

WEEK 4

Sort a given set of N integer elements using Merge Sort technique

CODE:

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

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

```
void merge(int low,int mid,int high,int array[20],int mer[20])
```

```
{
```

```
    int i = low;
```

```
    int j = mid+1;
```

```
    int k = 0;
```

```
    while(i<=mid && j<=high)
```

```
    {
```

```
        if(array[i]<array[j])
```

```
        {
```

```
            mer[k] = array[i];
```

```
            i++;
```

```
            k++;
```

```
        }
```

```
    else
```

```
    {
```

```
        mer[k] = array[j];
```

```
        j++;
```

```
        k++;
```

```
    }
```

```
}
```

```
while (i <= mid)
```

```
{
```

```
    mer[k] = array[i];
```

```

        i++;
        k++;
    }

    while (j <= high)
    {
        mer[k] = array[j];
        j++;
        k++;
    }

    for(int i=0;i<k;i++)
    {
        array[low+i] = mer[i];
    }
}

void merge_sort(int low,int high,int array[20],int merged[20])
{
    if(low<high)
    {
        int mid = (low+high)/2;
        merge_sort(low,mid,array,merged);
        merge_sort(mid+1,high,array,merged);
        merge(low,mid,high,array,merged);
    }
}

int main()
{
    int n,array[30];
    printf("Enter no. of elements:");
    scanf("%d",&n);
    printf("Enter elements:");

```

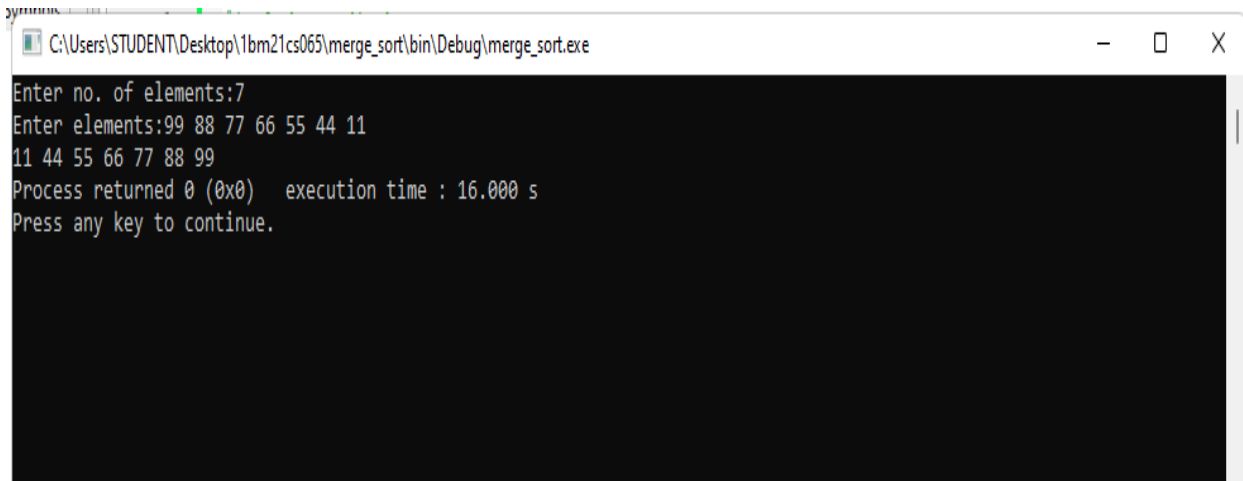
```
for(int i=0;i<n;i++)
{
    scanf("%d",&array[i]);
}

int merged[30];

merge_sort(0,n-1,array,merged);

for(int i=0;i<n;i++)
{
    printf("%d ",array[i]);
}
}
```

OUTPUT:



The screenshot shows a Windows command prompt window with the title bar "C:\Users\STUDENT\Desktop\1bm21cs065\merge_sort\bin\Debug\merge_sort.exe". The window contains the following text:

```
Enter no. of elements:7
Enter elements:99 88 77 66 55 44 11
11 44 55 66 77 88 99
Process returned 0 (0x0)   execution time : 16.000 s
Press any key to continue.
```