Data Structures and Algorithms

Lab: Sorting and Searching

1. Search an element in an array and print the index position.

Input:

```java
import java.util.*;
public class Searching {
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);

        System.out.println("Enter the number of elements in the array");
        int tot=sc.nextInt();
        int[] arr= new int[tot];
        for(int i=0;i<tot;i++)
        {
            System.out.println("Enter the "+(i+1)+" element");
            arr[i]=sc.nextInt();
        }
        int flag=0;
        System.out.println("Enter the element you would like to search for");
        int ele=sc.nextInt();
        System.out.println("Using Linear Search");
        for(int i=0;i<arr.length;i++)
            if(arr[i]==ele)
                flag= i+1;
        if (flag>0)
            System.out.println("Element is at "+flag+" pos");
        else
            System.out.println("Element not found");

        System.out.println("Using Binary Search");
        System.out.println("After array sorted:");
        int n = arr.length;
```

```java
        for (int i = 0; i < n - 1; i++)
            for (int j = 0; j < n - i - 1; j++)
                if (arr[j] > arr[j + 1])
                {
                    int temp = arr[j];
                    arr[j] = arr[j + 1];
                    arr[j + 1] = temp;
                }
        for (int i = 0; i < arr.length; i++)
            System.out.print(arr[i]+" ");
        System.out.println("");
        int low=0, high=n-1;
        while (low <= high)
        {
            int mid = low + (high - low) / 2;

            if (arr[mid] == ele)
            { flag=mid+1;
                break;
            }
            if (arr[mid] < ele)
                low = mid + 1;
            else
                high = mid - 1;
        }
        if (flag>0)
            System.out.println("Element is at "+flag+" pos");
        else
            System.out.println("Element not found");

    }
}
```

Output:

```
Enter the number of elements in the array
10
Enter the 1 element
12
Enter the 2 element
65
Enter the 3 element
232
Enter the 4 element
11
Enter the 5 element
76
Enter the 6 element
25
Enter the 7 element
96
Enter the 8 element
44
Enter the 9 element
12
Enter the 10 element
343
Enter the element you would like to search for
76
Using Linear Search
Element is at 5 pos
Using Binary Search
After array sorted:
11 12 12 25 44 65 76 96 232 343
Element is at 7 pos
```

2. Find the duplicate element frequency and change the value as 0 when more than one same element present in an array

Input:

```java
import java.util.*;
class freq {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of elements in the array");
        int tot = sc.nextInt();
        int[] arr = new int[tot];
        int[] narr = new int[tot];
        int flagnum = 0;
        int flag = 1;
        for (int i = 0; i < tot; i++) {
            System.out.println("Enter the " + (i + 1) + " element");
            arr[i] = sc.nextInt();
            narr[i] = arr[i];
            for (int j = 0; j < i; j++) {
                if (arr[i] == arr[j]) {
                    flagnum = arr[j];
                    narr[i] = 0;
                    flag++;
                }
            }
        }
        System.out.println("The entered array is:");
        for (int i = 0; i < tot; i++)
            System.out.print(arr[i] + " ");
        System.out.println("\n The new array is:");
        for (int i = 0; i < tot; i++)
            System.out.print(narr[i] + " ");

        if (flag > 1)
            System.out.println("\n The frequency of " + flagnum + " is: " + flag);
        else
            System.out.println("\n None of the numbers are repeated");
    }
}
```

Output:

```
Enter the number of elemer

9
Enter the 1 element
2
Enter the 2 element
1
Enter the 3 element
4
Enter the 4 element
7
Enter the 5 element
1
Enter the 6 element
8
Enter the 7 element
5
Enter the 8 element
6
Enter the 9 element
3

The entered array is:
2 1 4 7 1 8 5 6 3
 The new array is:
2 1 4 7 0 8 5 6 3
 The frequency of 1 is: 2
```

Name: Harshita Pasupuleti

Reg No: 21BCE8421