Assignment-09

1. Given a boolean 2D array, where each row is sorted. Find the row with the maximum number of 1s. Use the optimal algorithm.

   Input: **Output:**

   let matrix = [2
           [0,1,1,1],
           [0,0,1,1]
           [1,1,1,1],
           [0,0,0,0]
           ]

2. Given an integer array of coins[ ] of size N representing different types of denominations and an integer sum, the task is to count all combinations of coins to make a given value sum. Assume that you have an infinite supply of each type of coin.

   Input: **Output:**

   Let coins = [1,2,3], sum = 4   4
   Explanation: there are four solutions: [1, 1, 1, 1], [1, 1, 2], [2, 2], [1, 3].

   let coins= [2, 5, 3, 6], sum = 10   5
   Explanation: There are five solutions: [2,2,2,2,], [2,2,3,3], [2,2,6], [2,3,5] and [5,5].

3. Given an unsorted array of positive integers, find the number of triangles that can be formed with three different array elements as three sides of triangles. For a triangle to be possible from 3 values, the sum of any of the two values (or sides) must be greater than the third value (or third side).

   Input: **Output:**

   let arr = [4, 6, 3, 7]   3
   Explanation: There are three triangles possible [3, 4, 6], [4, 6, 7] and [3, 6, 7].
   Note that [3, 4, 7] is not a possible triangle.

   let arr = [10, 21, 22, 100, 101, 200, 300]   6
   Explanation: There can be 6 possible triangles: [10, 21, 22], [21, 100, 101], [22, 100, 101], [10, 100, 101], [100, 101, 200] and [101, 200, 300]

4. Given an array of integers. Write a program to find the K-th largest sum of contiguous subarray within the array of numbers that has both negative and positive numbers.

Input: **Output:**
arr = [20, -5, -1], K = 3   14

Explanation: All sum of contiguous subarrays are [20, 15, 14, -5, -6, -1]
so the 3rd largest sum is 14.

arr = [10, -10, 20, -40], k = 6   -10
 Explanation: The 6th largest sum among sum of all contiguous subarrays is -10.

5. Consider a scenario where you are provided with a binary array nums. We define a subarray as "alternating" if within that subarray, no two adjacent elements have the same value. Your objective is to count such alternating subarrays within the given binary array nums.

Input: **Output:**

nums = [0,1,1,1]   5
Explanation: The following subarrays are alternating: [0], [1], [1], [1], and [0,1].

6. You are given a string s and a character c. Return the total number of substrings of s that start and end with c.

Input: **Output:**
s = "abada", c = "a"   6
Explanation: Substrings starting and ending with "a" are: "abada", "abada", "abada", "abada", "abada", "abada".