Day-37 of the #101 days of coding challenge

Merge Sort:--

Code:-

```
#include <iostream>
using namespace std;

void merge(int *arr, int start, int end)
{
    int i;
    // here each time also need to find the mis elements
    int mid = (start + end)/2;

    // finding the length of the both array
    int len1 = mid - start + 1;
    int len2 = end - mid;
    //creating two index for traversing the array's elements
    int index1 = 0;
    int index2 = 0;
    // creating the two array to compy the left and right data
```

```
    int *firstArray = new int[len1];
    int *secondArray = new int[len2];

    // coping the elements into the array based on size of
elements

    int mainArrayIndex = start; // starting from the first index
    for(i= 0; i< len1; i++)
    {
        firstArray[i] = arr[mainArrayIndex++];
    }

    mainArrayIndex = mid+1;
    for(i= 0; i< len2; i++)
    {
        secondArray[i] = arr[mainArrayIndex++];
    }

    mainArrayIndex = start;
```

```
    // sorting and adding the array's data into the one single main Array
    while(index1 < len1 && index2 < len2)

    {
        if(firstArray[index1]<secondArray[index2])
        {
            arr[mainArrayIndex++] = firstArray[index1++];
        }
        else{
            arr[mainArrayIndex++] = secondArray[index2++];
        }
    }

    // checking the condition wheather data is successfully added into main array
    while(index1<len1)
    {
        arr[mainArrayIndex++] = firstArray[index1++];
    }
```

```
    while(index2<len2)
    {
        arr[mainArrayIndex++] = secondArray[index2++];
    }

    // free the memory allocated

    delete[] firstArray;
    delete[] secondArray;
}
void mergeSort(int *arr, int start, int end)
{
    if(start>=end)
    {
        return;
    }
   // finding mid value
   int mid = (start + end)/2;
```

```
    // sorting left elements
    mergeSort(arr, start, mid);

    // sorting right side array
    mergeSort(arr, mid+1, end);

    // merging both array after sorting
    merge(arr, start, end);

}

int main() {

    int n;
    cout<<"Enter the size of the array"<<endl;
    cin>>n;

    int arr[n];

    cout<<"Enter the Elements"<<endl;
```

```
   for(int i = 0; i<n; i++)
   {
      cin>>arr[i];
   }

   // calling the function
   mergeSort(arr, 0, n-1);

   cout<<"Sorted Elements"<<endl;

   for(int i = 0; i<n; i++)
   {
      cout<<arr[i]<<"  ";
   }

    return 0;
}
```

Output:-

```
Enter the size of the array
5
Enter the Elements
3 2 7 1 5
Sorted Elements
1  2  3  5  7
--------------------------------
```