RISHUL M
1BM19CS173

4D

19CS4PCADA

① merge sort

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

double time_spent=0.0;
int n;

void merge(int arr[], int p, int q, int r) {

int n1=q-p+1;
int n2=r-q;
int L[n1] M[n2];

for (int i=0; i<n1; i++)
  L[i]=arr[p+i];
for (int j=0; j<n2; j++)
  M[j]=arr[q+1+j];

int i,j,k;
i=0;
j=0;
k=p;
```

```
while (i<n1 && j<n2) {
if(L[i]<=M[j]) {
arr[k]=L[i];
i++;
}
  else {
arr[k]=M[j];
  j++;
  }
k++;
}
while (i<n1) {
  arr[k]=L[i];
    i++;
  k++;
}
while (j<n2) {
    arr[k]=M[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 arr[], int size) {
  for (int i=0; i<size; i++)
    printf("%d", arr[i]);
    printf("\n");

}

  int main() {
  int arr[1000];
  printf ("enter the number of elements \n");
  scanf ("%d", &n);
```

```
for(int i=0; i<n; i++) {
arr[i] = rand() % 300;
}
clock_t start, end;
start = clock();
mergesort(arr, 0, n-1);
end = clock();
printf("Sorted Array: \n");
printArray(arr, n);
timespent = (double)(end-start) / (clocks_per_sec);
printf("Time elapsed is %f seconds\n", timespent);
}
```

modification:-

I am accepting 2 unsorted arrays separately and merge them and then I am applying merge sort

```
int main()
{
  int arr1[10], arr2[10], arr[20];
  int i, n1, n2, m, index=0;
  printf("Enter the number of elements in array1");
  scanf("%d", &n1);
  printf("Enter the elements");
  printf("\n****\n");
  for(i=0; i<n1; i++) {
  scanf("%d", &arr1[i]);
}
printf("Enter the no in array2:");
scanf("%d", &n2);
printf("Enter the elements");
for(i=0; i<n2; i++) {
    scanf("%d", &arr2[i]);
    m=n1+n2;
```

```
for(i=0; i<n1; i++) {
    scanf("%d", &arr2[i]);
    arr3[index] = arr1[i];
    index++;
}
for(i=0; i<n2; i++) {
    arr3[index] = arr2[i]
    index++;
}
printf("\n\n The merger array is");
for(i=0; i<m; i++) {
    printf("\t\n Arr[%d] = %d", i, arr3[i]);
    printf(" The sorted array is ");
mergesort(arr3, 0, m-1);
printf("\n");
printArray(arr3, m);
return 0;
```