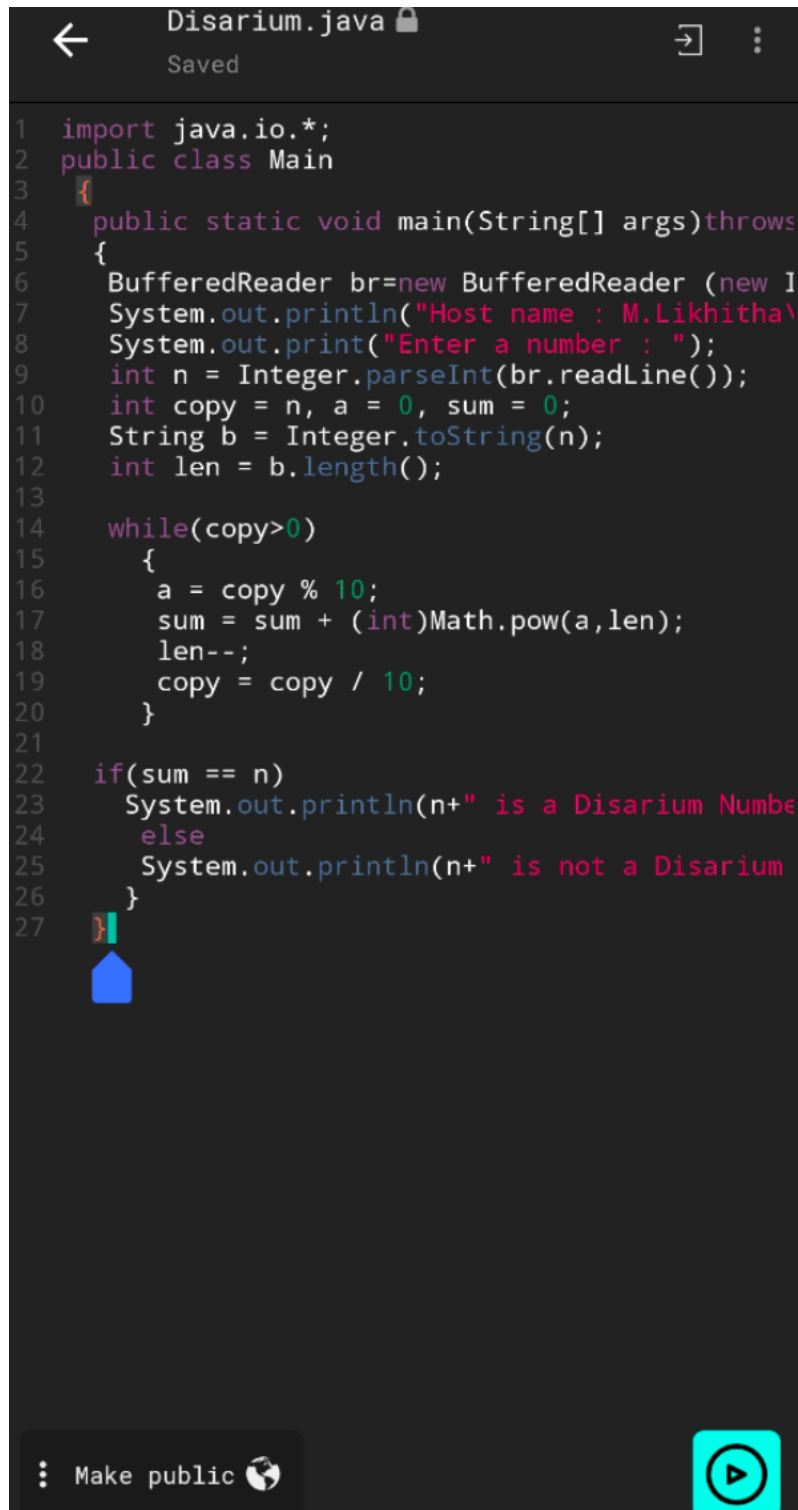


M.Likhitha.

SAP ID : 51834624.

