Instructions: Please read carefully

- Please rename this file as only your ID number **(e.g. 18-*****-1.doc or 18-*****-1.pdf).**
- Submit the file before **11:59pm on 25/11/2020** in the Portal Lab Performance section labeled **Lab task 8.** **If you cannot complete the full task, do not worry. Just upload what you have completed.**

1. Write a code to implement Bubble Sort for the following list

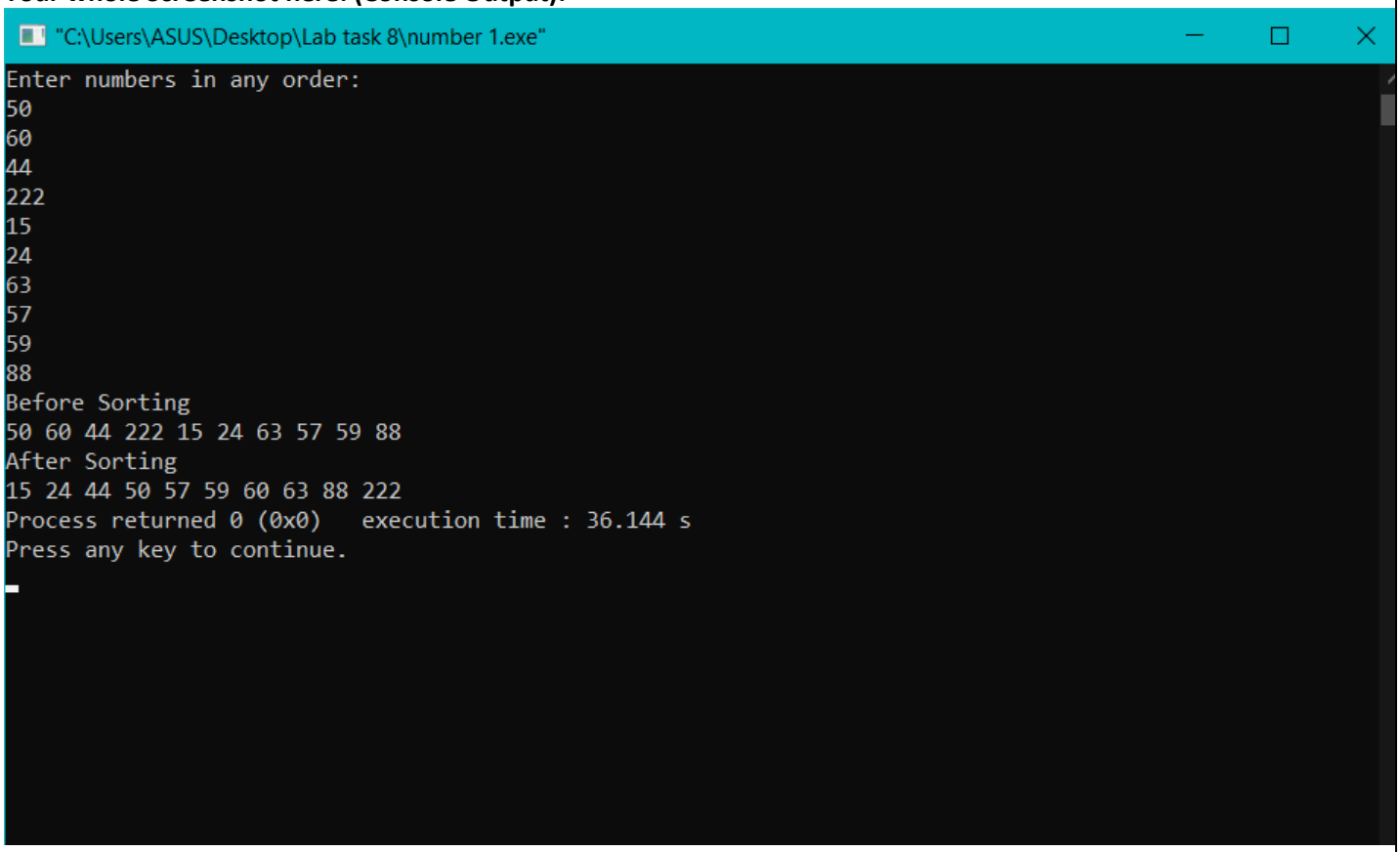
50	60	44	222	15	24	63	57	59	88

Your code here:

