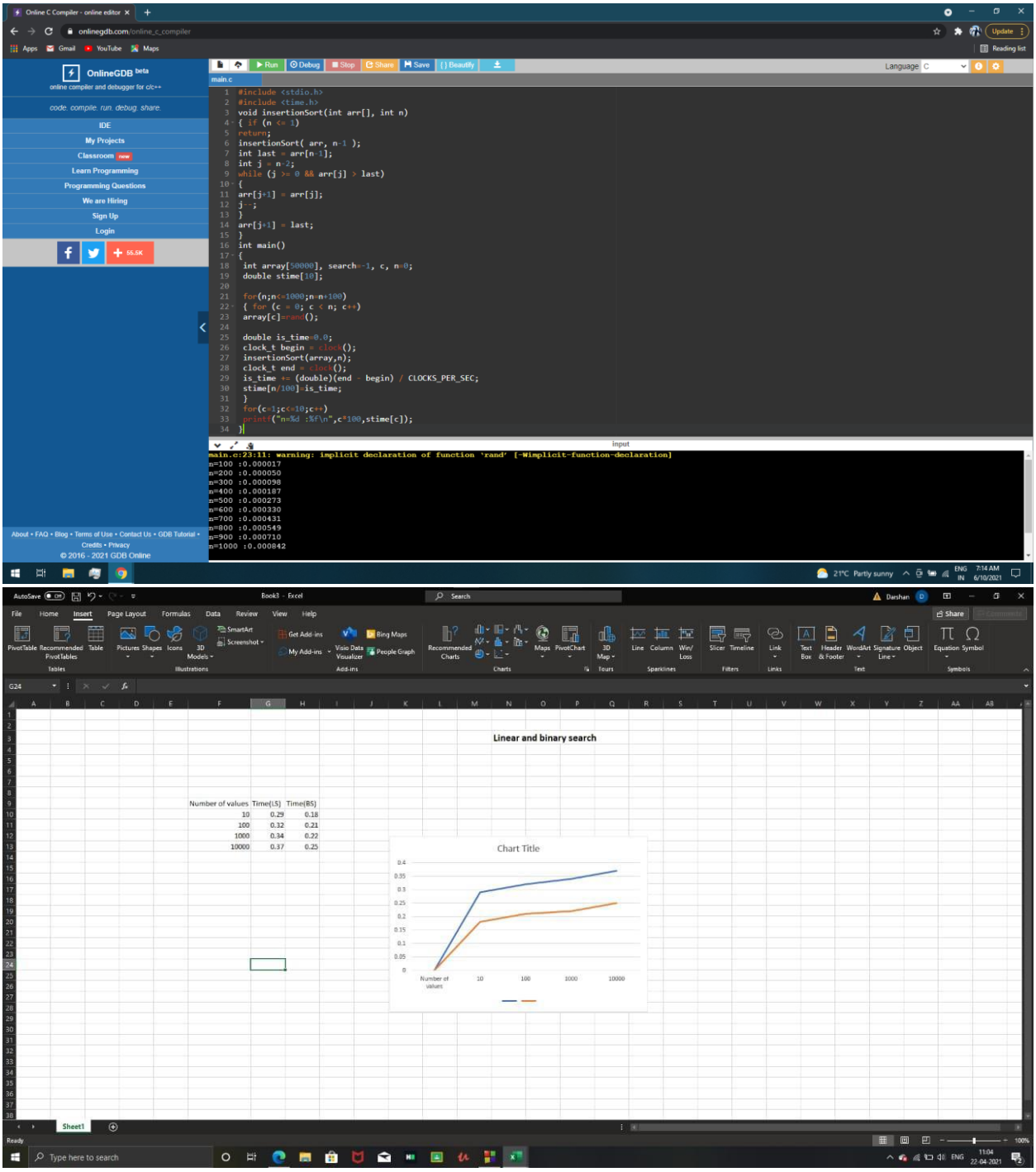
OUTPUT FOR INSERT SORT



CODE for insert sort

#include <stdio.h>

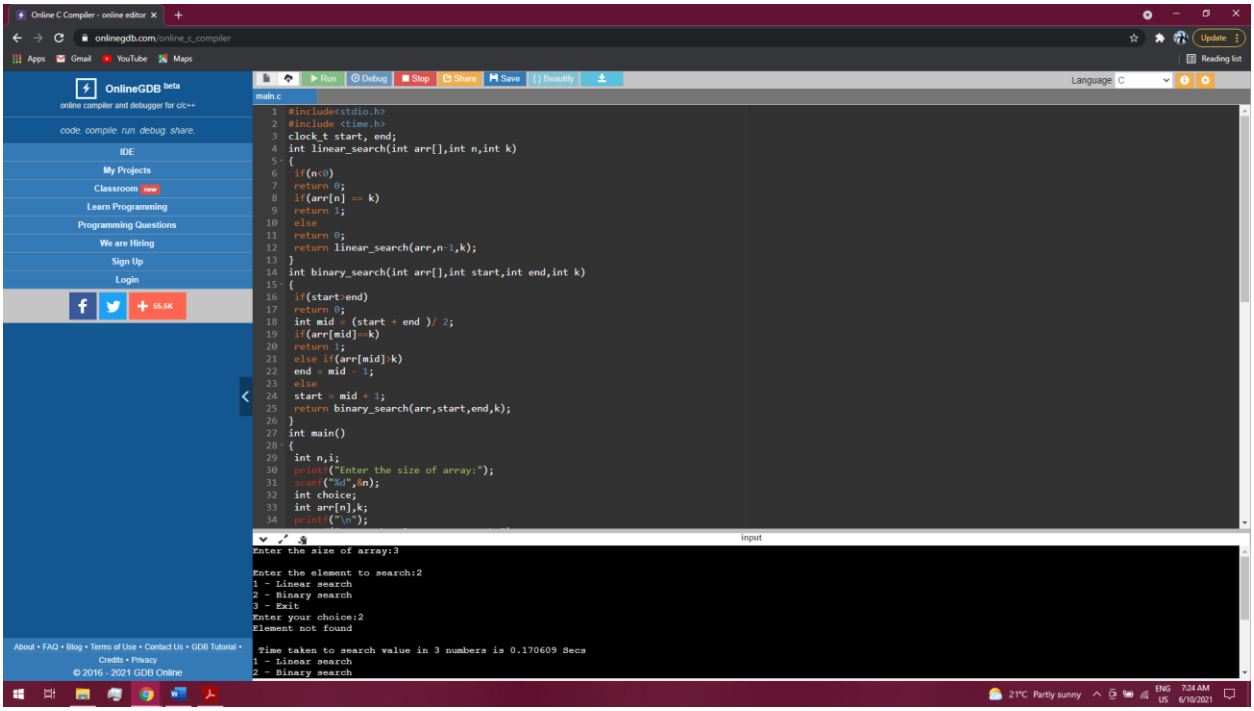
#include <time.h>

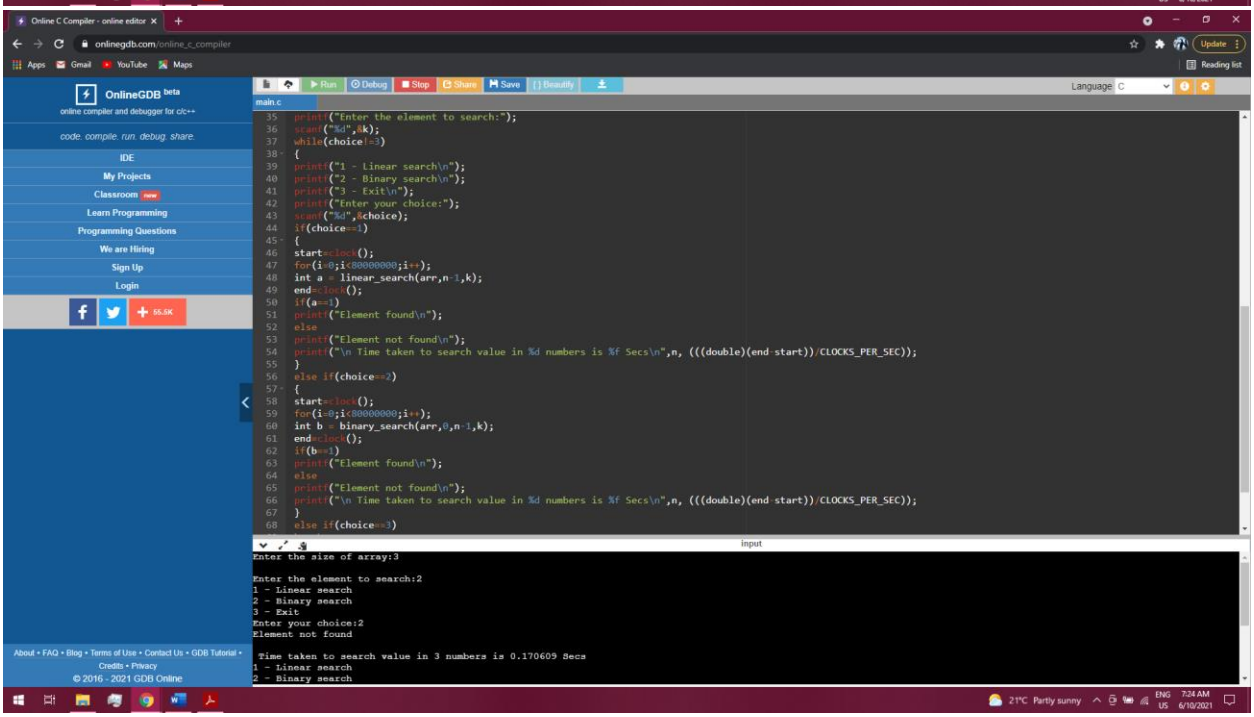
void insertionSort(int arr[], int n)

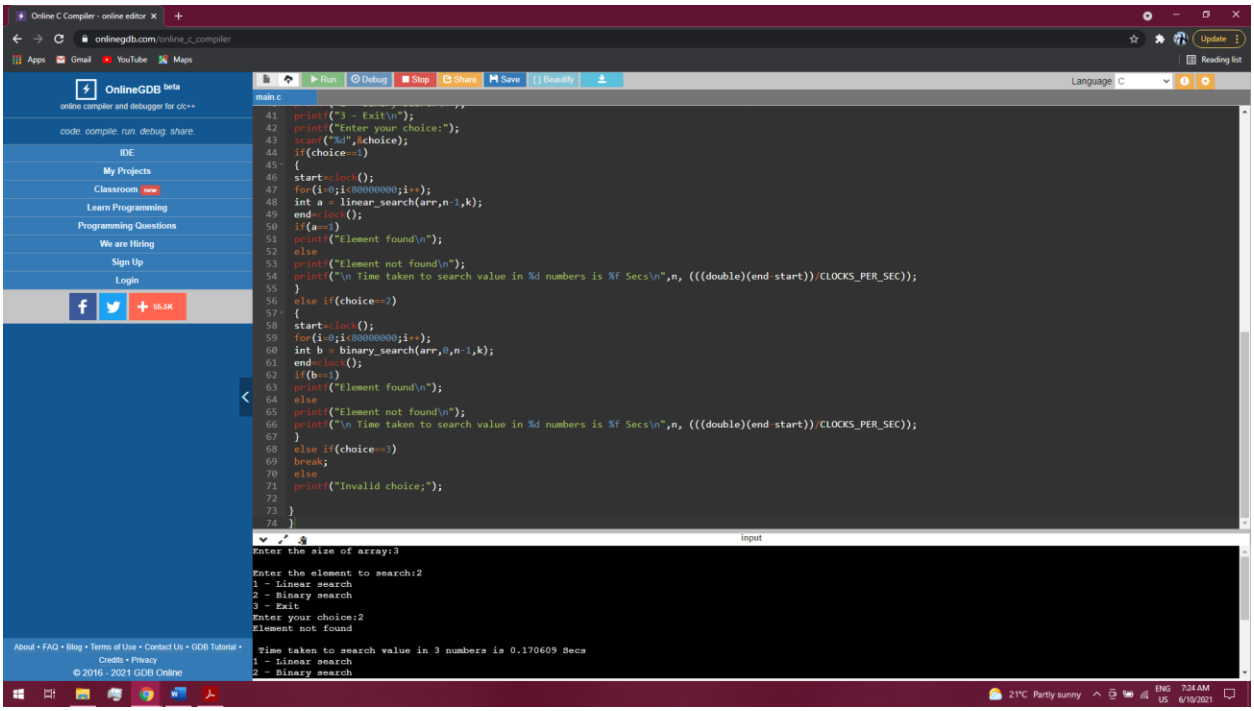
```
{ if (n <= 1)
return;
