

	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 x lsearch.cpp x bsearch.cpp x
1  #include<iostream>
2
3  using namespace std;
4
5  int linear_search(int num, int arr[], int arrSize){
6      for (int i=0; i<arrSize; i++){
7          if (arr[i]==num){
8              return i+1;
9          }
10     }
11     return -1;
12 }
13
14 int main(){
15     int aSize, searchNum;
16
17     cout << "How many numbers in your array? : ";
18     cin >> aSize;
19     int A[aSize]={};
20     for (int j=0; j<aSize; j++){
21         cout << "Enter a number: ";
22         cin >> A[j];
23     }
24
25     cout << "What is the number you want to search? ";
26     cin >> searchNum;
27     cout << "Position of the searched element : "<<linear_search(searchNum,A,aSize);
28 }
29

```

```
"E:\Campus Semesters\5th Semester\CS 53003 Data Structure and Algorithms\Lab\05\SEU_JS_18_EG_013_Lab_05\search.exe"
How many numbers in your array? : 6
Enter a number: 19
Enter a number: 2
Enter a number: 20
Enter a number: 1
Enter a number: 0
Enter a number: 18
What is the number you want to search? 20
Position of the searched element : 3
Process returned 0 (0x0)   execution time : 13.064 s
Press any key to continue.
```

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 x |search.cpp x |bsearch.cpp x
1 | #include<iostream>
2 |
3 | using namespace std;
4 |
5 | int binarySearch(int num, int arr[], int arrSize){
6 |     int first = 0;
7 |     int last = arrSize-1;
8 |     int totalCount = 0;
9 |
10 |    while (first <= last){
11 |        int mid = (first+last)/2;
12 |        totalCount = totalCount+1;
13 |        if (num == arr[mid]){
14 |            return totalCount;
15 |        }
16 |        else if (num < arr[mid]){
17 |            last = mid-1;
18 |        }
19 |        else {
20 |            first = mid+1;
21 |        }
22 |    }
23 |    return -1;
24 | }
25 |
26 | int main(){
27 |     int aSize, searchNum;
28 |
29 |     cout << "How many numbers in your array? : ";
30 |     cin >> aSize;
31 |     int A[aSize]={};
32 |     for (int j=0; j<aSize; j++){
33 |         cout << "Enter a number: ";
34 |         cin >> A[j];
35 |     }
36 |
37 |     cout << "What is the number you want to search? ";
38 |     cin >> searchNum;
39 |     cout << "Position of the searched element : "<<binarySearch(searchNum,A,aSize);
40 |
41 | }
42 |
```

```
"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.
```

3. Write a C++ program to implement binary search using recursion

```
Start here x RecusBsort.cpp x EX04.cpp x
1  #include<iostream>
2
3  using namespace std;
4
5  int binarySearch(int arr[],int first, int last, int num){
6      int mid = (first+last)/2;
7      if (num == arr[mid]){
8          return mid+1;
9      }
10     else if (num < arr[mid]){
11         binarySearch(arr,first,mid-1,num);
12     }
13     else {
14         binarySearch(arr,first+1,last,num);
15     }
16 }
17
18 int main(){
19     int aSize, searchNum;
20     int start = 0;
21
22     cout << "How many numbers in your array? : ";
23     cin >> aSize;
24     int A[aSize]={};
25     for (int j=0; j<aSize; j++){
26         cout << "Enter a number: ";
27         cin >> A[j];
28     }
29
30     cout << "What is the number you want to search? ";
31     cin >> searchNum;
32     cout << "Position of the searched element : "<<binarySearch(A,start,aSize-1,searchNum);
33 }
34
35
```

```
"E:\Campus Semseters\5th Semester\CS 53003 Data Structure and Algorithms\Lab\05\Final New ..."
How many numbers in your array? : 8
Enter a number: 15
Enter a number: 36
Enter a number: 24
Enter a number: 78
Enter a number: 95
Enter a number: 62
Enter a number: 34
Enter a number: 16
What is the number you want to search? 78
Position of the searched element : 4
Process returned 0 (0x0)   execution time : 18.031 s
Press any key to continue.
```

4. Write a C++ program to find out the number of occurrences of a number in a sorted array using binary search.

```

Start here x RecusBsort.cpp x EX04.cpp x
1  #include<iostream>
2
3  using namespace std;
4
5  int binary_search(int arr[] , int first , int last , int key){
6      if(first > last)
7          return -1;
8      else{
9          int mid = (first + last)/2;
10         if(key == arr[mid])
11             return mid;
12         else if (key<arr[mid])
13             return binary_search(arr , first , mid-1 , key);
14         else
15             return binary_search(arr , mid+1 , last , key);
16     }
17 }
18
19 int count_occur (int arr[] , int n , int key){
20     int result = binary_search(arr , 0 , n-1 , key);
21     if (result == -1)
22         return 0;
23
24     int count = 1;
25     int i = result-1;
26     while (i>=0 && arr[i] == key)
27         count++ , i--;
28
29     int j = result + 1;
30     while (j<n && arr[j] == key)
31         count++ , j++;
32     return count;
33 }
34
35 int main(){
36     int elements , key , n ;
37     int start=0 ;
38
39     cout << "No of elements in array : ";
40     cin >> elements ;
41
42     int arr[elements]={};
43     for (int i=0; i<elements; i++){
44         cout << "Enter the elements : ";
45         cin >> arr[i];
46     }
47
48     cout<< "searching element : ";
49     cin>>key;
50
51     int X = sizeof(arr)/sizeof(arr[0]);
52     int results = count_occur (arr , X , key);
53
54     cout << "Number of occurrences of the number : "<< results;
55 }
56

```

```
"E:\Campus Semseters\5th Semester\CS 53003 Data Structure and Algorithms\La...
No of elements in array : 8
Enter the elements : 1
Enter the elements : 1
Enter the elements : 1
Enter the elements : 2
Enter the elements : 6
Enter the elements : 9
Enter the elements : 8
Enter the elements : 7
searching element : 1
Number of occurrences of the number : 3
Process returned 0 (0x0)    execution time : 9.670 s
Press any key to continue.
```