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

void bubbleSort(int a[]) //a is a pointer to my do my array where my array is a address

{

 for (int i = 0; i < 10; i++) //outer loop where size of the array is 10
  {
    for (int j = 0; j < (10 - i - 1); j++) //here i have used a condition(j =0  to  j< n-i-1) for the inner loop
    {
      if (a[j] > a[j + 1])
      {
        int temp = a[j]; //swap(a[j] , a[j+1]
        a[j] = a[j + 1];
        a[j + 1] = temp;

        }
      }
    }
  }




int main()
{
 int myarray[10];
 int size;

 cout << "Enter numbers in any order: " << endl;

 for (int i = 0; i < 10; i++)
  {
    cin >> myarray[i];
 }


 cout << "Before Sorting" << endl;
 for (int i = 0; i < 10; i++)
  {
  cout << myarray[i] << " ";
 }
```

```
 bubbleSort(myarray); // sorting

 cout << endl << "After Sorting" << endl;

 for (int i = 0; i < 10; i++)
  {
  cout << myarray[i] << " ";
 }

 return 0;
}
```

Your whole Screenshot here: (Console Output):



2. Write a code to implement Selection Sort for the following list

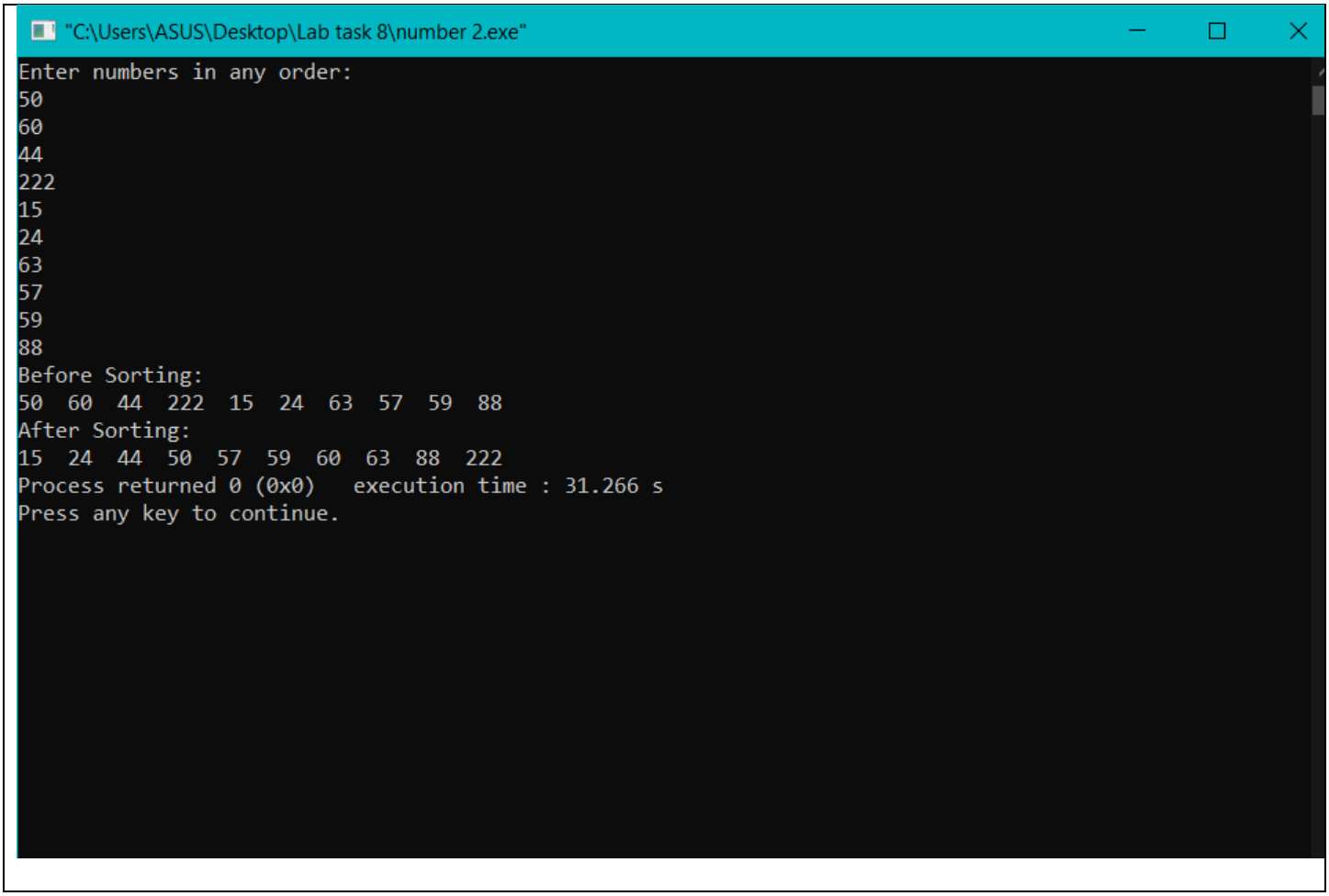
50	60	44	222	15	24	63	57	59	88

Your code here:

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

void selectionSort(int arr[])
{
 for (int i = 0; i < 9; i++) //outer loop will run 1 less than the array (10-1=9),used condition is (i = 0; i<n-1; i++) set current element as minimum
   {
```

```
   int min = i;  // set current element as minimum, min = i

   for (int j = i + 1; j < 10; j++) //inner loop ( j = i+1; j<n; j++)
      {
         if (arr[j] < arr[min])
         {
            min = j;
    }
   }

   if (min != i) //swap the minimum element & current element
   {
     int temp = arr[min];  //swap arr[min] and arr[i]
     arr[min] = arr[i];
     arr[i] = temp;
   }
  }
}

int main()
{

  int myarr[10];

  cout << "Enter numbers in any order: " << endl;


  for (int i = 0; i < 10; i++)
   {
      cin >> myarr[i];
  }

  cout << "Before Sorting: " << endl;

  for (int i = 0; i < 10; i++)
   {
      cout << myarr[i] << "  ";
  }

  cout << endl;

  selectionSort(myarr); // sorting actually happening

  cout << "After Sorting: " << endl;

  for (int i = 0; i < 10; i++)
   {
      cout << myarr[i] << "  ";
  }
  return 0;
}
```