insertionSort( arr, n-1 );
int last = arr[n-1];
int j = n-2;
while (j >= 0 && arr[j] > last)
{
arr[j+1] = arr[j];
j--;
}
arr[j+1] = last;
}
int main()
{
 int array[50000], search=-1, c, n=0;
 double stime[10];

 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++)
```

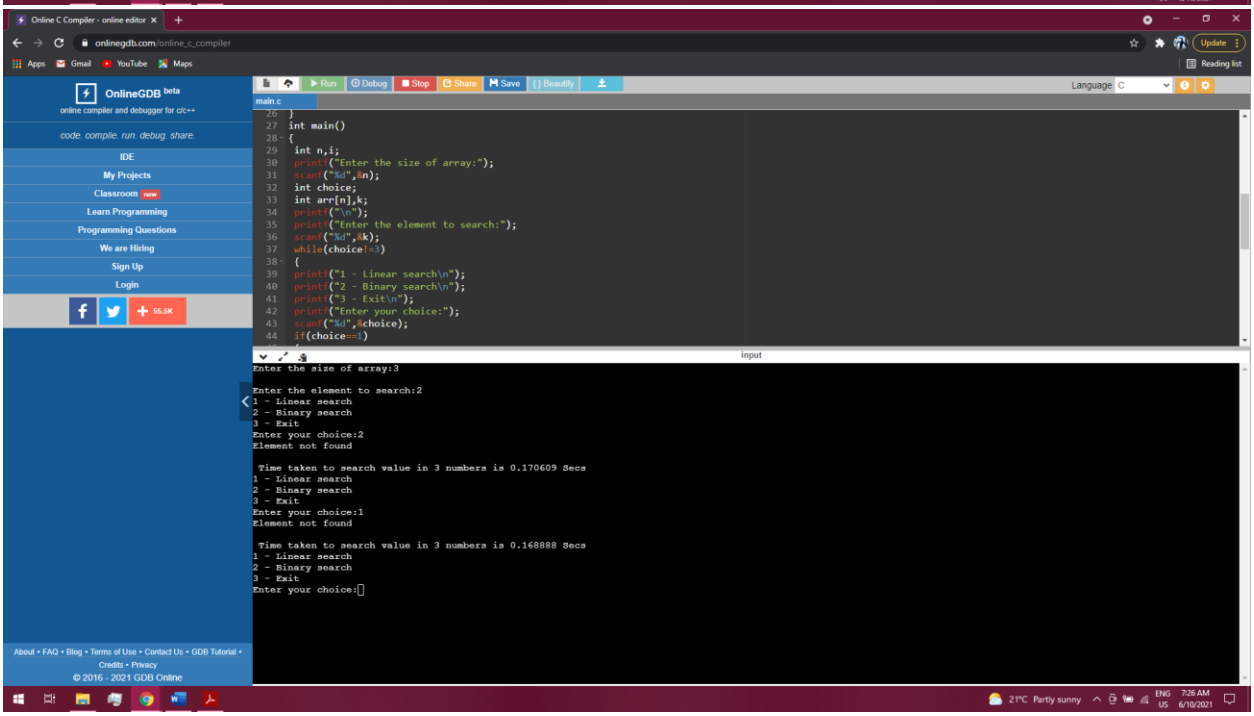
