LABSHEET-3

CS F211 Data Structures and Algorithms

Academic Year: 2025-2026(2nd Semester)

General Instructions for this Labsheet:

- Question 1, 7 and 9 can be done using any suitable data structure.
- Question 2, 3, 4, 5, 6, 8 and 10 should be done using binary search. Using linear search is not allowed.

Question 1: Student Ranking with Tie Handling

Problem Description:

An online examination system stores student performance data using two values for each student: marks obtained and submission delay in minutes. You are required to generate the final ranking of students based on the following conditions.

Students with higher marks should be ranked before students with lower marks. If two students have the same marks, the student with the smaller submission delay should be ranked higher. If both the marks and the submission delay are the same for two or more students, you may print them in any order.

Input Format:

Line 1: N (Number of students)
Line 2 to Line N+1: Two integers per line each having marks and delay of a student

Output Format:

Print the final ranking order, one student per line, displaying marks and delay.

Sample Test Case 1:
Input:
5
90 5
85 2
90 3
85 2
70 0

Output:
90 3
90 5
85 2
85 2
70 0

Sample Test Case 2:
Input:
4
80 4
80 4
95 1
80 3

Output:
95 1
80 3
80 4
80 4

Question 2: Peak Element in an Array

Problem Description:

A peak element is defined as an element that is strictly larger than its immediate neighbors. For an array of integers, you should imagine that the values located immediately outside the boundaries of the array are negative infinity, meaning that the first element is a peak if it is larger than its successor and the last element is a peak if it is larger than its predecessor.

Given a zero-indexed array where it is guaranteed that no two consecutive elements are identical, your task is to identify and return the index of any single peak element. If the array contains multiple peaks, returning the index of any one of them is acceptable. You must solve this problem using a binary search approach.

Input Format:
Line 1: N (Size of the array, N ≥ 3)
Line 2: N space-separated integers

Output Format:
Print the index of a peak element

Sample test case 1:
Input:
6
-3 -1 0 -2 -5 -6

Output:
2

Sample test case 2:
Input:
5
-10 -5 -3 -4 -9

Output:
2

Question 3: Minimum Eating Speed

Problem Description:

A student has N piles of snacks, where the number of snacks in each pile is given in an array. The student eats snacks from only one pile at a time and eats at a constant speed of X snacks per hour. If a pile has fewer than X snacks, the student finishes that pile within the hour and does not eat from any other pile during the same hour. The student has a total of H hours to finish eating all the snacks. Your task is to determine the minimum integer value of X such that the student can finish all the snack piles within H hours.

You must solve this problem using binary search.

Input Format:
Line 1: Two integers N and H, where N is the number of piles and H is the total number of hours available($N \leq H$).
Line 2: N space-separated integers representing the number of snacks in each pile.

Output Format:
Print a single integer representing the minimum eating speed.

Sample Test Case 1:
Input:
4 8
3 6 7 11

Output:
4

Sample Test Case 2:
Input:
3 6
10 10 10

Output:
5

Question 4: Search in Rotated Sorted Array

Problem Description:

You are given an array of integers that was originally sorted in increasing order and then left rotated at an unknown pivot. The array may contain negative numbers, but all elements are distinct. Given a target value, your task is to search for this value in the rotated array using a binary search approach and return the index of this element.

Input Format:
Line 1: N (Size of the list)
Line 2: N space-separated integers
Line 3: Target value

Output Format:
Index of the target element if found, otherwise -1

Sample test case 1:
Input:
7
4 5 6 -3 -2 0 2
-2

Output:
4

Sample test case 2:
Input:
5
1 2 3 4 5
6

Output:
-1

Question 5: Find First and Last Occurrence

Problem Description:

An e-commerce platform logs customer ratings as a sorted array of integers (non-decreasing order). To analyze the popularity of a specific rating score, you need to find the **starting** and **ending** index of a given target number T.

If the target is not found in the array, print −1 −1.

You must solve this problem using a binary search approach, or else 0 marks will be awarded.

Input Format:

Line 1: N (Size of the array)

Line 2: N space-separated integers (sorted)

Line 3: T (Target value)

Output Format:

Print the starting and ending indices separated by a space.

