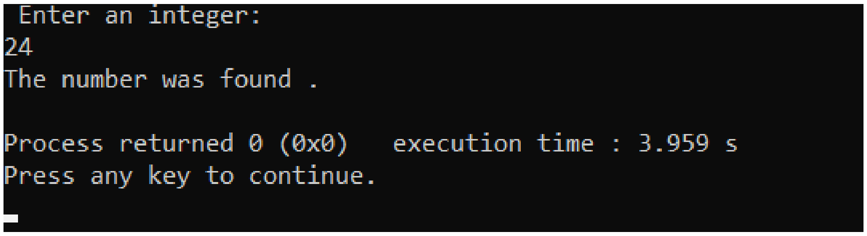
1.linear search input output and find value

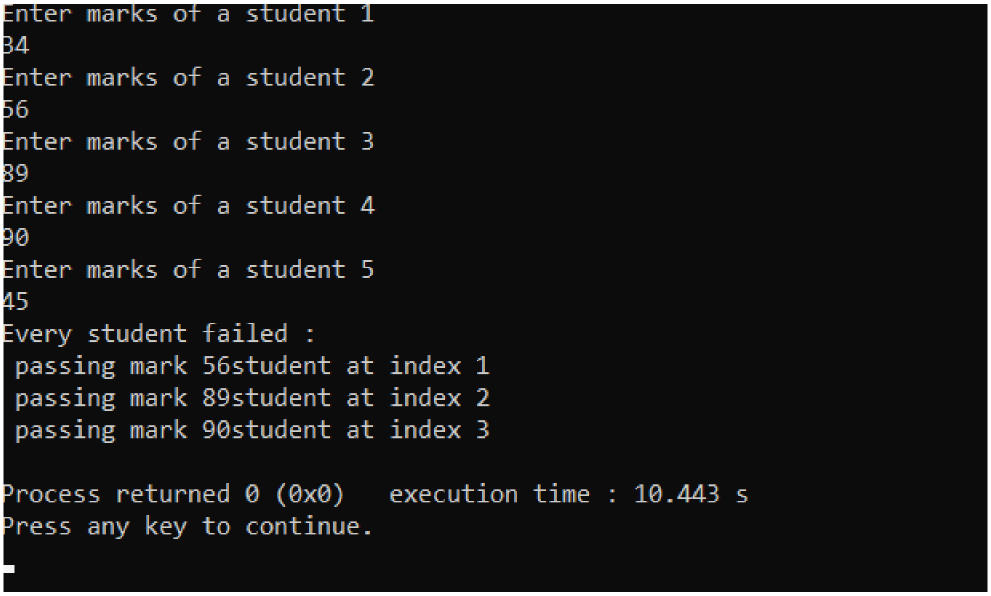
```
#include<iostream>

using namespace std;

int linear_search(int array[],int size ,int value)
{


  for(int i=0;i<size;i++)
  {
  if(array[i]==value)
  {
  return i;
  }


  }
  return-1;
}
int main()
{

 int a[]={14,24,26};

 int value;
 cout<<" Enter an integer: "<<endl;
 cin>>value;
 int index=linear_search(a,24,value);
 if(index>=0)


 {
   cout<<"The number was found ."<<endl;

 }
 else
 {
   cout<<"The number was not found ."<<endl ;
 }



}
```



```
 Enter an integer:
24
The number was found .

Process returned 0 (0x0)   execution time : 3.959 s
Press any key to continue.
```

2.linear search passing mark.

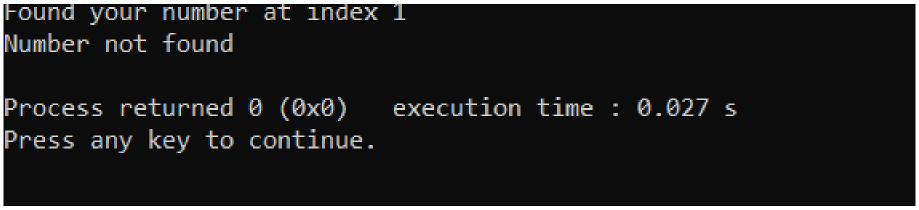
```
#include<iostream>

#define size 5

using namespace std;

void LiSearch(int array[])

{

    int i;

    int count=-1;

    for(int i=0;i<size;i++)

    {

        if (50<=array[i]&&array[i]<=100)

        {


            cout<<" passing mark "<< array[i] <<"student at index " <<i<<endl;

            count=1;

        }

         if(count==(-1))

         {

             cout<<"Every student failed :"<<endl;
        }



    }


}
int main(){
int n[size];
for(int i=0;i<size;i++)
{
    cout<<"Enter marks of a student "<<i+1<< " " <<endl;
    cin>>n[i];
}
LiSearch(n);
}
```



```
Enter marks of a student 1
34
Enter marks of a student 2
56
Enter marks of a student 3
89
Enter marks of a student 4
90
Enter marks of a student 5
45
Every student failed :
 passing mark 56student at index 1
 passing mark 89student at index 2
 passing mark 90student at index 3

Process returned 0 (0x0)   execution time : 10.443 s
Press any key to continue.
```

