Data Structures - Assignment no. 8

Remarks:

- **Write both your name and your ID number very clearly on the top of the exercise. Write your exercises in pen, or in clearly visible pencil. Please write *very* clearly.**
- **Give correctness and complexity proofs for every algorithm you write.**
- **For every question where you are required to write pseudo-code, also explain your solution in words.**

1. Describe and analyze an algorithm that checks in time $O(n \log\log n)$ whether a given array of n numbers contains the same number at least $n/\log n$ times.

2. Let S be a set of $n+1$ real numbers between m and M.

   (a) Prove that there are two numbers $a, b \in S$ such that $|a-b| \leq (M-m)/n$.

   (b) Write an algorithm that gets an array of $n+1$ numbers between m and M, and returns two numbers a, b in this array, such that $|a-b| \leq (M-m)/n$. The running time of the algorithm should be $O(n)$.

3. (a) Suppose you are given an array of size n in which no element appears more than once, and two integers k_1 and k_2 such that $1 \leq k_1 \leq k_2 \leq n$. Describe an $O(n)$ time algorithm that returns all of the elements whose ranks in the array are $k_1, k_1+1, k_1+2, \ldots, k_2$. Note that you do not have to return these elements in a sorted order. (Note: The *rank* of an element in an array is the number of elements smaller or equal to it).

   (b) Denote $k = k_2 - k_1$. What running time could you get if we did require that the elements be returned in sorted order? Write the best running time you can get as a function of both n and k.

4. Suppose you are given 3 sorted arrays of size n. You may assume that no element appears in the input more than once. Describe an algorithm that computes the median of the union of these arrays in time $O(\log n)$. Prove the correctness of your algorithm and prove the running-time bound.

5. A family $\mathcal{H}$ of functions from U to $\{0, 1, \ldots, m-1\}$ is called *100-weakly universal* if for all $x, y \in U$ such that $x \neq y$ it holds that

$$Pr_{h \leftarrow \mathcal{H}}[h(x) = h(y)] \leq \frac{100}{m} .$$

   Let $U = \{0, 1, \ldots, u-1\}$. Define f_a to be the function $f_a(x) = a \cdot x \mod m$. Show that $\mathcal{H} = \{f_a | 0 < a < m\}$ is not a 100-weakly universal family.

6. 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.