Algorithm Design and Analysis (Fall 2023)
Assignment 1
Deadline: Nov 1, 2023

1. (25 points) Prove the following generalization of the master theorem. Given constants $a \geq 1, b > 1, d \geq 0$, and $w \geq 0$, if $T(n) = 1$ for $n < b$ and $T(n) = aT(n/b) + n^d \log^w n$, we have
$$T(n) = \begin{cases} O(n^d \log^w n) & \text{if } a < b^d \\ O(n^{\log_b a}) & \text{if } a > b^d \\ O(n^d \log^{w+1} n) & \text{if } a = b^d \end{cases}.$$

2. (25 points) Recall the median-of-the-medians algorithm we learned in the lecture. It groups the numbers by 5. What happens if we group them by $3, 7, 9, \ldots$? Please analyze those different choices and discuss which one is the best.

3. (25 points) Let X and Y be two sets of integers. Write $X \succ Y$ if $x \geq y$ for all $x \in X$ and $y \in Y$. Given a set of m integers, design an $O(m \log(m/n))$ time algorithm that partition these m integers to k groups $X_1, \ldots, X_k$ such that $X_i \succ X_j$ for any $i > j$ and $|X_1|, \ldots, |X_k| \leq n$. Notice that k is not specified as an input; you can decide the number of the groups in the partition, as long as the partition satisfies the given conditions. You need to show that your algorithm runs in $O(m \log(m/n))$ time.

   Remark: We have not formally define the asymptotic notation for multi-variable functions in the class. For f and g be functions that maps $\mathbb{R}^k_{>0}$ to $\mathbb{R}_{>0}$, we say $f(\mathbf{x}) = O(g(\mathbf{x}))$ if there exist constants $M, C > 0$ such that $f(\mathbf{x}) \leq C \cdot g(\mathbf{x})$ for all $\mathbf{x}$ with $x_i \geq M$ for some i. The most rigorously running time should be written as $O(m \cdot \max\{\log(m/n), 1\})$, although it is commonly just written as $O(m \log(m/n))$ for this kind of scenarios.

4. (25 points) Given an array $A[1, \ldots, n]$ of integers sorted in ascending order, design an algorithm to **decide** if there exists an index i such that $A[i] = i$ for each of the following scenarios. Your algorithm only needs to decide the existence of i; you do not need to find it if it exists.

   (a) The n integers are positive and distinct.

   (b) The n integers are distinct.

   (c) The n integers are positive.

   (d) The n integers are positive and are less than or equal to n.

   (e) No further information is known for the n integers.

   Prove the correctness of your algorithms. For each part, try to design the algorithm with running time as low as possible.

5. How long does it take you to finish the assignment (including thinking and discussion)? Give a score (1,2,3,4,5) to the difficulty. Do you have any collaborators? Please write down their names here.