Module Name: JAVA Logic Placement Preparation Test 4
Batch: PG-DAC

Set 1

Q.1) Print unique sorted array ➔Accept data in sorted order having duplicate value. You need to print unique array using single loop . Unique sorted array using 1 loop

Input➔ 1 1 2 2 2 5 output➔1 2 5

Q.2) To find the maximum sum of all subarrays of size K:

Given an array of integers of size 'n', Our aim is to calculate the maximum sum of 'k' consecutive elements in the array.

Input : arr[] = {100, 200, 300, 400}, k = 2

Output : 700

Module Name: JAVA Logic Placement Preparation Test 4
Batch: PG-DAC

Set 2

Q.1) Print unique sorted array ➔Accept data in sorted order having duplicate value. You need to print unique array using single loop . Unique sorted array using 1 loop
Input➔ 1 1 2 2 2 5 output➔1 2 5

Q.2) Given a 1-based indexing array **arr[]** of **non-negative** integers and an integer **sum**. You mainly need to return the left and right indexes(**1-based indexing**) of that subarray. In case of multiple subarrays, return the subarray indexes which come first on moving from left to right. If no such subarray exists return an array consisting of element **-1**.

Examples:

Input: *arr[] = [15, 2, 4, 8, 9, 5, 10, 23], target = 23*

Output: *[2, 5]*

Explanation: *Sum of subarray arr[2...5] is 2 + 4 + 8 + 9 = 23.*

Input: *arr[] = [1, 10, 4, 0, 3, 5], target = 7*

Output: *[3, 5]*

Explanation: *Sum of subarray arr[3...5] is 4 + 0 + 3 = 7.*

Input: *arr[] = [1, 4], target = 0*

Output: *[-1]*

Explanation: *There is no subarray with 0 sum.*