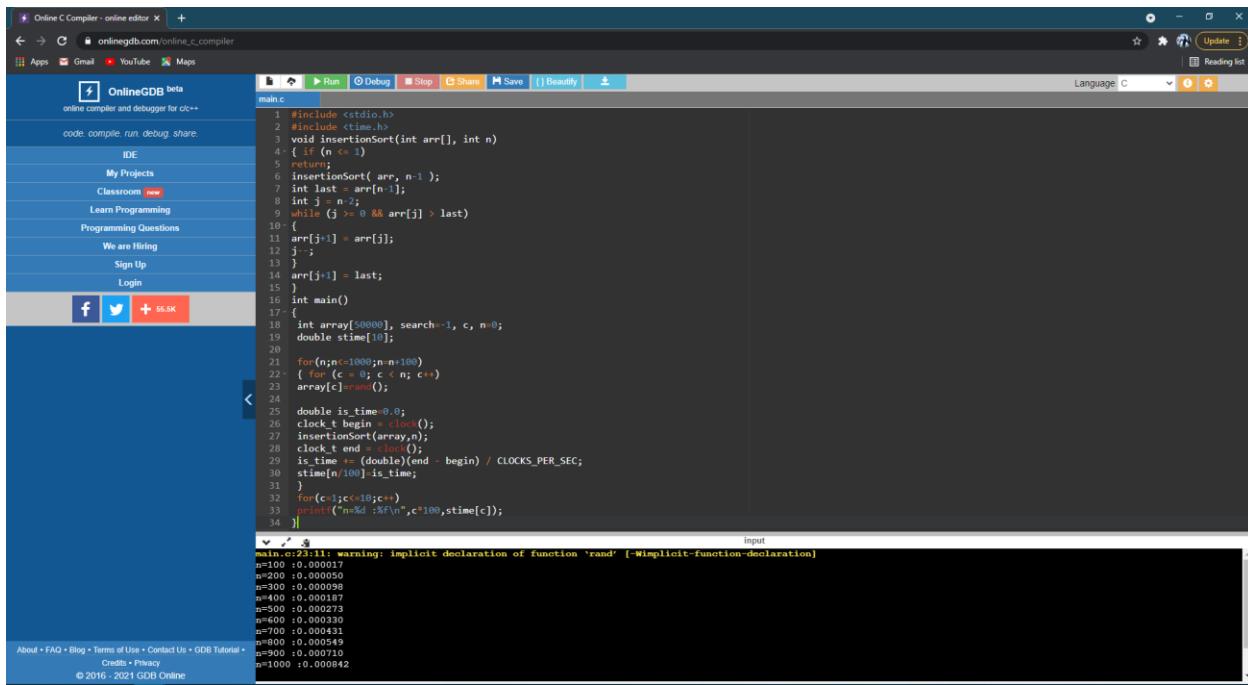


## OUTPUT FOR INSERT SORT



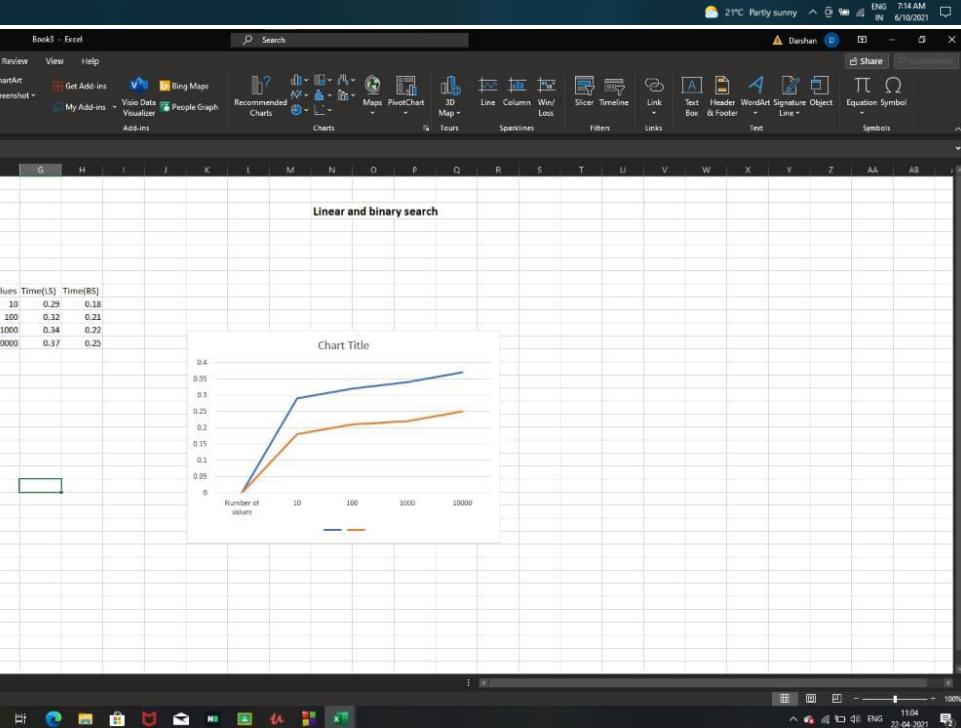
```

main.c
1 #include <stdio.h>
2 #include <time.h>
3 void insertionSort(int arr[], int n)
4 {
5     if (n <= 1)
6         return;
7     int last = arr[n-1];
8     int j = n-2;
9     while (j >= 0 && arr[j] > last)
10    {
11        arr[j+1] = arr[j];
12        j--;
13    }
14    arr[j+1] = last;
15 }
16 int main()
17 {
18     int array[50000], search=-1, c, n=0;
19     double stime[10];
20
21     for(n;n<1000;n+=100)
22     {
23         for(c=c; c < n; c++)
24             array[c]=rand();
25
26         double i_time=0.0;
27         clock_t begin = clock();
28         insertionSort(array,n);
29         clock_t end = clock();
30         i_time += (double)(end - begin) / CLOCKS_PER_SEC;
31         stime[n/100] = i_time;
32     }
33     for(c;c<10;c++)
34         printf("n=%d :%f\n",c*100,stime[c]);
35 }

```

min.c(23): warning: implicit declaration of function 'rand' [-Wimplicit-function-declaration]

n	i_time
100	0.000017
200	0.000050
300	0.000098
400	0.000187
500	0.000273
600	0.000330
700	0.000341
