

Rules: Discussion of the problems is permitted, but writing the assignment together is not (i.e. you are not allowed to see the actual pages of another student). You can get at most 100 points if attempting all problems. Please make your answers precise and concise.

1. (10 pts) Give an algorithm that takes a real number a and a positive integer n as input, and outputs the value of a^n in $O(\log n)$ time.

Present your algorithm in pseudo-code.

Assume that any of the $+, -, *, /$ operations on any two real numbers takes $O(1)$ time.

2. (15 pts) Suppose we have $T(n) \leq c = O(1)$ for all $n \leq 3$, and for every $n \geq 4$, we have

$$T(n) \leq T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lfloor \frac{n}{4} \right\rfloor\right) + c \cdot n.$$

Use Mathematical Induction to prove that $T(n) = O(n)$ for all $n \geq 4$.

3. (10 pts) Given two sequences of elements A and B , where $|A| = |B| = n$ and for each sequence, the elements have different values. Design an $O(n \log n)$ algorithm to count the number of values that appear in both A and B . In other words, you need to output the size of $A \cap B$ in $O(n \log n)$ time.

Please state the steps of your algorithm clearly in pseudo-code. You do not need to prove the correctness or analyze the running time.

4. (15 pts) Given a sequence of numbers A , design an algorithm that counts the number of inversions, where an inversion is a pair (a_i, a_j) such that $i < j$ and $a_i > a_j$. Please state the steps of your algorithm clearly and analyze its running time.
- (a) (8 pts) Given two sorted arrays L and R , design a linear $(O(|L| + |R|))$ time algorithm that counts the number of pairs (l, r) such that $l \in L$, $r \in R$ and $l > r$.
- (b) (7 pts) Suppose we have a linear time algorithm for question (a), design an $O(n \log n)$ time algorithm that computes the number of inversion in A .

5. (15 pts) A hotel manager has to process n advance bookings of rooms. Each booking i consists of an arrival date a_i and a departure date d_i . His hotel has k identical rooms and he needs to find out whether there are enough rooms in the hotel to satisfy the demand. Give an algorithm in pseudocode that solves this problem in time $O(n \log n)$, i.e., the algorithm should output “YES” if the hotel has enough rooms to satisfy the n advance bookings and “NO” otherwise.

You need to provide an analysis for the complexity of your algorithm.

6. (15 pts) Given an array $A = \{a_1, a_2, \dots, a_n\}$ of n integers in the range $[0, n^2 - 1]$, design an algorithm for sorting A in $O(n)$ time. Please state the steps of your algorithm clearly and analyze its running time.

7. (40 pts) **Comparison of Sorting Algorithms.**

In this problem you need to implement the different sorting algorithms, and compare their running times on different inputs. Implement each of the following algorithms as a function that takes as input an array (which can be very long), and outputs the sorted version of the array (from minimum to maximum).

- InsertionSort : based on Insertion-Sort from the lecture notes;
- BubbleSort : based on Bubble-Sort from the lecture notes;
- SelectionSort : based on Selection-Sort from the lecture notes;
- HeapSort : use heap implementation of priority queue for sorting;
- MergeSort : based on Merge-Sort from the lecture notes;
- QuickSort : use median of three random elements as the pivot.

In the main function, we read an array $A = \{a_1, a_2, \dots, a_n\}$ of different integers from a file, and use different sorting algorithm to do sorting. For each algorithm, test whether the returned array is sorted or not, and output its running time.

Several test cases of array A will be provided.