	Department of Computer Science and Engineering Faculty of Engineering, South Eastern University of Sri Lanka		
	Subject	**CS53003: Data Structure and Algorithms**	
	Batch	**E18**	**Semester**	**5**

Lab no and title : Lab 05: Searching
Name : G.W.P.R.R. Wijesinghe
Reg No : SEU-IS-18-EG-013
Submission Date : 18-01-2022

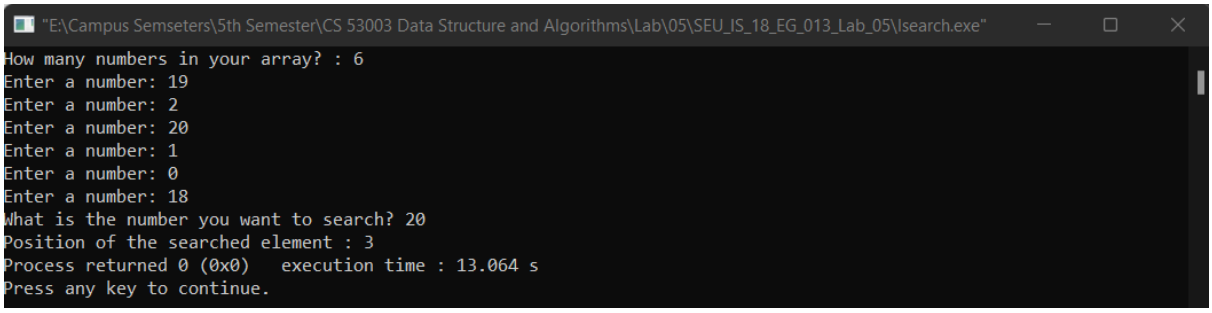
---

01. Given an integer and an array of integers, write a function named linear_search that returns the number of comparisons performed doing a linear search. The function should take 3 arguments:
    - The number searched for
    - The array of integers
    - The number of elements in the array

    If the number searched for is not in the array then the function should return -1.

Start here | lsearch.cpp | bsearch.cpp

```cpp
#include<iostream>

using namespace std;

int linear_search(int num, int arr[], int arrSize){
    for (int i=0; i<arrSize; i++){
        if (arr[i]==num){
            return i+1;
        }
    }
    return -1;
}

int main(){
    int aSize, searchNum;

    cout << "How many numbers in your array? : ";
    cin >>  aSize;
    int A[aSize]={};
    for (int j=0; j<aSize; j++){
        cout << "Enter a number: ";
        cin >> A[j];
    }

    cout << "What is the number you want to search? ";
    cin >> searchNum;
    cout << "Position of the searched element : "<<linear_search(searchNum,A,aSize);
}
```



02. Given an integer and a sorted array of integers, write a function named binary_search that prints the number of comparisons performed doing a binary search. The function should take 3 arguments:
    - The number searched for
    - The array of integers
    - The number of elements in the array

    If the number searched for is not in the array then the function should return -1.

Start here | lsearch.cpp | bsearch.cpp

```cpp
#include<iostream>

using namespace std;

int binarySearch(int num, int arr[], int arrSize){
    int first = 0;
    int last = arrSize-1;
    int totalCount = 0;

    while (first <= last){
        int mid = (first+last)/2;
        totalCount = totalCount+1;
        if (num == arr[mid]){
            return totalCount;
        }
        else if (num < arr[mid]){
            last = mid-1;
        }
        else {
            first = mid+1;
        }

    }
    return -1;
}

int main(){
    int aSize, searchNum;

    cout << "How many numbers in your array? : ";
    cin >>  aSize;
    int A[aSize]={};
    for (int j=0; j<aSize; j++){
        cout << "Enter a number: ";
        cin >> A[j];
    }

    cout << "What is the number you want to search? ";
    cin >> searchNum;
    cout << "Position of the searched element : "<<binarySearch(searchNum,A,aSize);
}
```


"E:\Campus Semseters\5th Semester\CS 53003 Data Structure and Algorithms\Lab\05\SEU_IS_18_EG_013_Lab_05\bsearch.exe"
How many numbers in your array? : 7
Enter a number: 0
Enter a number: 1
Enter a number: 2
Enter a number: 18
Enter a number: 19
Enter a number: 20
Enter a number: 25
What is the number you want to search? 20
Position of the searched element : 2
Process returned 0 (0x0)   execution time : 22.175 s
Press any key to continue.

***