800	0.000549
900	0.000710
1000	0.000842

Linear and binary search

Number of values	Time(LS)	Time(BS)
10	0.29	0.18
100	0.32	0.21
1000	0.34	0.22
10000	0.37	0.25

## CODE for insert sort

```
#include <stdio.h>

#include <time.h>

void insertionSort(int arr[], int n)
```

```

{ if (n <= 1)
return;
insertionSort( arr, n-1 );
int last = arr[n-1];
int j = n-2;
while (j >= 0 && arr[j] > last)
{
arr[j+1] = arr[j];
j--;
}
arr[j+1] = last;
}

int main()
{
int array[50000], search=-1, c, n=0;
double stime[10];

for(n;n<=1000;n=n+100)
{ for (c = 0; c < n; c++)
array[c]=rand();

double is_time=0.0;
clock_t begin = clock();
insertionSort(array,n);
clock_t end = clock();
is_time += (double)(end - begin) / CLOCKS_PER_SEC;
stime[n/100]=is_time;
}

for(c=1;c<=10;c++)

```

```

printf("n=%d :%f\n",c*100,stime[c]);
}

```

## OUTPUT FOR SEARCH

```

OnlineC Compiler - online editor x
onlinegdb.com/online_c_compiler
code compile run debug share
IDE
My Projects
Classroom
Learn Programming
Programming Questions
We are Hiring
Sign Up
Login
Facebook Twitter + 66.8K
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy © 2016 - 2021 GDB Online