1st Question :



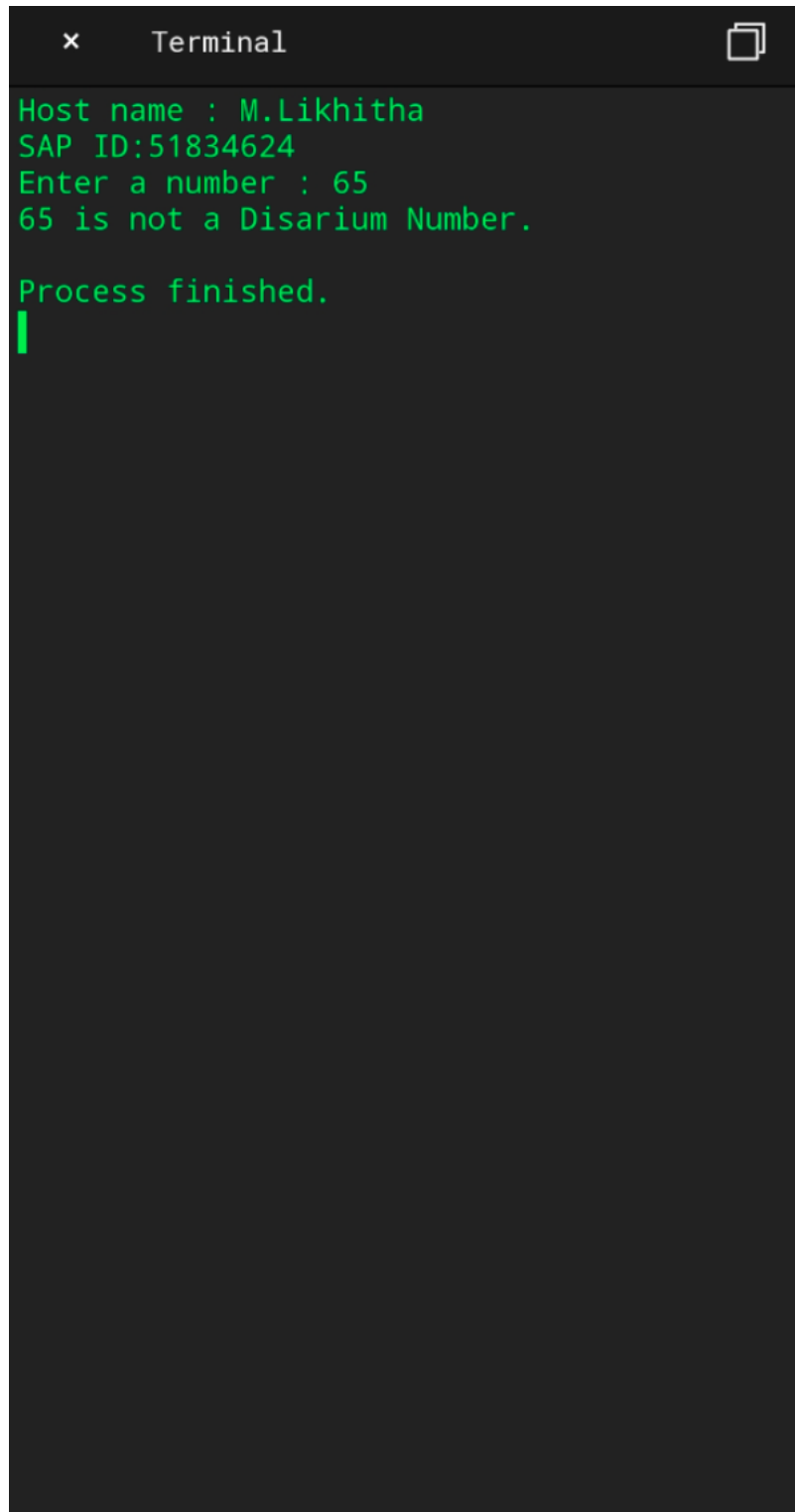
```
1  import java.io.*;
2  public class Main
3  {
4      public static void main(String[] args) throws
5      {
6          BufferedReader br=new BufferedReader (new I
7          System.out.println("Host name : M.Likhitha\
8          System.out.print("Enter a number : ");
9          int n = Integer.parseInt(br.readLine());
10         int copy = n, a = 0, sum = 0;
11         String b = Integer.toString(n);
12         int len = b.length();
13
14         while(copy>0)
15         {
16             a = copy % 10;
17             sum = sum + (int)Math.pow(a,len);
18             len--;
19             copy = copy / 10;
20         }
21
22         if(sum == n)
23             System.out.println(n+" is a Disarium Numbe
24         else
25             System.out.println(n+" is not a Disarium
26         }
27     }
```