```
 printf("n=%d :%f\n",c*100,stime[c]);

}
```

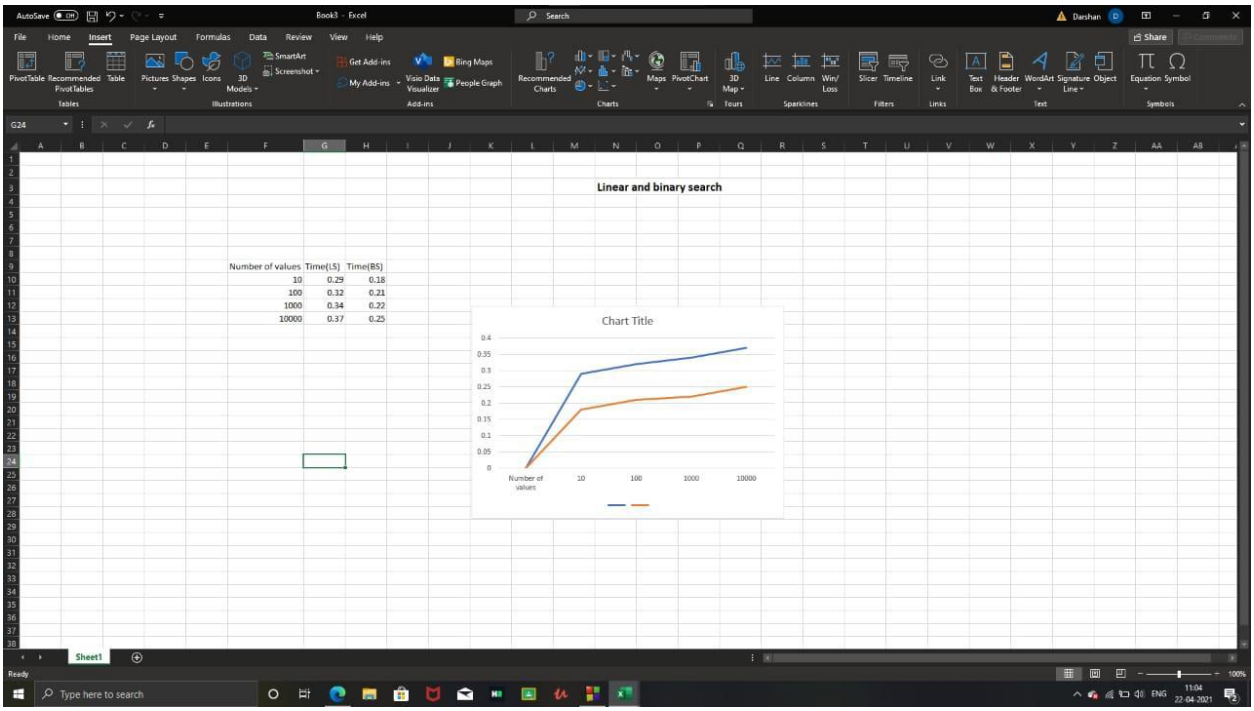
OUTPUT FOR SEARCH






Online C Compiler - online editor
onlinegdb.com/online_c_compiler
OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
We are Hiring
Sign Up
Login
Run
Debug
Stop
Share
Save
{ } Beautify
Language C
