1. Find the order of growth for the following recursively given functions, using the Master Method.
   (a) $T(n) = 4T(n/2) + n$
   (b) $T(n) = 4T(n/2) + n^2$
   (c) $T(n) = 4T(n/2) + n^3$
   (d) Explain why $T(n) = 2T(n/2) + nlog^2n$ cannot be solved using the Mater Method.

2. Find the order of growth for the following recursively given functions. Explain your answer.
   (a) $T(n) = T(n - a) + T(a) + n$ where $a \geq 1$ is constant.
   (b) $T(n) = T(cn) + T((1 - c)n) + n$ where $0 < c < 1$ is a constant.
   (c) **(Corrected 11/5/06)** $T(n) = T(2cn)+T((1-c)n)+n$ where $0 < c < 1/2$ is a constant.
   (d) $T(n) = 2T(n/2) + nlog^2n$ (Hint: The approach to solving this is similar to the approach to solving $T(n) = 2T(n/2) + n$).

3. You are given an array of length n. You are told that there is a value that appears in it exactly $n/2$ times, another value that appears exactly $n/4$ times, another that appears exactly $n/8$ times, and so on. (note that you are not told what these values are). Describe an algorithm for sorting this array that runs in time $O(n)$.

4. **(Modified 15/5/06)** Show how to modify the Hash Table data structure, such that all operations (find, insert and delete) take $O(\log n)$ time in the worst-case. All operations should still take $O(1)$ time in expectation. This question refers to the hash table data structure that you learned in class, using universal hashing and chaining. This does not refer to perfect hashing.

5. **(Modified 10/5/06)** Let H be a Hash Table with $3n$ cells. (where each cell has room only for a single element, that is, without chaining). In case you insert into H an element that causes a collision, the element is inserted into an auxiliary linked list, L. Suppose we insert $2n$ different elements into this Hash Table. What is the expected length of L after this sequence of insertions?