Sample Test Case 1:

Input:

6

1 2 4 4 4 5

4

Output:

2 4

Sample Test Case 2:

Input:

5

1 2 3 6 9

5

Output:

-1 -1

Question 6: Equation Solver

Problem Description:

You are given a mathematical function f(x) = 2(x^3) − (x^2) + 5x. This function is strictly increasing for all positive integers. Given a large positive integer Y, your task is to find the largest integer x such that f(x) ≤ Y.

You must solve this problem using a binary search approach, or else 0 marks will be awarded.

The same function will be provided in the exam.

Input Format:

Line 1: Y (0 < Y ≤ 1,000,000)

Output Format:

Print the largest integer x that satisfies the condition.

Given the constraints, it can be shown that the value of x will always lie in the range [0, 80].

Sample Test Case 1:

Input:

200

Output:

4

Sample Test Case 2:

Input:

50,000

Output:

29

Question 7: Group Sorting

Problem Description:

You are given an array of N integers. Your task is to divide this array into consecutive groups of strictly increasing sizes and sort each group individually in **ascending order**.

The grouping follows this pattern:

1. The **1st group** takes the first **1** element.
2. The **2nd group** takes the next **2** elements.
3. This continues (the k-th group takes the next k elements) until the end of the array is reached.

If the number of remaining elements is less than the required size for the next group, include **all remaining elements** in the final group and sort them.

Input Format:

Line 1: N (Size of the array)

Line 2: N space-separated integers

Output Format:

Print the final array with individually sorted groups, separated by spaces.

Sample Test Case 1:

Input:

10

10 9 8 7 6 5 4 3 2 1

Output:

10 8 9 5 6 7 1 2 3 4

Sample Test Case 2:

Input:

5

5 4 3 6 2

Output:

5 3 4 2 6

===> The underlined elements are in the same group.

Question 8: The Logistics Manager

Problem Description:

A conveyor belt has N packages that must be shipped from warehouse to a port within D days. The ith package has a weight of weights[i]. The conveyor belt must be loaded with packages in the order they appear (you cannot re-order them). Each day, you load the ship with packages until the total weight reaches the ship's capacity .

- **Objective:** Find the **Minimum Capacity** of the ship required to transport all packages within D days.

Input Format:

- Line 1: N (Number of packages) and D (Days).
- Line 2: N integers (Weights).

Output Format:

- Print the minimum capacity required.

  Sample Test Case 1:

  Input:
  5 2
  1 2 3 4 5

  Output:
  9

  (Explanation: Capacity 9. Day 1: [1, 2, 3] sum=6. Day 2: [4, 5] sum=9. Total 2 days.)

  Sample Test Case 2:

  Input:
  3 3
  3 2 2

  Output:

  3

Question 9: Inversion Count

Problem Description:

You are given an array of N integers. A pair of indices is called an inversion if the element at the earlier position in the array is greater than the element at the later position.

Task: Count the total number of inversions in the array.

Use merge sort to solve this question.

Input Format:

- Line 1: N(Size of array).
- Line 2: N integers.

Output Format:

- Print the total number of inversions.

Sample Test Case 1:

Input:
5
2 4 1 3 5

Output:
3

Explanation:

The inversions are:

1. (2, 1)-> 2 > 1
2. (4, 1)-> 4 > 1
3. (4, 3)-> 4 > 3

Sample Test Case 2:

Input:
4
8 4 2 1

Output:
6

Question 10: The Median Merger

Problem Description:

You are given two sorted arrays nums1 and nums2 of size N and M respectively. Find the Median of the two sorted arrays combined.

- The median is the middle value in an ordered integer list. If the size of the list is even, there is no single middle value and the median is the mean of the two middle values.
- Use binary search to implement this question. Do not use a sorting algorithm.

Input Format:

- Line 1: N(Size of first array).
- Line 2: N sorted integers.
- Line 3: M (Size of second array).
- Line 4: M sorted integers.

Output Format:

- Print the median value formatted to **2 decimal places** .

Sample Test Case 1:

Input:
5
1 4 7 10 12
6
2 3 6 8 9 11

Output**:**
7.00

Explanation: The combined sorted array has 11 elements. The median is the 6th element, which is 7.

Sample Test Case 2:

Input:
7
1 2 3 4 5 6 7
3
100 200 300

Output:
5.50
Explanation: The combined size is 10. The middle elements are 5 and 6.