Disarium.java Saved

Make public



Output Screen :

A screenshot of a terminal window with a dark background and green text. The window has a title bar with a close button (X) and the word "Terminal". The text inside the terminal reads: "Host name : M.Likhitha", "SAP ID:51834624", "Enter a number : 65", "65 is not a Disarium Number.", and "Process finished." followed by a green cursor line.

```
× Terminal  
Host name : M.Likhitha  
SAP ID:51834624  
Enter a number : 65  
65 is not a Disarium Number.  
Process finished.  
|
```



2nd Question :



The screenshot shows a code editor window titled "SortingOfBinary.java" with a lock icon and a "Saved" status. The code is a Java program implementing a sorting algorithm. It includes an import statement for java.io.*, a class named "countzot", and a static method "sort012" that takes an array "a" and its size "arr_size". The "sort012" method uses a while loop and a switch statement to sort the array. The switch statement has three cases: case 0, case 1, and case 2. Case 0 swaps the element at index "lo" with the element at index "mid" and increments "lo" and "mid". Case 1 increments "mid". Case 2 swaps the element at index "mid" with the element at index "hi" and decrements "hi". After the "sort012" method, there is a "printArray" method that prints the elements of the array. The "main" method prints the host name, initializes an array "arr" with the values {0, 1, 1, 0, 1, 2, 1, 2, 0, 0}, sets "arr_size" to the length of "arr", calls "sort012", and prints the array after sorting. A "Make public" button is visible at the bottom left, and a play button icon is at the bottom right.

```
1 import java.io.*;
2 class countzot {
3     static void sort012(int a[], int arr_size)
4     {
5         int lo = 0;
6         int hi = arr_size - 1;
7         int mid = 0, temp = 0;
8         while (mid <= hi) {
9             switch (a[mid]) {
10                case 0: {
11                    temp = a[lo];
12                    a[lo] = a[mid];
13                    a[mid] = temp;
14                    lo++;
15                    mid++;
16                    break;
17                }
18                case 1:
19                    mid++;
20                    break;
21                case 2: {
22                    temp = a[mid];
23                    a[mid] = a[hi];
24                    a[hi] = temp;
25                    hi--;
26                    break;
27                }
28            }
29        }
30    }
31    static void printArray(int arr[], int arr_si
32    {
33        int i;
34        for (i = 0; i < arr_size; i++)
35            System.out.print(arr[i] + " ");
36        System.out.println("");
37    }
38    public static void main(String[] args)
39    {
40        System.out.println("Host name : M.Likhitha
41        int arr[] = { 0, 1, 1, 0, 1, 2, 1, 2, 0, 0
42        int arr_size = arr.length;
43        sort012(arr, arr_size);
44        println("Array after seg
45        arr, arr_size);
```

