Arrays

Question 1:

Write a Java program that finds and prints the second largest element in an integer array.

Question 2:

Implement a Java function to merge two sorted arrays into a single sorted array.

Question 3:

Create a Java program that finds the "leader" elements in an integer array. An element is considered a leader if it is greater than all the elements to its right.

Question 4:

Write a Java function to find the "nth" smallest element in an unsorted integer array.

Question 5:

Write a Java program that segregates (moves) all zeros in an integer array to the end while maintaining the order of non-zero elements.

Question 6:

Create a Java program that calculates the product of all elements in an array, except the current element. The output array should have the same length as the input array.

Question 7:

Write a Java function that determines whether there are any duplicate elements in an array.

List:

Question 1:

Write a Java program that takes a list of integers as input and returns a new list with all the duplicate elements removed.

Question 2:

Implement a Java function that finds the kth smallest element in an ArrayList of integers.

Question 3:

Create a Java program that takes two ArrayLists of strings as input and returns a new ArrayList containing strings that are present in the first list but not in the second list.

Question 4:

Write a Java function to rotate an ArrayList of integers to the right by a given number of steps.

Question 5:

Implement a Java program that computes the intersection of two ArrayLists of integers. The intersection should contain only distinct elements.

Question 6:

Create a Java function that partitions an ArrayList of integers into two sublists, one containing even numbers and the other containing odd numbers, while maintaining their relative order.

Question 7:

Write a Java program that finds the longest increasing subsequence in an ArrayList of integers. The subsequence does not need to be contiguous.

Question 8:

Implement a Java function to shuffle the elements of an ArrayList of strings randomly.

Question 9:

Create a Java program that takes an ArrayList of strings and returns a new ArrayList with the strings sorted by their lengths in ascending order.

Question 10:

Write a Java function that merges two sorted ArrayLists of integers into a single sorted ArrayList.

HashMaps:

Question 1:

Write a Java program that takes a paragraph of text as input and returns a HashMap containing the frequency of each word in the paragraph.

Question 2:

Implement a Java function that finds the first non-repeated character in a string using a HashMap.

Question 3:

Create a Java program that simulates a simple student grading system using a HashMap. Allow users to enter student names and their corresponding grades, and then display the average grade.

Question 4:

Write a Java function that takes an array of integers as input and returns a HashMap containing the frequency of each number along with the list of indices at which each number appears.

Question 5:

Implement a Java program that finds the intersection of two HashMaps, meaning the keys that are present in both HashMaps.

Question 6:

Create a Java function that takes a string as input and returns a HashMap containing the frequency of each character in the string.

Question 7:

Write a Java program that finds the longest substring with distinct characters in a given string using a HashMap.

Question 8:

Implement a Java function to group an array of strings into a HashMap where the keys are the lengths of the strings and the values are lists of strings of the same length.

Question 9:

Create a Java program that takes a list of transactions (sender, receiver, amount) as input and returns a HashMap containing the total balance for each account.

Question 10:

Write a Java function that takes a list of integers as input and returns a HashMap containing the count of even and odd numbers.