3.Binary search input and find number.

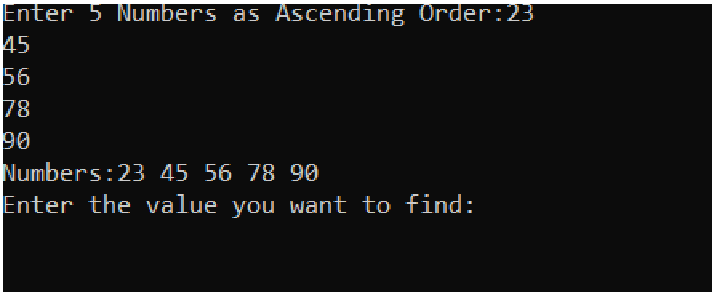
```
#include<iostream>

using namespace std;

int main(){


int a[]={1,2,4,5,67,89,100};

int L= sizeof(a[0]);

int first=0 , last=L-1 , middle, location ;




int number = 2 ;




while(first<=last)

{

   middle=(first+last)/2 ;

   if(a[middle] == number){

      cout<<"Found your number at index " <<middle ;

      break ;

   }


else if( a[middle]>number ){

   last = middle -1 ;

   }
else if( a[middle]<number ) {

    first = middle + 1;

    }

}

cout <<endl;

cout<<"Number not found  " <<endl;


return 0;

}
```



```
Found your number at index 1
Number not found

Process returned 0 (0x0)   execution time : 0.027 s
Press any key to continue.
```

4.Binary search ascending order

```
#include<iostream>

#define length 5

using namespace std;

int AscendingBinarySearch(int Arr[],int key,int low,int high){

  for (int mid = 0; mid < length; mid=(low+high)/2)

  {
    if (Arr[mid]==key)
    {
        return mid;
    }else if (Arr[mid]>key)
    {
        high=mid-1;
    }else if (Arr[mid]<key)
    {
        low=mid+1;
    }

   }

    return -1;


}
void input(int arr[]){
    for (int i = 0; i < length; i++)
    {
        cin>>arr[i];
    }
}
void output(int arr[]){
    cout<<"Numbers:";
    for (int i = 0; i < length; i++)
    {
        cout<<arr[i]<<" ";
    }
    cout<<endl;
}
int main()
{
   int n[length];
   int find;
    cout<<"Enter "<<length<<" Numbers as Ascending Order:";
    input(n);
    output(n);
    cout<<"Enter the value you want to find: ";
    cin>>find;

    int index=AscendingBinarySearch(n,find,0,length-1);
    if (index!=-1)
    {
        cout<<"Value Found at index: "<<index<<endl;
    }else{
        cout<<"Value Not found!"<<endl;
    }
    return 0;
}
```