```

```

#include<stdio.h>
#include <time.h>
clock_t start, end;
int linear_search(int arr[],int n,int k)
{
    if(n<0)
        return 0;
    if(arr[n] == k)
        return 1;
    else
        return 0;
    return linear_search(arr,n-1,k);
}
int binary_search(int arr[],int start,int end,int k)
{
    if(start>end)
        return 0;
    int mid = (start + end) / 2;
    if(arr[mid] == k)
        return 1;
    else if(arr[mid]>k)
        end = mid - 1;
    else
        start = mid + 1;
    return binary_search(arr,start,end,k);
}
int main()
{
    int n;
    printf("Enter the size of array:");
    scanf("%d",&n);
    int choice;
    int arr[n];
    printf("\n");
    Enter the size of array:3
    Enter the element to search:2
    1 - Linear search
    2 - Binary search
    3 - Exit
    Enter your choice:2
    Element not found
    time taken to search value in 3 numbers is 0.170609 Secs
    1 - Linear search
    2 - Binary search

```

21°C Party sunny ENG 7:24 AM US 6/10/2021

```

OnlineC Compiler - online editor x
onlinegdb.com/online_c_compiler
code compile run debug share
IDE
My Projects
Classroom
Learn Programming
Programming Questions
We are Hiring
Sign Up
Login
Facebook Twitter + 66.8K
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy © 2016 - 2021 GDB Online

```

```

printf("Enter the element to search:");
scanf("%d",&k);
while(choice!=3)
{
    printf("1 - Linear search\n");
    printf("2 - Binary search\n");
    printf("3 - Exit\n");
    printf("Enter your choice:");
    scanf("%d",&choice);
    if(choice==1)
    {
        start=clock();
        for(i=0;i<80000000;i++);
        int a=linear_search(arr,0,n-1,k);
        end=clock();
        if(a==1)
            printf("Element found\n");
        else
            printf("Element not found\n");
        printf("\n Time taken to search value in %d numbers is %f Secs\n",n, ((double)(end-start))/CLOCKS_PER_SEC);
    }
    else if(choice==2)
    {
        start=clock();
        for(i=0;i<80000000;i++);
        int b=binary_search(arr,0,n-1,k);
        end=clock();
        if(b==1)
            printf("Element found\n");
        else
            printf("Element not found\n");
        printf("\n Time taken to search value in %d numbers is %f Secs\n",n, ((double)(end-start))/CLOCKS_PER_SEC);
    }
    else if(choice==3)
        break;
}
Enter the size of array:3
Enter the element to search:2
1 - Linear search
2 - Binary search
3 - Exit
Enter your choice:2
Element not found
    time taken to search value in 3 numbers is 0.170609 Secs
    1 - Linear search
    2 - Binary search

```

21°C Party sunny ENG 7:24 AM US 6/10/2021

Online C Compiler - online editor

code compile run debug share.

IDE

My Projects

Classroom

Learn Programming

Programming Questions

We are Hiring

Sign Up

Login

Facebook Twitter LinkedIn

```
main.c
41 printf("3 - Exit\n");
42 printf("Enter your choice:");
43 scanf("%d", &choice);
44 if(choice==3)
45 {
46     start=clock();
47     for(i=0;i<800000000;i++);
48     int a = linear_search(arr,n-3,k);
49     end=clock();
50     if(a==1)
51         printf("Element found\n");
52     else
53         printf("Element not found\n");
54     printf("\n Time taken to search value in %d numbers is %f Secs\n",n, (((double)(end-start))/CLOCKS_PER_SEC));
55 }
56 else if(choice==2)
57 {
58     start=clock();
59     for(i=0;i<800000000;i++);
60     int b = binary_search(arr,0,n-1,k);
61     end=clock();
62     if(b==1)
63         printf("Element Found\n");
64     else
65         printf("Element not found\n");
66     printf("\n Time taken to search value in %d numbers is %f Secs\n",n, (((double)(end-start))/CLOCKS_PER_SEC));
67 }
68 else if(choice==3)
69 {
70     break;
71 }
72 else
73 {
74 }
```

