

```

1  #include<stdio.h>
2  #include <time.h>
3  void quicksort(int number[],int first,int last){
4      int i, j, pivot, temp;
5      if(first<last){
6          pivot=first;
7          i=first;
8          j=last;
9          while(i<j){
10             while(number[i]<=number[pivot]&& i<last)
11                 i++;
12             while(number[j]>number[pivot])
13                 j--;
14             if(i<j){
15                 temp=number[i];
16                 number[i]=number[j];
17                 number[j]=temp;
18             }
19         }
20         temp=number[pivot];
21         number[pivot]=number[j];
22         number[j]=temp;
23         quicksort(number,first,j-1);
24         quicksort(number,j+1,last);
25     }
26 }
27 int main(){

```



```

19     }
20     temp=number[pivot];
21     number[pivot]=number[j];
22     number[j]=temp;
23     quicksort(number,first,j-1);
24     quicksort(number,j+1,last);
25 }
26 }
27 int main(){
28     int i, n, count, number[n];
29     printf("Enter number of elements in the array:\n");
30     scanf("%d",&count);
31     printf("Enter %d numbers\n", count);
32     for(i=0;i<count;i++)
33         scanf("%d",&number[i]);
34     clock_t begin = clock();
35     quicksort(number,0,count-1);
36     printf("Printing the sorted array:\n");
37     for(i=0;i<count;i++){
38         printf(" %d",number[i]);
39     }
40     clock_t end = clock();
41     double time_spent = (double)(end - begin) / CLOCKS_PER_SEC;
42     printf("\nExecution Time : %.10fseconds\n", time_spent);
43     return 0;
44 }

```

number[i]-number[i].

Enter number of elements in the array:

5

Enter 5 numbers

4 9 10 3 9

Printing the sorted array:

3 4 9 9 10

Execution Time : 0.0000220000seconds

...Program finished with exit code 0

Press ENTER to exit console.