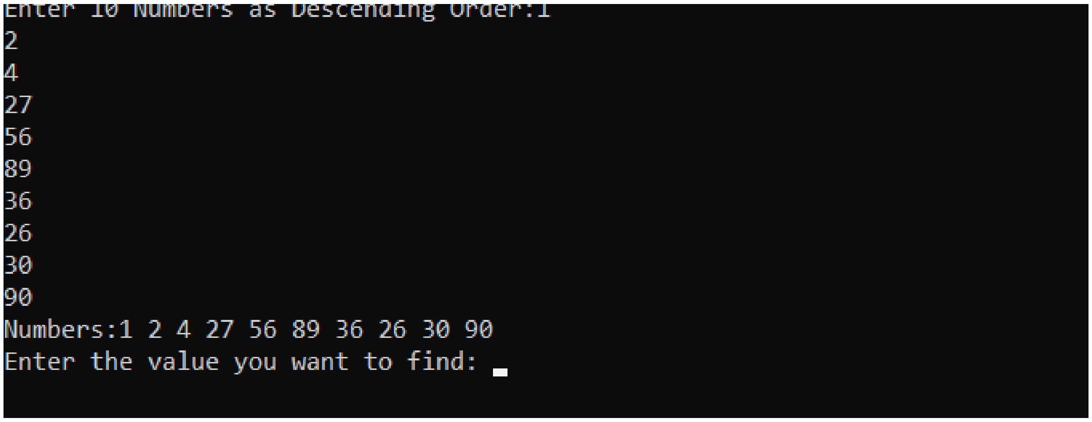


```
Enter 5 Numbers as Ascending Order:23
45
56
78
90
Numbers:23 45 56 78 90
Enter the value you want to find:
```

5.Binary search descending order

```
#include<iostream>

#define siz 10

using namespace std;

int DescendingBinarySearch(int Arr[],int key,int low,int high){

   while (high>=low)

   {

      int mid=(high+low)/2;

      if (Arr[mid]==key)

      {

         return mid;

      }else if (Arr[mid]>key)

      {

         low=mid+1;

      }else{

         high=mid-1;
```

```
        }
    }
    return -1;

}
void input(int arr[]){
    for (int i = 0; i < siz; i++)
    {
        cin>>arr[i];
    }
}
void output(int arr[]){
    cout<<"Numbers:";
    for (int i = 0; i < siz; i++)
    {
        cout<<arr[i]<<" ";
    }
    cout<<endl;
}
int main(){
    int n[siz];
    int find;
    cout<<"Enter "<<siz<<" Numbers as Descending Order:";
    input(n);
    output(n);
    cout<<"Enter the value you want to find: ";
    cin>>find;
    int index=DescendingBinarySearch(n,find,0,siz-1);
    if (index!=-1)
    {
        cout<<"Value Found at index: "<<index<<endl;
    }else{
        cout<<"Value Not found!"<<endl;
    }




}
```

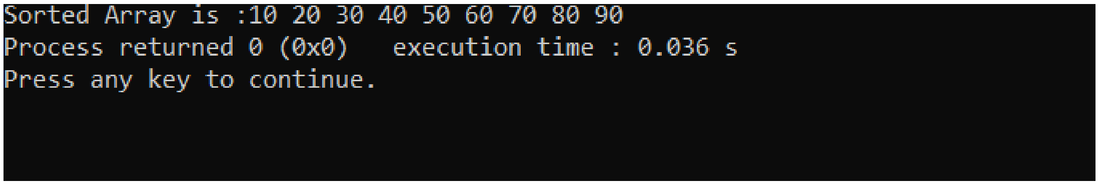


```
Enter 10 Numbers as Descending Order:1
2
4
27
56
89
36
26
30
90
Numbers:1 2 4 27 56 89 36 26 30 90
Enter the value you want to find: 
```

6.Bubble sort in binary search.

```
#include <iostream>
using namespace std;

int main()
{
        int a[]={90,70,20,60,80,30,40,50,10};
        int L=sizeof(a)/sizeof(a[0]);
        int i,j;
        int temp;

        for(i=0; i<L; i++)
        {
                for(j=0; j<i; j++)
                {
                        if(a[i]<a[j])
                        {
                                temp = a[i];
                                a[i] = a[j];
                                a[j] = temp;
                        }
                }
        }



        cout<<"Sorted Array is :";
        for(i=0; i<L; i++)
  cout<<a[i]<<" ";
```

```
        return 0;


}
```

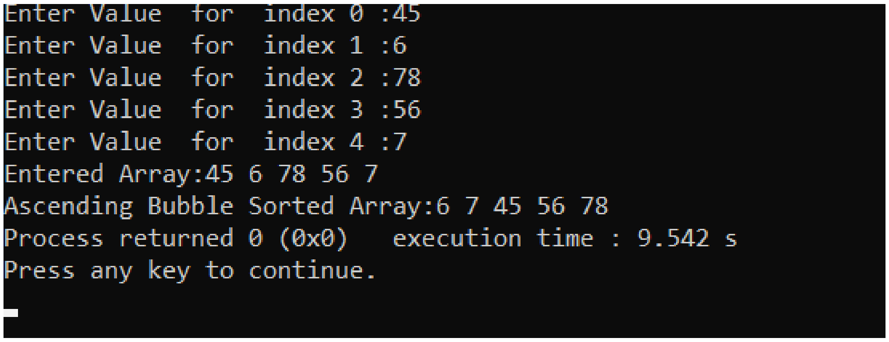


```
Sorted Array is :10 20 30 40 50 60 70 80 90
Process returned 0 (0x0)   execution time : 0.036 s
Press any key to continue.
```

