

Name: Vedant Mhatre

Moodle ID: 18102055

Roll Number: 31

Modification 1:

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

```
void mergesort(int[],int,int);
```

```
void mergearray(int[],int,int,int);
```

```
void sort(int arr[],int first, int last, int arr2[])
```

```
{
    int middle=first+last/2;
    int i=first,j=middle,k=0;

    while(i<middle && j<last)
    {
        if(arr[i] < arr[j])
            arr2[k++] = arr[i++];
        else
            arr2[k++] = arr[j++];
    }

    while(i<middle)
        arr2[k++] = arr[i++];

    while(j<last)
        arr2[k++] = arr[j++];
}
```

```
void quicksort(int x[],int first,int last)
```

```
{
    int pivot,i,j,t;
    if(first<last)
    {
        pivot=first;
        i=first;
        j=last;
        while(i<j)
        {
            while(x[i]<=x[pivot]&& i<last)
```

```

        i++;
        while(x[j]>x[pivot])
            j--;
        if(i<j)
        {
            t=x[i];
            x[i]=x[j];
            x[j]=t;
        }
    }
    t=x[pivot];
    x[pivot]=x[j];
    x[j]=t;
    quicksort(x,first,j-1);
    quicksort(x,j+1,last);
}
}

```

```

void mergesort(int a[],int beg,int end)
{
    int mid;
    if(beg<end)
    {
        mid=(beg+end)/2;
        mergesort(a,beg,mid);
        mergesort(a,mid+1,end);
        mergearray(a,beg,mid,end);
    }
}

```

```

void mergearray(int a[],int beg,int mid,int end){
    int i,leftend,num,temp,j,k,b[50];
    for(i=beg;i<=end;i++)
        b[i]=a[i];
    i=beg;
    j=mid+1;
    k=beg;
    while((i<=mid)&&(j<=end)) {
        if(b[i]<=b[j])
        {
            a[k]=b[i];
            i++;
            k++;
        }
    }
}

```

```

        else {
            a[k]=b[j];
            j++;
            k++;
        }
    }

    if(i<=mid){
        while(i<=mid) {
            a[k]=b[i];
            i++;
            k++;
        }
    }
    else {
        while(j<=end){
            a[k]=b[j];
            j++;
            k++;
        }
    }
}

void main()
{
    int arr[10]={12,42,14,52,36,42,23,63,13,43};
    int arr2[10];
    printf("Before sorting array:\n");
    for(int i=0;i<10;i++)
        printf("%d\n",arr[i]);

    quicksort(arr,0,4);
    mergesort(arr,5,9);
    sort(arr,0,10,arr2);

    printf("After sorting array:\n");
    for(int i=0;i<10;i++)
        printf("%d\n",arr2[i]);
}

```

```
vedant@hp: ~/Documents/College/AOA/Exp4
File Edit View Search Terminal Help
vedant@hp:~/Documents/College/AOA/Exp4$ gcc part1.c -o part1;./part1
Before sorting array:
12
42
14
52
36
42
23
63
13
43
After sorting array:
12
13
14
23
36
42
42
43
52
63
vedant@hp:~/Documents/College/AOA/Exp4$
```

Modification 2:

```
#include<stdio.h>
#include<stdlib.h>
void mergesort(int[],int,int);
void mergearray(int[],int,int,int);

void sort(int arr[],int first, int last, int arr2[])
{
    int middle=first+last/2;
    int i=first,j=middle,k=0;

    while(i<middle && j<last)
    {
        if(arr[i] < arr[j])
            arr2[k++] = arr[i++];
        else
            arr2[k++] = arr[j++];
    }

    while(i<middle)
```

```

        arr2[k++] = arr[i++];

        while(j<last)
            arr2[k++] = arr[j++];
    }

```

```

void quicksort(int x[],int first,int last)
{
    int pivot,i,j,t;
    if(first<last)
    {
        pivot=first;
        i=first;
        j=last;
        while(i<j)
        {
            while(x[i]<=x[pivot]&& i<last)
                i++;
            while(x[j]>x[pivot])
                j--;
            if(i<j)
            {
                t=x[i];
                x[i]=x[j];
                x[j]=t;
            }
        }
        t=x[pivot];
        x[pivot]=x[j];
        x[j]=t;
        quicksort(x,first,j-1);
        quicksort(x,j+1,last);
    }
}

```

```

void mergesort(int a[],int beg,int end)
{
    int mid;
    if(beg<end)
    {
        mid=(beg+end)/2;

```

```

        mergesort(a,beg,mid);
        mergesort(a,mid+1,end);
        mergearray(a,beg,mid,end);
    }
}

void mergearray(int a[],int beg,int mid,int end){
    int i,leftend,num,temp,j,k,b[50];
    for(i=beg;i<=end;i++)
        b[i]=a[i];
    i=beg;
    j=mid+1;
    k=beg;
    while((i<=mid)&&(j<=end)) {
        if(b[i]<=b[j])
        {
            a[k]=b[i];
            i++;
            k++;
        }
        else {
            a[k]=b[j];
            j++;
            k++;
        }
    }

    if(i<=mid){
        while(i<=mid) {
            a[k]=b[i];
            i++;
            k++;
        }
    }
    else {
        while(j<=end){
            a[k]=b[j];
            j++;
            k++;
        }
    }
}

```

```

void main()

```

```
{  
    int n;  
    printf("Enter number of elements:\n");  
    scanf("%d",&n);  
    int* arr;  
    arr = (int*) malloc(n * sizeof(int));  
  
    printf("Enter elements:\n");  
    for (int i = 0; i < n; i++)  
        scanf("%d",&arr[i]);  
  
    quicksort(arr,0,(n/2)-1);  
    mergesort(arr,(n/2),n-1);  
    int arr2[n];  
    sort(arr,0,n,arr2);  
    printf("After sorting\n");  
    for(int i=0;i<n;i++)  
        printf("%d\n",arr2[i]);  
}
```

```
vedant@hp: ~/Documents/College/AOA/Exp4
File Edit View Search Terminal Help
vedant@hp:~/Documents/College/AOA/Exp4$ gcc part2.c -o part2;./part2
Enter number of elements:
10
Enter elements:
52
134
15
123
54
52
2365
24
85
64
After sorting
15
24
52
52
54
64
85
123
134
2365
vedant@hp:~/Documents/College/AOA/Exp4$
```