

**LAB-02**

**Title: Algorithm to search an element using Binary Search.**

**Name : Azizul Abedin Azmi**

**ID : 2022-1-60-130**

**Section: 03**

**Course Code: CSE207**

**Course Title: (Data Structures)**

**Date: 19/02/2024**

**Course Instructor:**

**Dr. Anup Kumar Paul**

**Associate Professor**

**Department of Computer Science and Engineering**

**Source Code:**

**Main.java:**

package Lab02;

import java.util.Scanner;

public class Main {

    public static void main(String[] args){

        int[] data = new int[10];

        Scanner input = new Scanner(System.in);

        System.out.print("Enter " +data.length+ " items = ");

        for (int i = 0 ; i < 10; i++){

            data[i] = input.nextInt();

        }

        BinarySearch search = new BinarySearch(data);

        System.out.print("Enter the item you want to search : ");

        int item = input.nextInt();

        search.PrintData();

        int index = search.binarySearch(item);

        if (index != -1) {

            System.out.println("\nItem found at index : " +index);

        } else{

            System.out.println("\nItem not found");

        }

        input.close();

    }

}

**BinearySearch.java:**

package Lab02;

public class BinarySearch {

    int[] data;

    public BinarySearch(int[] data) {

        this.data = data;

    }

    public void PrintData() {

        System.out.println("The Data in the Array");

        for (int i = 0; i < data.length; i++) {

            System.out.print(data[i] + " ");

        }

    }

    public int binarySearch(int item) {

        boolean found = false;

        int location = -1;

        int low = 0;

        int high = data.length - 1;

        while (low <= high) {

            int mid = (low + high) / 2;

            if (item == data[mid]) {

                found = true;

                location = mid;

                break;

            } else if (item < data[mid]) {

                high = mid - 1;

            } else {

                low = mid + 1;

            }

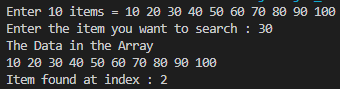
        }

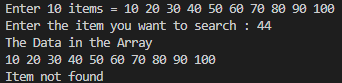
        return location;

    }

}

**Output:**

****

****