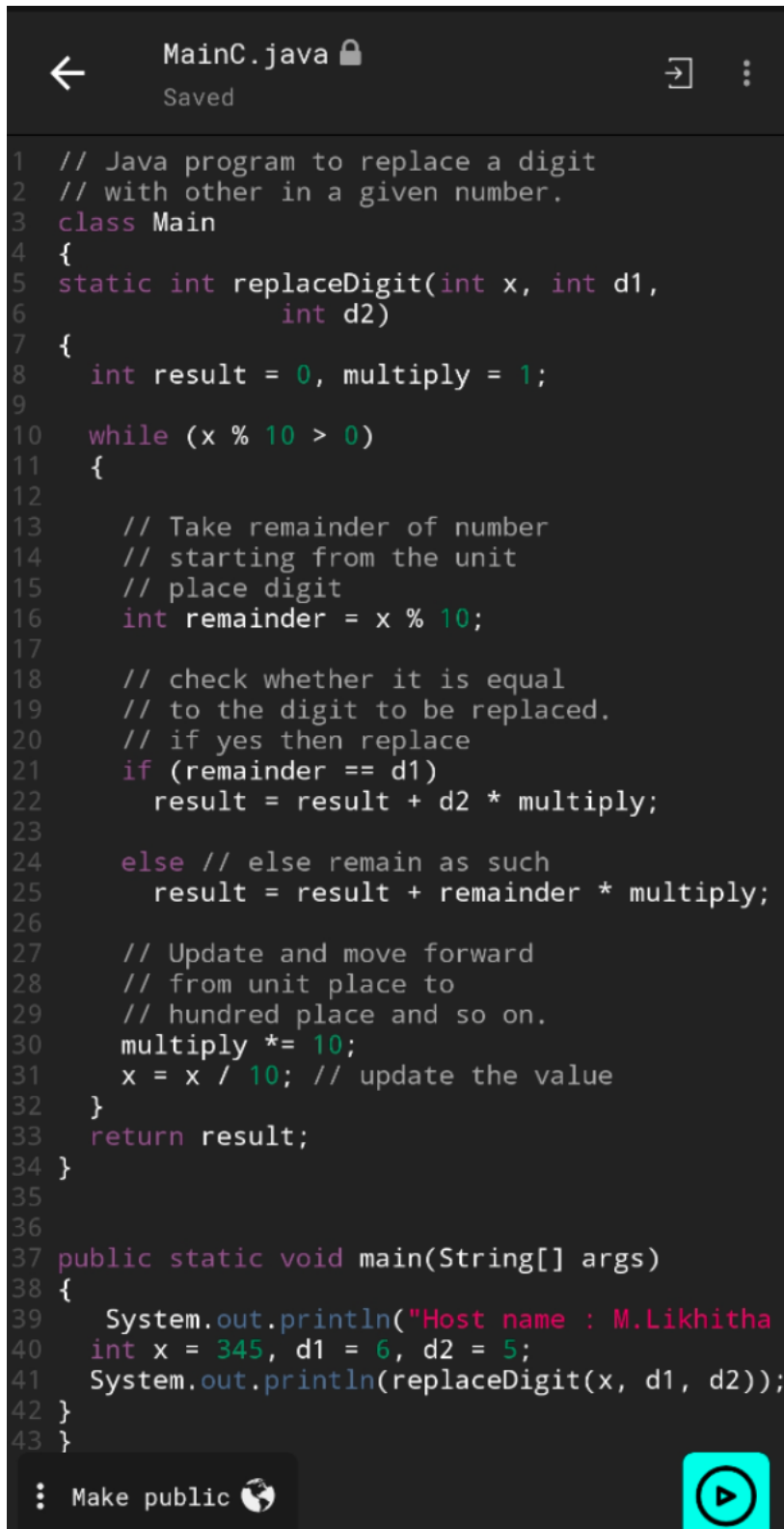


Out Put Screen :

```
x Terminal
Host name : M.Likhitha
Sap id : 51834624
Array after seggregation
0 0 0 0 0 1 1 1 1 1 2 2
Process finished.
```



3rd Question :

