

Assignment 4: Sorting

Total Marks: 70

In this assignment, we will explore the following concepts studied in the lecture:

1. Sorting
2. Data Structures

It involves writing an efficient and well-commented C code for the first Problem Statement and for the second Problem Statement submit a pdf with the answers clearly. Make a zip folder named RollNo_Assignment4.zip . **Python program will not be accepted.**

1) Problem Statement:

Marks: 30

A man wants to buy a house in a locality where there are 2 different sectors. The number of houses in sector 1 and sector 2 is **m** and **n** respectively. The prices(in lakhs) of houses in sector 1 and sector 2 are stored in two arrays **sec1** and **sec2** respectively. The man wants to know the median price of the property in a particular locality.

Given the 2 integer arrays **sec1** and **sec2** and the integers **m** and **n**. Print *median of the locality*. You are **not allowed** to use a built-in sorting function.

Write the pseudocode for the proposed algorithm in a comment in the file along with time complexity. Note that programs with lower time complexity will be given higher marks.

Constraints:

- `sec1.length == m`
- `sec2.length == n`
- `0 <= m <= 1000`
- `0 <= n <= 1000`
- `1 <= m + n <= 2000`
- `0 <= sec1[i], sec2[i]`

Example 1:

Input: `m = 3, n = 4, sec1 = [14], sec2 = [30,10]`

Output: 14.0 L

Explanation: merged array = [10,14,30] and median is 14.0

Example 2:

Input: m = 2, n = 2, sec1 = [4, 1], sec2 = [2,3]

Output: 2.5 L

Explanation: merged array = [1,2,3,4] and median is $(2 + 3) / 2 = 2.5$.

Example 3:

Input: m = 0, n = 1, sec1 = [], sec2 = [2]

Output: 2.0 L

2) Problem Statement:

Marks: 40

- a) Show how to multiply two non-zero complex numbers $a + b i$ and $c + d i$ using only three multiplications of real numbers. The result should be equal to $(ac - bd) + (ad + bc) i$.
- b) Solve the recurrence $T(n) = 3 T(\sqrt{n}) + \log n$. Your solution should be asymptotically tight.
- c) In a permutation a_1, \dots, a_n , of n distinct integers, and inversion is a pair (a_i, a_j) such that $i < j$ and $a_i > a_j$. What would be worst case time complexity of the Insertion Sort algorithm, if the inputs are restricted from $1, \dots, n$ with at most n inversions?
- d) Consider the following recurrence $T(n) = 3T(n/5) + \lg n * \lg n$. What is the value of $T(n)$?