PROBLEMS OF THE DAY – 8

1.Two sum -Pairs with 0 Sum

Given an integer array arr, return all the unique pairs [arr[i], arr[j]] such that i != j and arr[i] + arr[j] == 0.

Note: The pairs must be returned in sorted order, the solution array should also be sorted, and the answer must not contain any duplicate pairs.

Examples:

Input: arr = [-1, 0, 1, 2, -1, -4]

Output: [[-1, 1]]

Explanation: arr[0] + arr[2] = (-1)+ 1 = 0. arr[2] + arr[4] = 1 + (-1) = 0. The distinct triplets are [-1,1].

Input: arr = [6, 1, 8, 0, 4, -9, -1, -10, -6, -5]

Output: [[-6, 6],[-1, 1]]

Explanation: The distinct triplets are [-1, 1] and [-6, 6].

Expected Time Complexity: **O(n log n)**

Expected Auxiliary Space: **O(n)**.

2.Power of 2

Given a non-negative integer n. The task is to check if it is a power of 2.

Examples:

Input: n = 8

Output: true

Explanation: 8 is equal to 2 raised to 3 (23 = 8).

Input: n = 98

Output: false

Explanation: 98 cannot be obtained by any power of 2.

Input: n = 1

Output: true

Explanation: (20 = 1).

Expected Time Complexity: **O(log n).**

Expected Auxiliary Space: **O(1).**

3.Common in 3 Sorted Arrays

You are given three arrays sorted in increasing order. Find the elements that are common in all three arrays.If there are no such elements return an empty array. In this case, the output will be -1.

Examples :

Input: arr1 = [1, 5, 10, 20, 40, 80] , arr2 = [6, 7, 20, 80, 100] , arr3 = [3, 4, 15, 20, 30, 70, 80, 120]

Output: [20, 80]

Explanation: 20 and 80 are the only common elements in arr1, arr2 and arr3.

Input: arr1 = [1, 2, 3, 4, 5] , arr2 = [6, 7] , arr3 = [8,9,10]

Output: [-1]

Explanation: There are no common elements in arr1, arr2 and arr3.

Input: arr1 = [1, 1, 1, 2, 2, 2], B = [1, 1, 2, 2, 2], arr3 = [1, 1, 1, 1, 2, 2, 2, 2]

Output: [1, 2]

Expected Time Complexity: **O(n)**

Expected Auxiliary Space: **O(n)**

4.Count Digits

Given a number n. Count the number of digits in n which evenly divide n. Return an integer, total number of digits of n which divides n evenly.

Examples :

Input: n = 12

Output: 2

Explanation: 1, 2 when both divide 12 leaves remainder 0.

Input: n = 2446

Output: 1

Explanation: Here among 2, 4, 6 only 2 divides 2446 evenly while 4 and 6 do not.

Input: n = 23

Output: 0

Explanation: 2 and 3, none of them divide 23 evenly.

Expected Time Complexity: **O(n)**

Expected Space Complexity: **O(1)**

5.Square root of a number

Given an integer n, find the square root of n. If n is not a perfect square, then return the floor value. Floor value of any number is the greatest Integer which is less than or equal to that number.

Examples:

Input: n = 5

Output: 2

Explanation: Since, 5 is not a perfect square, floor of square_root of 5 is 2.

Input: n = 4

Output: 2

Explanation: Since, 4 is a perfect square, so its square root is 2.

6.Segregate 0s and 1s

Given an array arr consisting of only 0's and 1's in random order. Modify the array in-place to segregate 0s onto the left side and 1s onto the right side of the array.

Examples :

Input: arr[] = [0, 0, 1, 1, 0]

Output: [0, 0, 0, 1, 1]

Explanation: After segregation, all the 0's are on the left and 1's are on the right. Modified array will be [0, 0, 0, 1, 1].

Input: arr[] = [1, 1, 1, 1]

Output: [1, 1, 1, 1]

Explanation: There are no 0s in the given array, so the modified array is [1, 1, 1, 1]

Expected Time Complexity: **O(n)**

Expected Auxiliary Space: **O(1)**