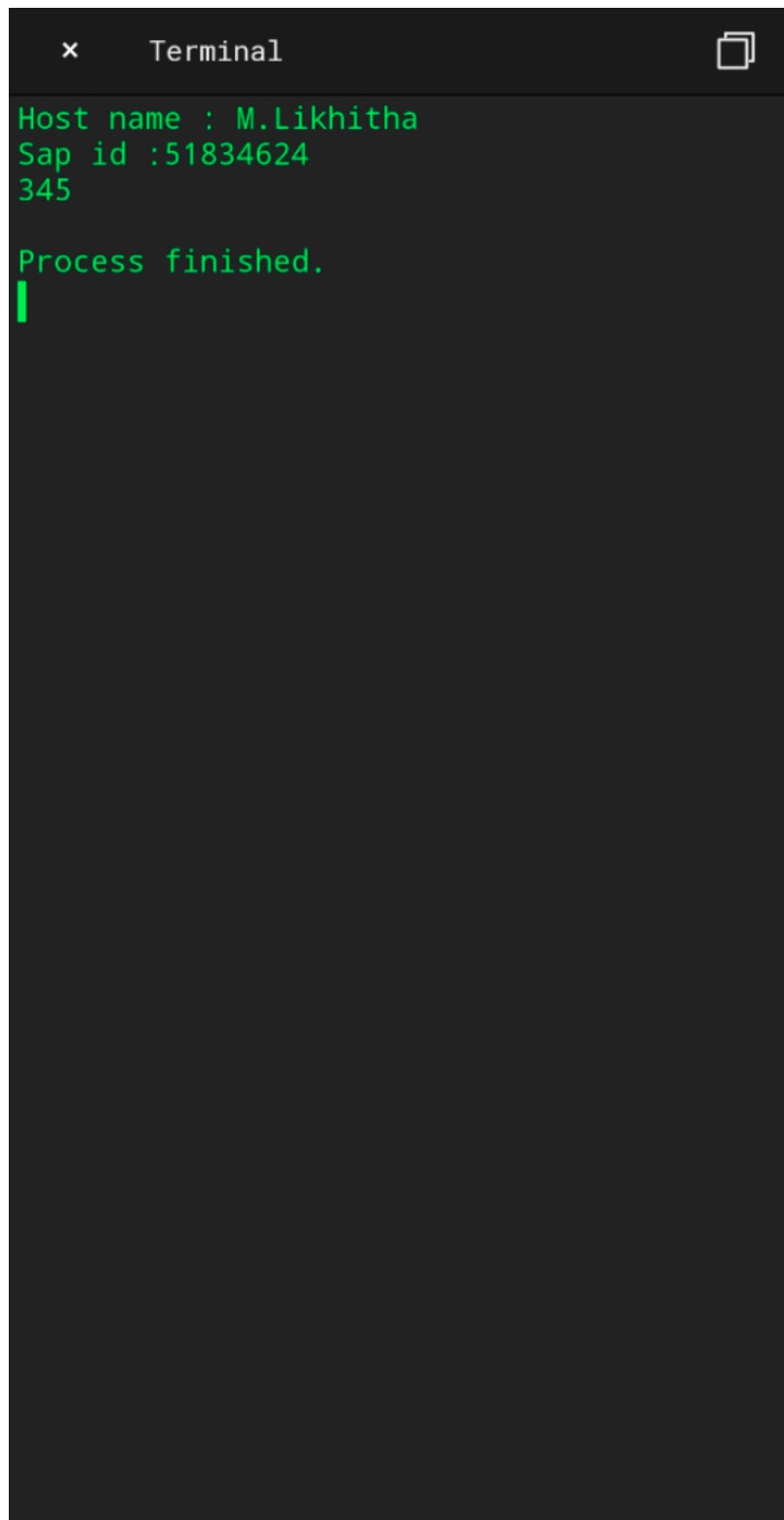


```
1 // Java program to replace a digit
2 // with other in a given number.
3 class Main
4 {
5     static int replaceDigit(int x, int d1,
6                             int d2)
7     {
8         int result = 0, multiply = 1;
9
10        while (x % 10 > 0)
11        {
12            // Take remainder of number
13            // starting from the unit
14            // place digit
15            int remainder = x % 10;
16
17            // check whether it is equal
18            // to the digit to be replaced.
19            // if yes then replace
20            if (remainder == d1)
21                result = result + d2 * multiply;
22
23            else // else remain as such
24                result = result + remainder * multiply;
25
26            // Update and move forward
27            // from unit place to
28            // hundred place and so on.
29            multiply *= 10;
30            x = x / 10; // update the value
31        }
32        return result;
33    }
34 }
35
36
37 public static void main(String[] args)
38 {
39     System.out.println("Host name : M.Likhitha");
40     int x = 345, d1 = 6, d2 = 5;
41     System.out.println(replaceDigit(x, d1, d2));
42 }
43 }
```

Make public



Output Screen :

A screenshot of a terminal window with a dark background. The window has a title bar with a close button (x) and the text 'Terminal'. On the right side of the title bar is a window icon. The terminal content is displayed in green text. It shows 'Host name : M.Likhitha', 'Sap id :51834624', and '345' on the first line. The second line shows 'Process finished.' followed by a green cursor bar.

```
Host name : M.Likhitha
Sap id :51834624
345

Process finished.
|
```



5th Question :