Your whole Screenshot here: (Console Output):



3. Write a code to implement Insertion Sort for the following list

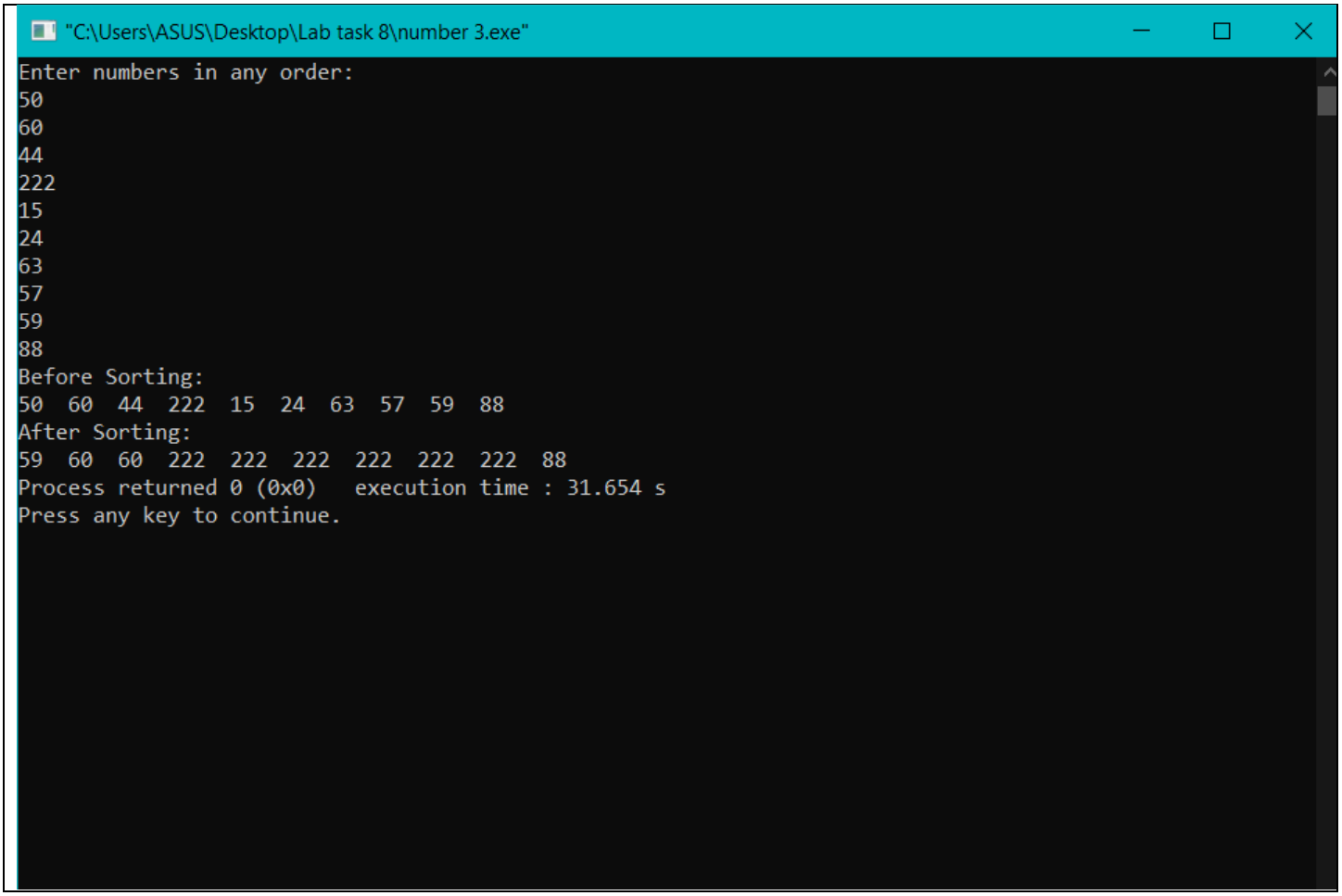
50	60	44	222	15	24	63	57	59	88

Your code here:

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

void insertionSort(int arr[])
{
   int key=0; // declare variables
   int j=0;

   for(int i=1; i<10; i++) //outer loop (loop : 1 to n-1
   {
      key = arr[i]; //picking the element
      j = i-1;
      while(j>=0 && arr[j]>key) //inner loop
      {
         arr[j+1] = arr[j];
         j=j-i;
      }
      arr[j+1] = key;
   }
}
```

```
int main()
{

  int myarray[10];

  cout << "Enter numbers in any order: " << endl;


  for (int i = 0; i < 10; i++)
    {
       cin >> myarray[i];
  }

  cout << "Before Sorting: " << endl;

  for (int i = 0; i < 10; i++)
    {
       cout << myarray[i] << "  ";
  }

  cout << endl;

  insertionSort(myarray); // sorting actually happening

  cout << "After Sorting: " << endl;

  for (int i = 0; i < 10; i++)
    {
       cout << myarray[i] << "  ";
  }
  return 0;
}
```

Your whole Screenshot here: (Console Output):



4. Write a code to implement Linear search to find a particular value in a linear array to find 63 in the following list

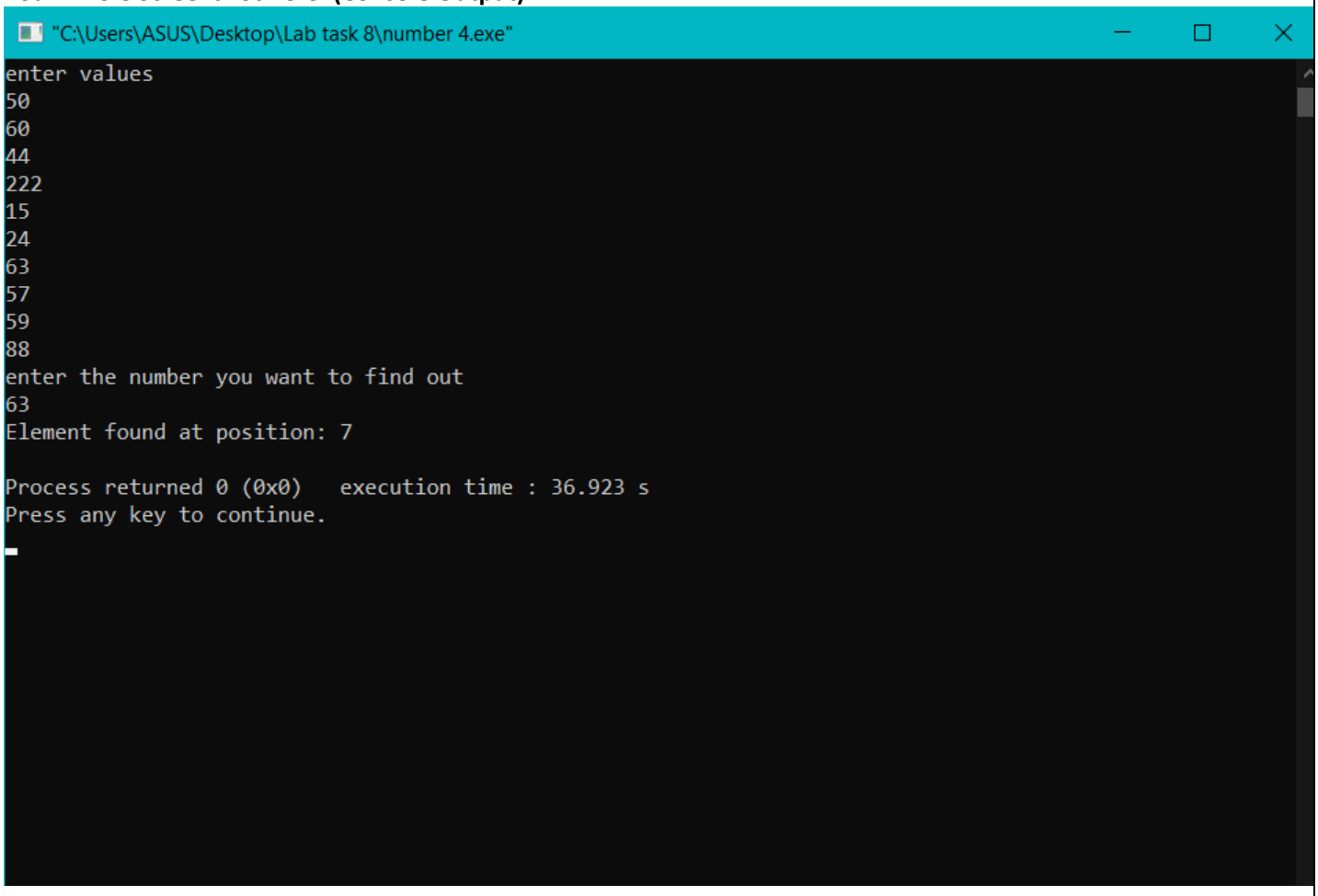
50	60	44	222	15	24	63	57	59	88

Your code here:

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

void linearSearch(int a[], int n)

{
 int temp = -1;

 for (int i = 0; i < 10; i++) //loop to continue
  {
    if (a[i] == n)
    {
       cout << "Element found at position: " << i + 1 << endl;
   temp = 0;
   break;// to exit once match found
  }
 }

 if (temp == -1) //checking if temp = -1
```

```
   {
    cout << "No Element Found" << endl;
  }

}

int main()

{
  int arr[10];
  cout << "enter values" << endl;
  for (int i = 0; i < 10; i++)
    {
    cin >> arr[i];
  }
  cout << "enter the number you want to find out" << endl;
  int num;
  cin >> num;

  linearSearch(arr, num);

  return 0;
}
```

Your whole Screenshot here: (Console Output):



5. Write a Program for Binary Search Implementation to find 63 in the following list

50	60	44	222	15	24	63	57	59	88

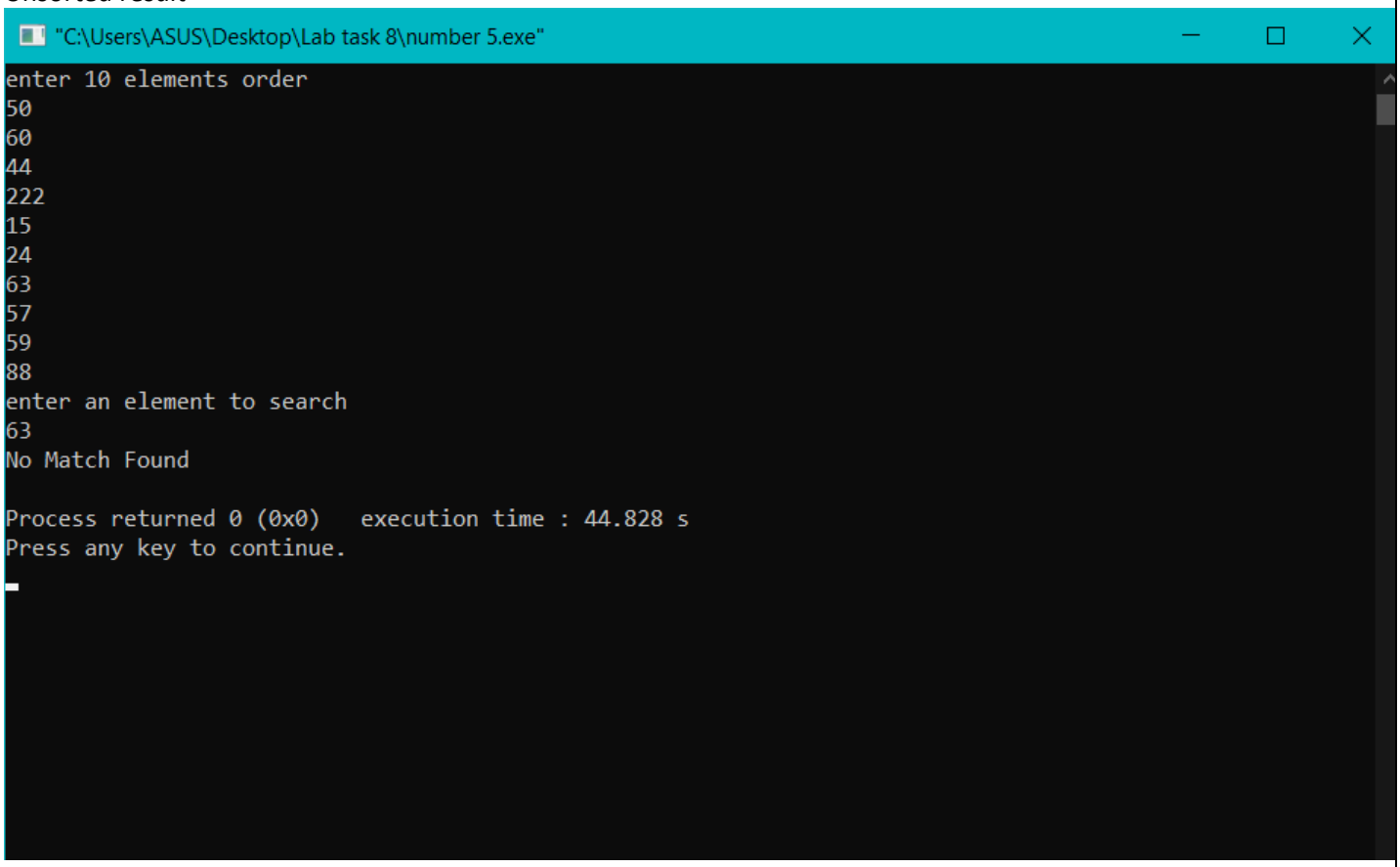
Your code here:

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

int binarySearch(int arr[], int left, int right, int x) //take input: array, left,right & x

{
   while (left <= right) //Start Loop {while(left<=right)}
{
   int mid = left + (right - left) / 2;

   if (arr[mid] == x) //checking if arr[mid] is ==x or we want to search
      {
    return mid; //if yes
   }

   else if (arr[mid] < x) //if no match
      {
    left = mid + 1; //reducing the interval
   }


   else
      {
    right = mid - 1;
   }
 }

 return -1;
}

int main()

{
 int myarr[10];
 int num;
 int output;

 cout << "enter 10 elements order" << endl; //Need array has to be in a sorted a way (ASCENDING order)
 for (int i = 0; i < 10; i++)
  {
  cin >> myarr[i];
 }
 cout << "enter an element to search" << endl;
 cin >> num;

 output = binarySearch(myarr, 0, 9, num);
```

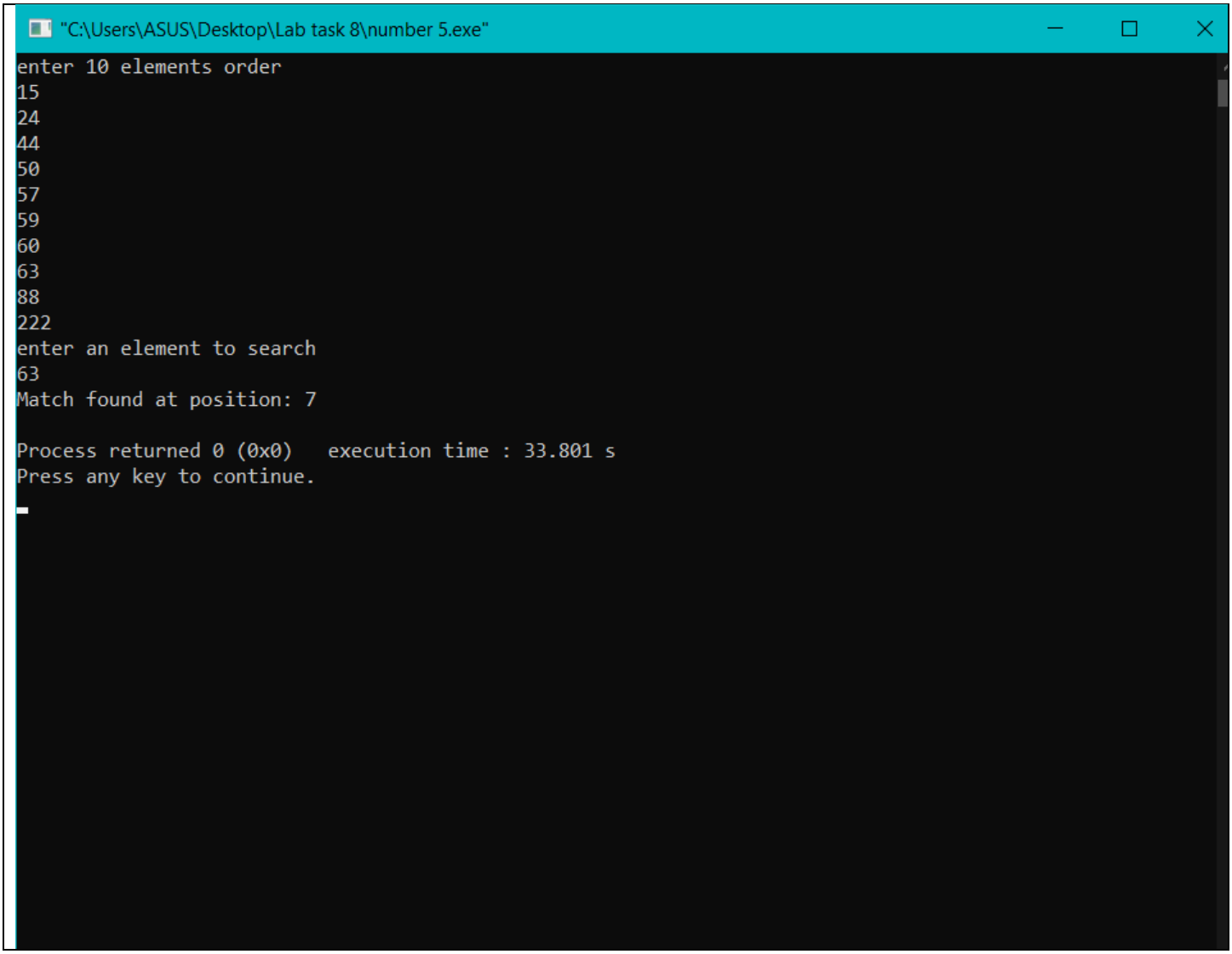
```
  if (output == -1)

   {
   cout << "No Match Found" << endl;
  }
  else
   {
   cout << "Match found at position: " << output << endl;
  }

  return 0;
}
```

Your whole Screenshot here: (Console Output):

Unsorted result



Sorted result


"C:\Users\ASUS\Desktop\Lab task 8\number 5.exe"
enter 10 elements order
15
24
44
50
57
59
60
63
88
222
enter an element to search
63
Match found at position: 7
Process returned 0 (0x0)   execution time : 33.801 s
Press any key to continue.