MERGE SORT

Q.1) Sort a given set of N integer elements using the Merge Sort technique and compute its time taken. Run the program for different values of N and record the time taken to sort.

```
#include <stdio.h>
#include <stdlib.h>

void merge(int A[], int si, int mid, int ei)
{
   int i, j, k, c[100];
   i = si;
   j = mid + 1;
   k = si;

   while (i <= mid && j <= ei)
   {
      if (A[i] < A[j])
         c[k++] = A[i++];
      else
         c[k++] = A[j++];
   }
   while (i <= mid)
      c[k++] = A[i++];
   while (j <= ei)
      c[k++] = A[j++];

   for (i = si; i <= ei; i++)
   {
      A[i] = c[i];
   }
}

void mergeSort(int A[], int si, int ei)
{
   int mid;
   if (si < ei)
   {
```

```
        mid = (si + ei) / 2;
        mergeSort(A, si, mid);
        mergeSort(A, mid + 1, ei);
        merge(A, si, mid, ei);
    }
}

int main()
{
    int n;
    int A[10];
    printf("Enter the number of elements: ");
    scanf("%d", &n);
    printf("Enter array elements:\n");
    for (int i = 0; i < n; i++)
    {
        scanf("%d", &A[i]);
    }
    printf("\nthe sorted elements are:\n");
    mergeSort(A, 0, n - 1);
    for (int i = 0; i < n; i++)
    {
        printf("%d ", A[i]);
    }
}
```

OUTPUT:

```
Enter the number of elements: 5
Enter array elements:
12 90 72 5 -1

the sorted elements are:
-1 5 12 72 90
```