Enter the size of array:3

Enter the element to search:

1 - Linear search  
2 - Binary search  
3 - Exit

Enter your choice:2

Element not found

Time taken to search value in 3 numbers is 0.170609 Secs

1 - Linear search  
2 - Binary search  
3 - Exit

About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy © 2016 - 2021 GDB Online

Online C Compiler - online editor

code compile run debug share.

IDE

My Projects

Classroom

Learn Programming

Programming Questions

We are Hiring

Sign Up

Login

Facebook Twitter LinkedIn

```
main.c
26 }
27 int main()
28 {
29     int n,i;
30     printf("Enter the size of array:");
31     scanf("%d",&n);
32     int choice;
33     int arr[n];
34     printf("\n");
35     printf("Enter the element to search:");
36     scanf("%d",&k);
37     while(choice!=3)
38     {
39         printf("1 - Linear search\n");
40         printf("2 - Binary search\n");
41         printf("3 - Exit\n");
42         printf("Enter your choice:");
43         scanf("%d", &choice);
44         if(choice==3)
```

Enter the size of array:3

Enter the element to search:

1 - Linear search  
2 - Binary search  
3 - Exit

Enter your choice:2

Element not found

Time taken to search value in 3 numbers is 0.170609 Secs

1 - Linear search  
2 - Binary search  
3 - Exit

Enter your choice:1

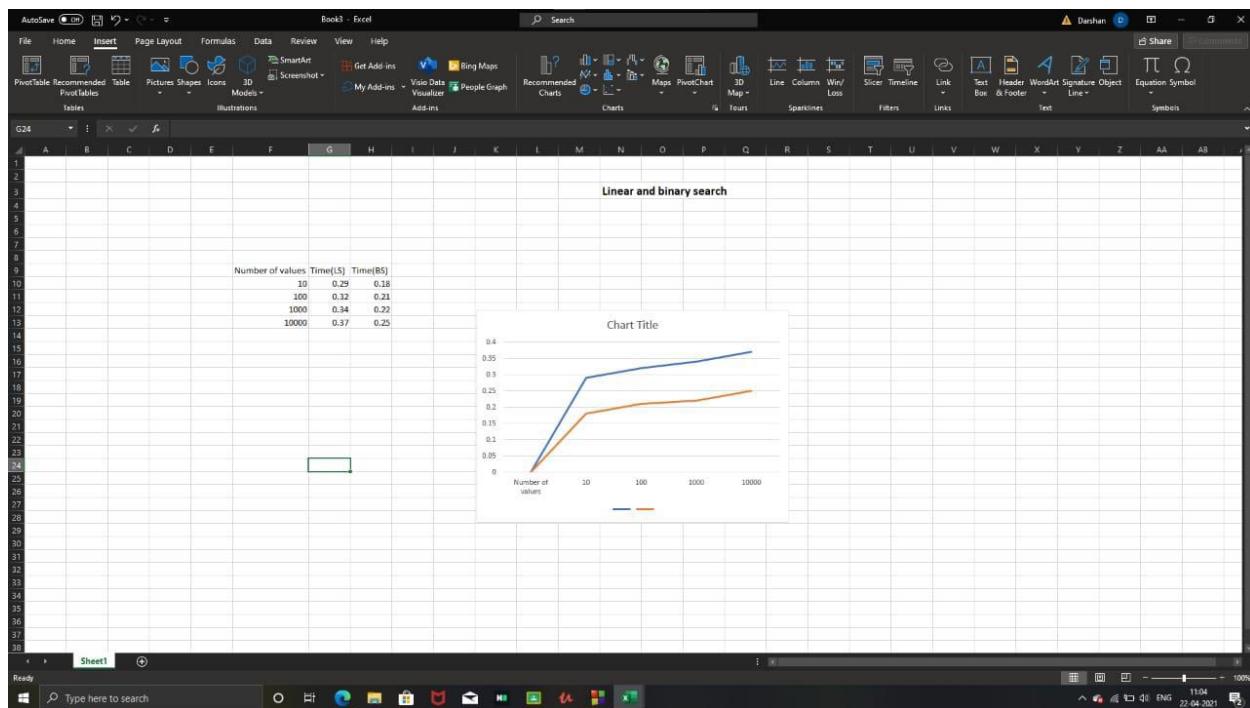
Element not found

Time taken to search value in 3 numbers is 0.169888 Secs

1 - Linear search  
2 - Binary search  
3 - Exit

Enter your choice:[]

About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy © 2016 - 2021 GDB Online