main.c
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;");
}
}
input
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
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2021 GDB Online
21°C Partly sunny
ENG US
7:24 AM 6/10/2021


Online C Compiler - online editor
onlinegdb.com/online_c_compiler
OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
We are Hiring
Sign Up
Login
Run
Debug
Stop
Share
Save
{ } Beautify
Language C
main.c
}
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)
input
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
3 - Exit
Enter your choice:1
Element not found
Time taken to search value in 3 numbers is 0.168888 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
21°C Partly sunny
ENG US
7:26 AM 6/10/2021



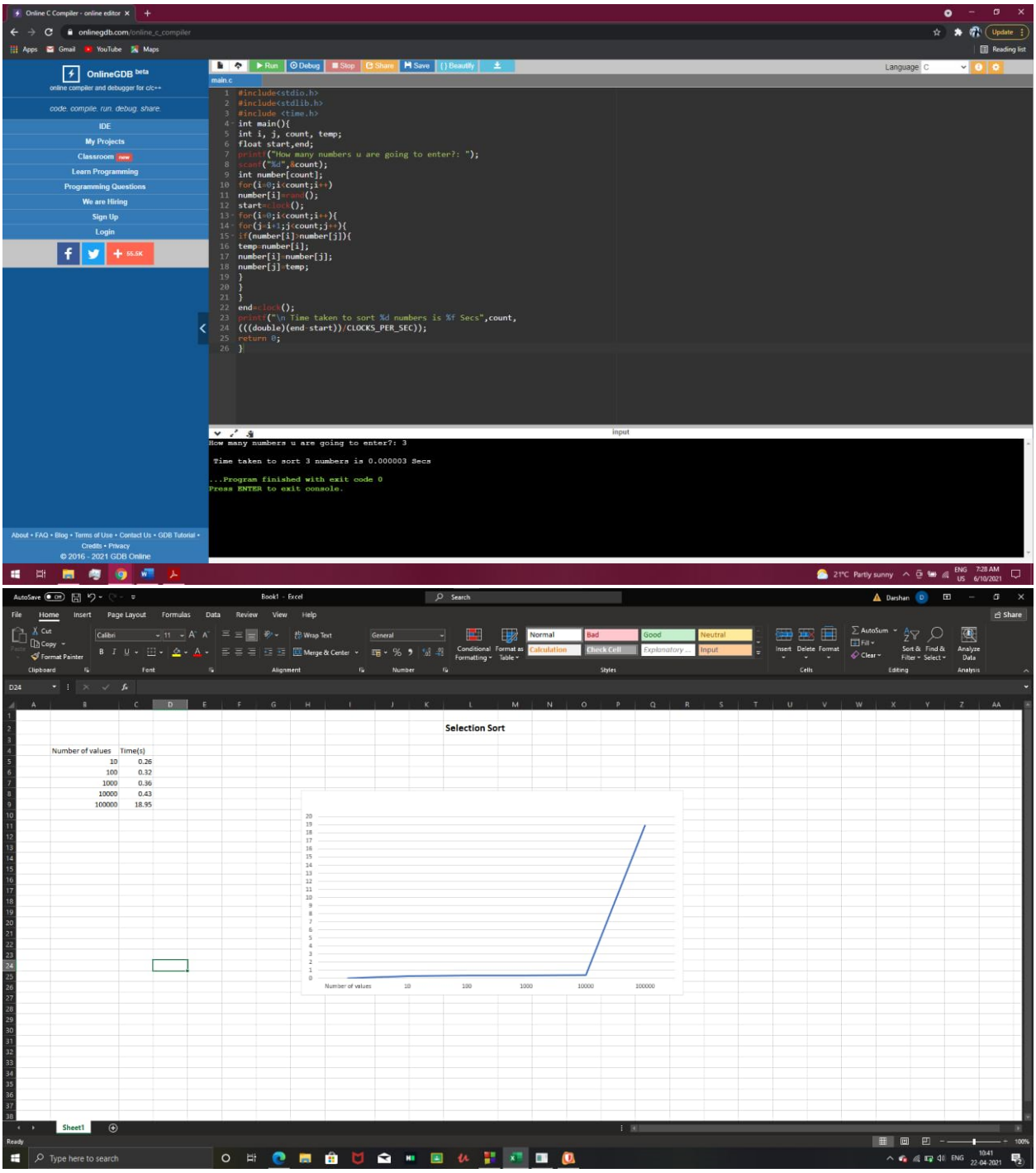
CODE FOR SEARCH

```
#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



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;
}
```