



Mohammad Ali Jinnah University

Chartered by Government of Sindh - Recognized by HEC

Lab 2

Name: Muhamad Fahad

Id: FA19-BSSE-0014

Subject: Data Structures and Algorithms Lab (CS 2511)

Lab Title: Binary Search

Section: AM

Teacher: MUHAMMAD MUBASHIR KHAN

Date: Tuesday, October 20, 2020

Lab Tasks

1) Implement Binary Search to find the given key?

Code:

```
public class Ex1 {  
    public static void main(String[] args) {  
  
        int sizeOfArray = 10;  
  
        int arr[] = new int[sizeOfArray];  
  
        for (int i = 0; i < sizeOfArray; i++) {  
            arr[i] = i+1;  
        }  
  
        int LP = 0,  
            RP = sizeOfArray-1,  
            mid,  
            key = 9,  
            hit = 0;  
  
        boolean flag = true;  
  
        while (LP<=RP){  
            mid = (LP+RP)/2;  
  
            if(key == arr[mid]){  
                System.out.println("Key found at: "+mid+" index in the hit: "+hit);  
                flag = false;  
                break;  
            }  
            else if(arr[mid]<key){  
                LP = mid+1;  
            }  
            else if(arr[mid]>key){  
                RP = mid-1;  
            }  
            hit++;  
        }  
        if (flag){  
            System.out.println("The key not found in the array in the hit: "+hit);  
        }  
    }  
}
```

Data Structures and Algorithms Lab

Output:

```
"C:\Program Files\Java\jdk-13.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA
1) Implement Binary Search to find the given key.
The array is: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
The key is: 9
Key found at: 8 index in the hit: 2

Process finished with exit code 0
```

2) Implement Binary Search to find the number of occurrences of the given key.

Code:

```
public class Ex2 {
    public static void main(String[] args) {

        int arr[] = {3,5,5,9,9,9,10,11};

        int LP = 0,
            RP = arr.length-1,
            mid,
            key = 9,
            count = 0,
            temp;

        boolean checkBefore=false,
            check = false,
            flag = false;

        while (LP<=RP){
            mid = (LP+RP)/2;
            if(key == arr[mid]){
                temp = mid;
                do {
                    if (!checkBefore && temp != 0) { //first we will go toward the back
                        temp--; // for example the mid index is 3 and value is 9 so we will check the next previous index
                        check=true; // means the index change and we have to check
                    }
                    else if(checkBefore && temp != (arr.length-1)){ //if we will not found it on previous index so known
                        we have check next index
                        temp++; // for example the mid index is 3 and value is 9 so we will check the next index
                        check=true; // means the index change and we have to check
                    }
                } else
                    check = false; // when both side are check

                if(check && arr[temp] == key){ // it will work on both side forward and backward indexes
                    count++; // mean the Occurrences found
                }
                else if(!checkBefore && check) { // if the same value not found so we have to move forward
                    checkBefore = !checkBefore; // so we have to move forward(on the next index from mid)
                    temp = mid; // again set mid point s you we will move forward
                }
            } while(check);
            count++;
            break;
        }

        if(arr[mid]<key){
            LP = mid+1;
        }
    }
}
```

Data Structures and Algorithms Lab

```
    }
    else if(arr[mid]>key){
        RP = mid-1;
    }
}

if (flag){
    System.out.println("The key not found in the array");
}
else {
    System.out.println("The Occurrences of the given key: " + key + " is: " + count + " times");
}

}
}
```

Output:

```
"C:\Program Files\Java\jdk-13.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA
2) Implement Binary Search to find the number of occurrences of the given key.
The array is: [3, 5, 5, 9, 9, 9, 10, 11]
The key is: 9
The Occurrences of the given key: 9 is: 3 times

Process finished with exit code 0
```