## CODE FOR SEARCH

```
#include<stdio.h>
#include <time.h>
clock_t start, end;
int linear_search(int arr[],int n,int k)
{
    if(n<0)
        return 0;
    if(arr[n] == k)
        return 1;
    else
        return 0;
    return linear_search(arr,n-1,k);
}
int binary_search(int arr[],int start,int end,int k)
{
```

```
if(start>end)
return 0;

int mid = (start + end )/ 2;

if(arr[mid]==k)
return 1;
else if(arr[mid]>k)
end = mid - 1;
else
start = mid + 1;

return binary_search(arr,start,end,k);
}

int main()
{
int n,i;
printf("Enter the size of array:");
scanf("%d",&n);
int choice;
int arr[n],k;
printf("\n");
printf("Enter the element to search:");
scanf("%d",&k);
while(choice!=3)
{
printf("1 - Linear search\n");
printf("2 - Binary search\n");
printf("3 - Exit\n");
printf("Enter your choice:");
scanf("%d",&choice);
if(choice==1)
```

```

{
start=clock();

for(i=0;i<80000000;i++);

int a = linear_search(arr,n-1,k);

end=clock();

if(a==1)

printf("Element found\n");

else

printf("Element not found\n");

printf("\n Time taken to search value in %d numbers is %f Secs\n",n, (((double)(end-
start))/CLOCKS_PER_SEC));

}

else if(choice==2)

{

start=clock();

for(i=0;i<80000000;i++);

int b = binary_search(arr,0,n-1,k);

end=clock();

if(b==1)

printf("Element found\n");

else

printf("Element not found\n");

printf("\n Time taken to search value in %d numbers is %f Secs\n",n, (((double)(end-
start))/CLOCKS_PER_SEC));

}

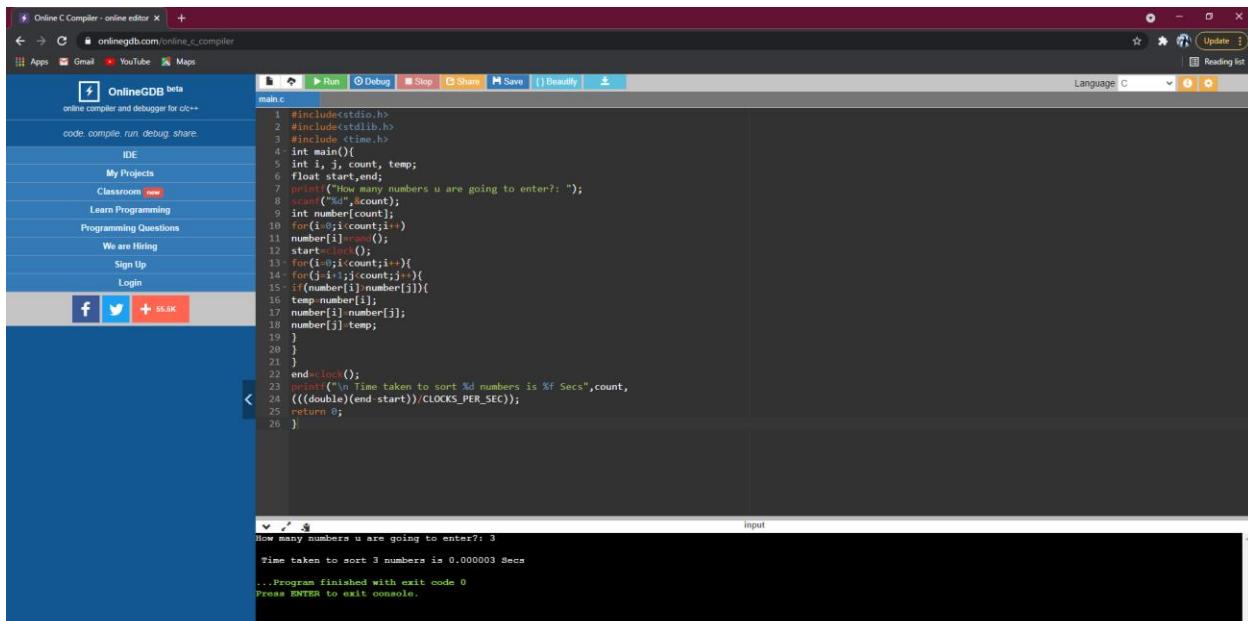
else if(choice==3)

break;

else

printf("Invalid choice;");}}
```

