DAY - 26

(1) Merge two sorted Arrays.

Approach :-

-> create a dist list3 (extra one.)

-> Simultaneously traverse given list1 and list2

-> Pick smaller of current elements in list1 and list2, copy this smaller element to next position in list3 and move ahead in list3 and the list whose element is picked.

-> If there are remaining elements in list1 or list2, copy them also in list3.

(2) Find Subarray with given Sum.

Given an unsorted array of nonnegative integers, find a continuous subarray which adds to a given number.

Example :-

input => listtu = [1, 2, 7, 0, 9], sum = 16

o/p => sum found between indices 2 and 4.

Simple Approach :-

→ consider all subarrays one by one and check the sum of every subarray using loops.

Algorithm :-

→ Traverse the array/list from start to end.

→ From every index start another loop from i to the end of array to get all subarray starting from i, ~~keep~~

⇒ keep a variable sum to calculate the sum.

→ For every index in inner loop update sum = sum + array[j]

→ If the sum is equal to the given sum then print the subarray.