

# main.c

```
1 #include<stdio.h>
2 #include<time.h>
3 #include<stdlib.h>
4 void split(int[],int,int);
5
6 void combine(int[],int,int,int);
7 int main()
8 {
9     int a[15000],n, i,j,ch, temp;
10    clock_t start,end;
11
12    while(1)
13    {
14        printf("\n1:For manual entry of N value and
array elements");
15        printf("\n2:To display time taken for sorting
number of elements N in the range 500 to 14500");
16        printf("\n3:To exit");
17        printf("\nEnter your choice:");
18        scanf("%d", &ch);
19        switch(ch)
20        {
21            case 1:printf("\nEnter the number of elements: ")
22            ;
23            scanf("%d",&n);
24            printf("\nEnter array elements: ");
25            for(i=0;i<n;i++)
26            {
27                scanf("%d",&a[i]);
28            }
29            if(ch==1)
30            {
31                split(a,n,i,j);
32            }
33            else if(ch==2)
34            {
35                combine(a,n,i,j,temp);
36            }
37            else if(ch==3)
38            {
39                break;
40            }
41        }
42    }
43}
```

```
27    }
28    start=clock();
29    split(a,0,n-1);
30    end=clock();
31    printf("\nSorted array is: ");
32    for(i=0;i<n;i++)
33        printf("%d\t",a[i]);
34    printf("\n Time taken to sort %d numbers is %f
Secs",n, (((double)(end-start))/CLOCKS_PER_SEC));
35    break;
36 case 2:
37 n=500;
38 while(n<=14500) {
39     for(i=0;i<n;i++)
40     {
41         a[i]=n-i;
42     }
43     start=clock();
44     split(a,0,n-1);
45     for(j=0;j<500000;j++){ temp=38/600;}
46     end=clock();
47     printf("\n Time taken to sort %d numbers is %f
Secs",n, (((double)(end-start))/CLOCKS_PER_SEC));
48     n=n+1000;
49 }
50 break;
51 case 3: exit(0);
52 }
53 getchar();
```

```
main.c
-- -----
54 }
55 }
56 void split(int a[],int low,int high)
57 {
58     int mid;
59     if(low<high)
60     {
61         mid=(low+high)/2;
62         split(a,low,mid);
63         split(a,mid+1,high);
64         combine(a,low,mid,high);
65     }
66 }
67 void combine(int a[],int low,int mid,int high)
68 {
69     int c[15000],i,j,k;
70     i=k=low;
71     j=mid+1;
72     while(i<=mid&&j<=high)
73     {
74         if(a[i]<a[j])
75         {
76             c[k]=a[i];
77             ++k;
78             ++i;
79         }
80     else
81     {
82         c[k]=a[i];
```

## main.c

```
83     ++k;
84     ++j;
85 }
86 }
87 if(i>mid)
88 {
89 while(j<=high)
90 {
91     c[k]=a[j];
92     ++k;
93     ++j;
94 }
95 }
96 if(j>high)
97 {
98 while(i<=mid)
99 {
100    c[k]=a[i];
101    ++k;
102    ++i;
103 }
104 }
105 for(i=low;i<=high;i++)
106 {
107     a[i]=c[i];
108 }
109 }
```

► clang-7 -pthread -lm -o main main.c  
► ./main

Q X

- 1:For manual entry of N value and array elements  
2:To display time taken for sorting number of elements N in the range 500 to 14500  
3:To exit

Enter your choice:1

Enter the number of elements: 3

Enter array elements: 34 56 78

Sorted array is: 34 56 78

Time taken to sort 3 numbers is 0.000013 Secs

- 1:For manual entry of N value and array elements  
2:To display time taken for sorting number of elements N in the range 500 to 14500  
3:To exit

Enter your choice:■

# MERGE SORT GRAPH (N VALUE VS TIME)