## OUTPUT FOR SELECTION SORT



The screenshot shows the Online C Compiler interface. The code in main.c implements Selection Sort. The output window displays the program's execution, including user input, sorting process, and completion message.

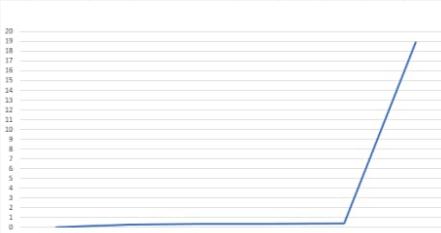
```
main.c
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include <time.h>
4 int main(){
5     int i, j, count, temp;
6     float start,end;
7     printf("How many numbers u are going to enter?: ");
8     scanf("%d",&count);
9     int number[count];
10    for(i=0;i<count;i++){
11        number[i]=rand();
12    }
13    start=clock();
14    for(i=0;i<count;i++){
15        for(j=i+1;j<count;j++){
16            if(number[i]>number[j]){
17                temp=number[i];
18                number[i]=number[j];
19                number[j]=temp;
20            }
21        }
22    }
23    end=clock();
24    printf("\n Time taken to sort %d numbers is %f Secs",count,
25    ((double)(end-start))/CLOCKS_PER_SEC));
26    return 0;
27 }
```

Output:

```
How many numbers u are going to enter?: 3
Time taken to sort 3 numbers is 0.000003 Secs
...Program finished with exit code 0
Press ENTER to exit console.
```

The screenshot also shows a Microsoft Excel spreadsheet titled "Book1 - Excel". It contains a table titled "Selection Sort" with data points for different array sizes and their execution times. A scatter plot is overlaid on the table, showing a non-linear increase in time as the number of values increases.

Number of values	Time(s)
10	0.26
100	0.32
1000	0.36
10000	0.43
100000	18.95



## CODE FOR SELECTION SORT

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

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

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

```
int main(){
    int i, j, count, temp;
    float start,end;
    printf("How many numbers u are going to enter?: ");
    scanf("%d",&count);
    int number[count];
    for(i=0;i<count;i++)
        number[i]=rand();
    start=clock();
    for(i=0;i<count;i++){
        for(j=i+1;j<count;j++){
            if(number[i]>number[j]){
                temp=number[i];
                number[i]=number[j];
                number[j]=temp;
            }
        }
    }
    end=clock();
    printf("\n Time taken to sort %d numbers is %f Secs",count,
        (((double)(end-start))/CLOCKS_PER_SEC));
    return 0;
}
```