

LAB ASSIGNMENT - 1

WAP FOR RECURSIVE BINARY & LINEAR SEARCH

BINARY SEARCH CODE:

```
import java.util.Scanner;
import java.util.Arrays;
public class BinarySearch {
    int binarySearch(int array[], int ele, int p, int r)
    {

        while (p <= r)
        {

            int q = p + (r - p) / 2;

            if (array[q] == ele)
            {
                return q;
            }

            if (array[q] < ele)
            {
                p = q + 1;
            }
            else
            {
                r = q - 1;
            }
        }

        return -1;
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
```

```

        BinarySearch obj = new BinarySearch();

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the Size of Array");
        int n = sc.nextInt();
        int[] array = new int[n];

        System.out.println("Enter the elements of
Array");
        for (int i=0;i<n;i++)
        {
            array[i] = sc.nextInt();
        }
        Arrays.sort(array);

        System.out.println("Enter number to be
searched:");
        int ele = sc.nextInt();
        sc.close();

        int result = obj.binarySearch(array, ele, 0, n -
1);

        if (result == -1)
        {
            System.out.println("Not found");
        }
        else
        {
            System.out.println("number found at position "
+ result + " & the number is " + ele);
        }
    }
}

```

BINARY SEARCH CODE OUTPUT:

The screenshot displays the Eclipse IDE interface with the following components:

- Package Explorer:** Shows the project structure with files like BinarySearch.java, BubbleSort.java, InsertionSort.java, LinearSearch.java, MergeSort.java, QuickSort.java, selectionsort.java, module-info.java, LICENSE, and README.md.
- Editor:** Displays the code for BinarySearch.java. The code includes a main method that prompts the user for the size of the array, the elements of the array, and the number to be searched. It then performs a binary search and prints the result.
- Console:** Shows the output of the program. It displays the prompts and the user's input: "Enter the Size of Array : 7", "Enter the elements of Array : 34 67 23 14 56 47 89", and "Enter number to be searched: 78". The final output is "Not found".
- Problems:** Shows a list of warnings, including "Warning: (5 items)".

```
public static void main(String[] args) {
    // TODO Auto-generated method stub

    BinarySearch obj = new BinarySearch();

    Scanner sc = new Scanner(System.in);

    System.out.println("Enter the Size of Array : ");
    int n = sc.nextInt();
    int[] array = new int[n];

    System.out.println("Enter the elements of Array : ");
    for(int i=0;i<n;i++)
    {
        array[i] = sc.nextInt();
    }
    Arrays.sort(array);

    System.out.println("Enter number to be searched:");
    int ele = sc.nextInt();
    sc.close();

    int result = obj.binarySearch(array, ele, 0, n - 1);

    if (result == -1)
    {
        System.out.println("Not found");
    }
}
```

Enter the Size of Array : 7
Enter the elements of Array : 34 67 23 14 56 47 89
Enter number to be searched: 78
Not found

LINEAR SEARCH CODE:

```
package rec_Program;

import java.util.Scanner;

public class LinearSearch {
    int linearRecursion(int[] arrNumber, int start, int
last, int k)
    {
        if(last < start)
            return -1;
        if(arrNumber[start] == k)
            return start;
        return linearRecursion(arrNumber, start + 1,
last, k);
    }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        {
            LinearSearch obj = new LinearSearch();
            int a, l, key, array[];
            Scanner sc = new Scanner(System.in);
            System.out.println("enter array length: ");
            l = sc.nextInt();
            array = new int[l];
            System.out.println("enter " + l + "
elements");
            for(a = 0; a < l; a++)
            {
                array[a] = sc.nextInt();
            }
            System.out.println("Enter the key value:
");
            key = sc.nextInt();
            int index = obj.linearRecursion(array, 0, l
- 1, key);
            if(index != -1)
            {
```

LINEAR SEARCH CODE OUTPUT: