DAILY ONLINE ACTIVITIES SUMMARY

Date:	**02/6/2020**	**Name:**	**Sumana Rehman**
Sem & Sec	**8th Sem B**	**USN:**	**4AL16CS107**
Online Test Summary			
Subject	--		
Max. Marks	--	**Score**	--
Certification Course Summary			
Course	**GUVI's Robotic Process Automation**		
Certificate Provider	**www.guvi.in/rpa**	**Duration**	**3 hrs**
Coding Challenges			
Problem Statement: Write a C Program to find inversion count of array.			
Status: Completed			
Uploaded the report in Github	**Yes**		
If yes Repository name	**Alvas-education-foundation/Sumana**		
Uploaded the report in slack	**yes**		

Certification Course Details:

Coding Challenges:

```
#include <stdio.h>
int _mergeSort(int arr[], int temp[], int left, int right);
int merge(int arr[], int temp[], int left, int mid, int right);
int mergeSort(int arr[], int array_size)
{
        int* temp = (int*)malloc(sizeof(int) * array_size);
        return _mergeSort(arr, temp, 0, array_size - 1);
}
int _mergeSort(int arr[], int temp[], int left, int right)
{
        int mid, inv_count = 0;
        if (right > left) {
                _mergeSortAndCountInv()
```

```
        for each of the parts */
                mid = (right + left) / 2;
                inv_count += _mergeSort(arr, temp, left, mid);
                inv_count += _mergeSort(arr, temp, mid + 1, right);
                inv_count += merge(arr, temp, left, mid + 1, right);
        }
        return inv_count;
}

int merge(int arr[], int temp[], int left, int mid, int right)
{
        int i, j, k;
        int inv_count = 0;
        i = left;
        j = mid;
        k = left;
        while ((i <= mid - 1) && (j <= right)) {
                if (arr[i] <= arr[j]) {
                        temp[k++] = arr[i++];
                }
                else {
                        temp[k++] = arr[j++];
                        inv_count = inv_count + (mid - i);
                }
        }
        while (i <= mid - 1)
                temp[k++] = arr[i++];
        while (j <= right)
                temp[k++] = arr[j++];
        for (i = left; i <= right; i++)
                arr[i] = temp[i];
        return inv_count;
```

```
}
int main(int argv, char** args)
{
	int arr[] = { 1, 20, 6, 4, 5 };
	printf(" Number of inversions are %d \n", mergeSort(arr, 5));
	getchar();
	return 0;
}
```