PROGRAM 4

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

CODE:

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>
#define MAX 50
void mergeSort(int[],int,int);
void simpleMerge(int[],int,int,int);
main()
{
   int array[MAX],size,i;
   printf("Enter the size of an array...\n");
   scanf("%d",&size);
   printf("Enter array elements are...\n");
   for(i=0;i<size;i++)
   {
      scanf("%d",&array[i]);
   }
   mergeSort(array,0,size-1);
   printf("\nAfter sorting array elements are...\n");
   for(i=0;i<size;i++)
      printf("%d\t",array[i]);
}
void mergeSort(int a[],int low,int high)
{
   int mid,i;
   if(low<high)
   {
      mid=(low+high)/2;
      mergeSort(a,low,mid);
      mergeSort(a,mid+1,high);
```

```
        simpleMerge(a,low,mid,high);
    }
}
void simpleMerge(int a[],int low,int mid,int high)
{
    int i=low,j=mid+1,k=low;
    int c[50];
    while(i<=mid && j<=high)
    {
        if(a[i]<a[j])
        c[k++]=a[i++];
        else
        c[k++]=a[j++];
    }
    while(i<=mid)
        c[k++]=a[i++];
    while(j<=high)
        c[k++]=a[j++];
    for(i=low;i<=high;i++)
        a[i]=c[i];
}
```

OUTPUT:

```
Enter the size of an array...
7
Enter array elements are...
99
88
77
66
55
44
33

After sorting array elements are...
33      44      55      66      77      88      99
```

OBSERVATION:

Merge Sort.

```
#include<stdio.h>
#define MAX 50.
void mergeSort (int[], int, int);
void simpleMerge (int[], int, int);
main()
{
    int array[MAX], size, i;
    printf ("Enter the size of an array...\n");
    scanf ("%d", &size);
    printf ("Enter array elements ...\n");
    for(i=0; i<size; i++)
    {
        scanf ("%d", &array[i]);
    }
    mergeSort (array, 0, size-1);
    printf ("After sorting array elements are :-\n");
    for (i=0; i<size; i++)
        printf ("%d\t", array[i]);
}
void mergeSort (int a[], int low, int high)
{
    int mid, i;
    if (low < high)
    {
        mid = (low + high)/2;
        mergeSort (a, low, mid);
        mergeSort (a, mid+1, high);
        simpleMerge (a, low, mid, high);
    }
}
void simpleMerge (int a[], int low, int mid, int high)
{
    int i=low, j=mid+1, k=low;
```

```
int c[50];
while (i<=mid && j<=high)
{
    if (a[i] < a[j])
        c[k++] = a[i++];
    else
        c[k++] = a[j++];
}
while (i<=mid)
    c[k++] = a[i++];
while (j<=high)
    c[k++] = a[j++];
for (i=low; i<=high; i++)
    a[i] = c[i];
}
```

Output:

```
Enter the size of an array...
6
Enter array elements...
45 56 23 87 45 90
After sorting array elements are...
23 45 45 56 87 90
```