Merge Sort

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

void merge(int arr[], int l, int m, int r)
{
    int i, j, k;
    int n1 = m - l + 1;
    int n2 =  r - m;
    int L[n1], R[n2];

    for (i = 0; i < n1; i++)
        L[i] = arr[l + i];
    for (j = 0; j < n2; j++)
        R[j] = arr[m + 1+ j];

    i = 0;
    j = 0;
    k = l;
    while (i < n1 && j < n2)
```

```
{
    if (L[i] <= R[j])
    {
        arr[k] = L[i];
        i++;
    }
    else
    {
        arr[k] = R[j];
        j++;
    }
    k++;
}

while (i < n1)
{
    arr[k] = L[i];
    i++;
    k++;
}

while (j < n2)
{
    arr[k] = R[j];
```

```
        j++;
        k++;
    }
}

void mergeSort(int arr[], int l, int r)
{
    if (l < r)
    {
         int m = l+(r-l)/2;
        mergeSort(arr, l, m);
        mergeSort(arr, m+1, r);

        merge(arr, l, m, r);
    }
}

void printArray(int A[], int size)
{
    int i;
    for (i=0; i < size; i++)
        printf("%d ", A[i]);
    printf("\n");
}
```

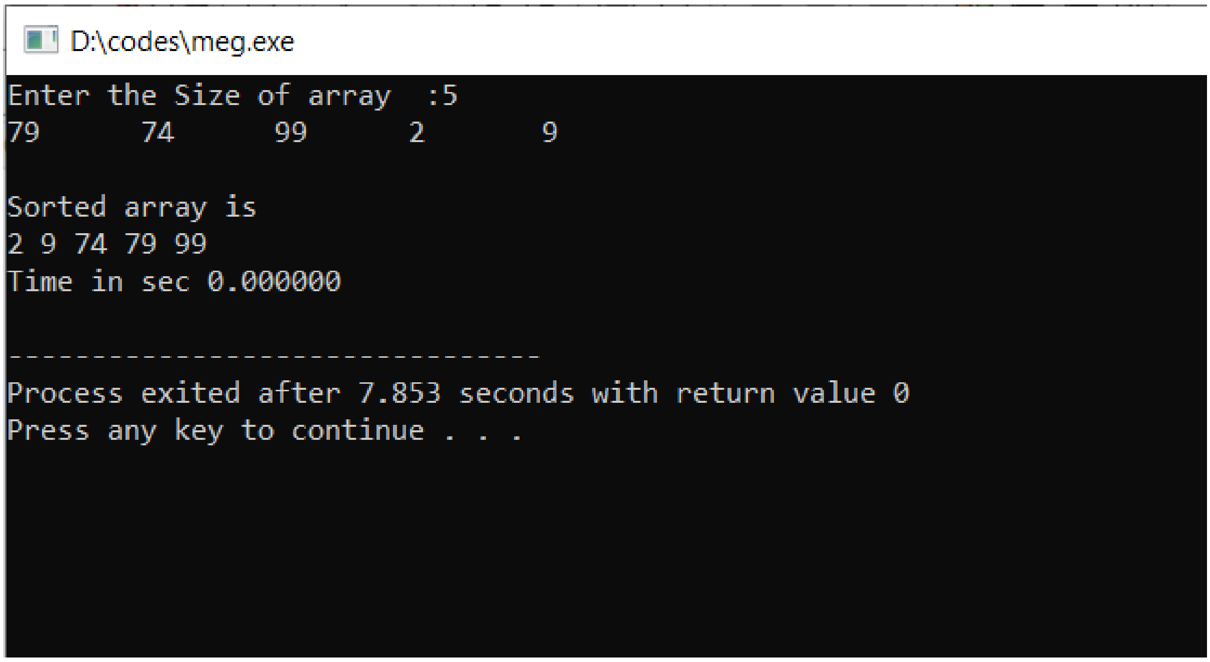
```
int main(){
  int last,arr[5000],j,temp;
  clock_t end,start;
printf("Enter the Size of array  :");
scanf("%d",&last);
srand(time(NULL));
for(int i=0;i<last;i++){
  arr[i]=rand()%100;
  printf("%d\t",arr[i]);
}printf("\n");
start=clock();

mergeSort(arr, 0, last - 1);

for(j=0;j<500000;j++){ temp=38/600;}
                 end=clock();

end=clock();
 printf("\nSorted array is \n");
    printArray(arr, last);
          printf("Time in sec %f\n",(((double)(end-
start))/CLOCKS_PER_SEC));
```

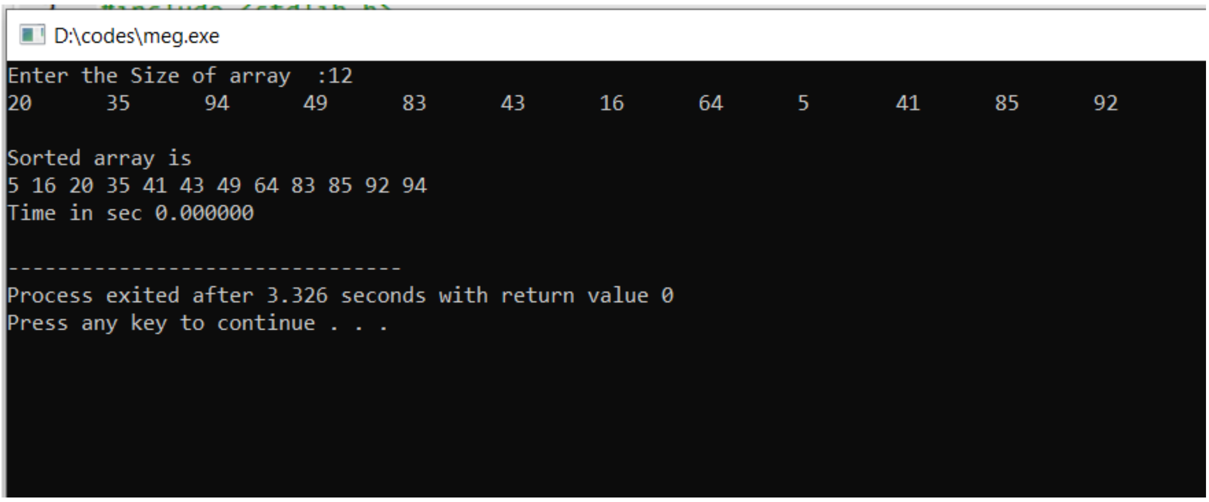
```
        return 0;
}
```



```
Enter the Size of array  :5
79      74      99      2       9

Sorted array is
2 9 74 79 99
Time in sec 0.000000

--------------------------------
Process exited after 7.853 seconds with return value 0
Press any key to continue . . .
```



```
Enter the Size of array  :12
20      35      94      49      83      43      16      64      5       41      85      92

Sorted array is
5 16 20 35 41 43 49 64 83 85 92 94
Time in sec 0.000000

--------------------------------
Process exited after 3.326 seconds with return value 0
Press any key to continue . . .
```