7.Bubble sort ascending order.

```
#include <iostream>
#define siz 5
using namespace std;

void AscendingBubbleSort(int a[])
{
  for (int i = 0; i < siz; i++)
  {
    for (int j = 0; j <= i; j++)
    {
      if (a[i] < a[j])
      {
        int temp;
```

```
                temp = a[i];
                a[i] = a[j];
                a[j] = temp;
            }
        }
    }
}

void input(int arr[])
{

    for (int i = 0; i < siz; i++)
    {
        cout << "Enter Value  for  index " << i << " :";
        cin >> arr[i];
    }
}

void showarray(int a[])
{

    for (int i = 0; i < siz; i++)
    {
        cout << a[i] << " ";
    }
}

int main()
{
    int a[siz];


    input(a);

    cout << "Entered Array:";

    showarray(a);

    cout << endl;

    cout << "Ascending Bubble Sorted Array:";

    (AscendingBubbleSort(a));

    showarray(a);


    return 0;

}
```



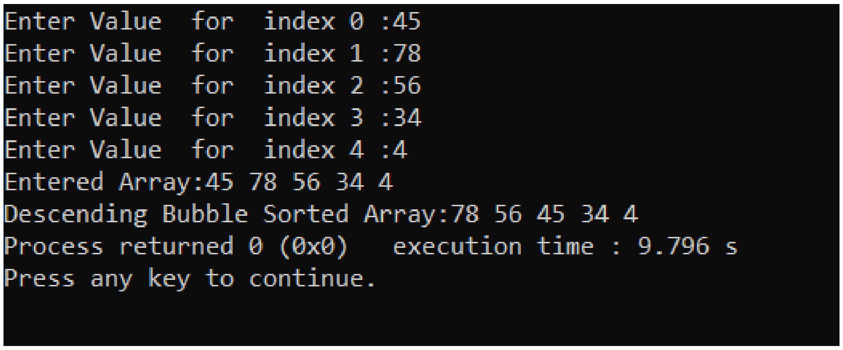
```
Enter Value  for  index 0 :45
Enter Value  for  index 1 :6
Enter Value  for  index 2 :78
Enter Value  for  index 3 :56
Enter Value  for  index 4 :7
Entered Array:45 6 78 56 7
Ascending Bubble Sorted Array:6 7 45 56 78
Process returned 0 (0x0)   execution time : 9.542 s
Press any key to continue.
```

8.Descending bubble sort

```
#include <iostream>

#define siz 5

using namespace std;
```

```
void DescendingBubbleSort(int a[])
{
    for (int i = 0; i < siz; i++)
    {
        for (int j = 0; j <= i; j++)
        {
            if (a[i] > a[j])
            {
                int temp;
                temp = a[i];
                a[i] = a[j];
                a[j] = temp;
            }
        }
    }
}

void input(int arr[])
{

    for (int i = 0; i < siz; i++)
    {
        cout << "Enter Value  for  index " << i << " :";
        cin >> arr[i];
    }
}

void showarray(int a[])
```

```
{

    for (int i = 0; i < siz; i++)
    {
        cout << a[i] << " ";
    }
}

int main()
{
    int a[siz];

    input(a);
    cout << "Entered Array:";
    showarray(a);
    cout << endl;
    cout << "Descending Bubble Sorted Array:";
    (DescendingBubbleSort(a));
    showarray(a);



    return 0;
}
```



```
Enter Value  for  index 0 :45
Enter Value  for  index 1 :78
Enter Value  for  index 2 :56
Enter Value  for  index 3 :34
Enter Value  for  index 4 :4
Entered Array:45 78 56 34 4
Descending Bubble Sorted Array:78 56 45 34 4
Process returned 0 (0x0)   execution time : 9.796 s
Press any key to continue.
```