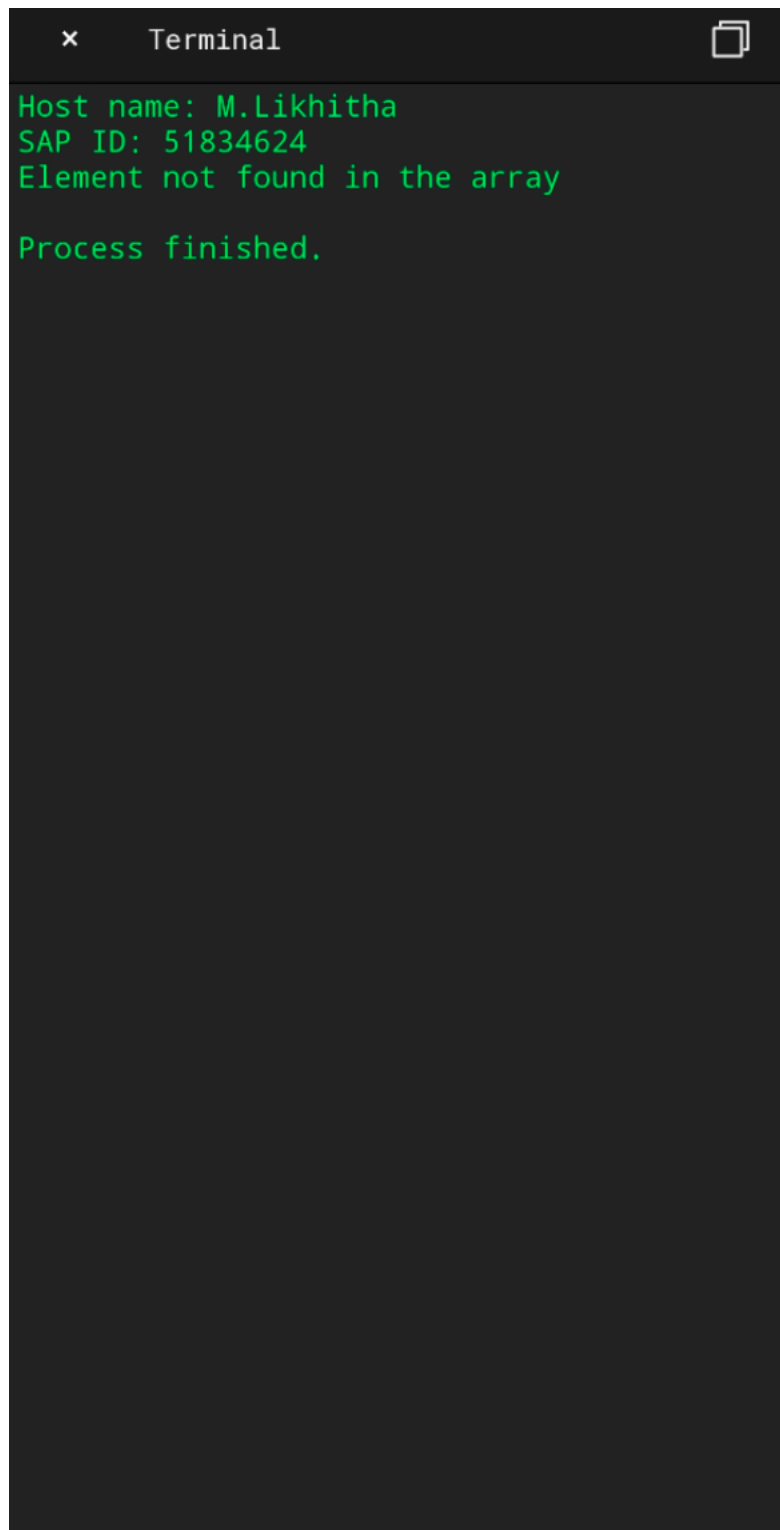
```
BinarySearch.java
Saved

1 public class Main
2 {
3     public static int binarySearch(int[] M, int left, int right, int n)
4     {
5         if (left > right) {
6             return -1;
7         }
8         int mid = (left + right) / 2;
9
10        if (n == M[mid]) {
11            return mid;
12        }
13        else if (n < M[mid]) {
14            return binarySearch(M, left, mid - 1, n);
15        }
16
17        else {
18            return binarySearch(M, mid + 1, right, n);
19        }
20    }
21
22    public static void main(String[] args)
23    {
24        int[] M = { 2, 5, 6, 8, 9, 10 };
25        int key = 3;
26
27        int left = 0;
28        int right = M.length - 1;
29
30        int index = binarySearch(M, left, right, key);
31
32        System.out.println("Host name: M.Likhitha");
33        if (index != -1) {
34            System.out.println("Element found at index: " + index);
35        } else {
36            System.out.println("Element not found in the array");
37        }
38    }
39 }
```

Make public



Output Screen :

A terminal window with a dark background and green text. The title bar at the top shows a close button (x), the word "Terminal", and a maximize button. The text inside the terminal reads: "Host name: M.Likhitha", "SAP ID: 51834624", "Element not found in the array", and "Process finished." on separate lines.

```
Host name: M.Likhitha
SAP ID: 51834624
Element not found in the array
Process finished.
```

