Name - Sanjoy Saha

Stream - Computer Science & Engineering

Sec - A

Roll no. - 3

University Roll no. :- 1090120003

Subject - DSA Lab

```c
#include <stdio.h>

int LINEAR_SEARCH(int inp_arr[], int size, int val)
{
    for (int i = 0; i < size; i++)
        if (inp_arr[i] == val)
            return i;
    return -1;
}
int main(void)
{
    int arr[] = { 10, 20, 30, 40, 50, 100, 0 };
    int key = 20;
    int size = 7;

    int res = LINEAR_SEARCH(arr, size, key);
    if (res == -1)
    printf("ELEMENT NOT FOUND!!");
    else
    printf("Item is present at index %d", res);

    return 0;

    }
```

```c
int main(void)
{
    int arr[] = { 10, 20, 30, 40, 50, 100, 0 };
    int key = 100;
    int size = 7;

}
```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\lenovo\Desktop\C DSA lab> cd "c:\Users\lenovo\Desktop\C DSA lab\" ;
Item is present at index 5
PS C:\Users\lenovo\Desktop\C DSA lab>

```
11  {
12      int arr[] = { 10, 20, 30, 40, 50, 100, 0 };
13      int key = 20;
14      int size = 7;
15
```

PROBLEMS 1   OUTPUT   TERMINAL   DEBUG CONSOLE

PS C:\Users\lenovo\Desktop\C DSA lab> cd "c:\Users\lenovo\Desktop\C DSA lab\" ; if
Item is present at index 1
PS C:\Users\lenovo\Desktop\C DSA lab>

```c
    int arr[] = { 10, 20, 30, 40, 50, 100, 0 };
    int key = 200;
    int size = 7;
```

PROBLEMS 1   OUTPUT   TERMINAL   DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\lenovo\Desktop\C DSA lab> cd "c:\Users\lenovo\Desktop\C DSA lab\"
ELEMENT NOT FOUND!!
PS C:\Users\lenovo\Desktop\C DSA lab>

```c
#include <stdio.h>

int binarySearch(int arr[], int l, int r, int x)
{
    if (r >= l) {
        int mid = l + (r - l) / 2;

        if (arr[mid] == x)
            return mid;

        if (arr[mid] > x)
            return binarySearch(arr, l, mid - 1, x);

        return binarySearch(arr, mid + 1, r, x);
    }

    return -1;
}
int main(void)
{
    int arr[] = { 2, 3, 4, 10, 40 };
    int n = sizeof(arr) / sizeof(arr[0]);
    int x = 10;
    int result = binarySearch(arr, 0, n - 1, x);
    (result == -1)
        ? printf("Element is not present in array")
        : printf("Element is present at index %d", result);
    return 0;
}
```

```c
20  {
21      int arr[] = { 2, 3, 4, 10, 40 };
22      int n = sizeof(arr) / sizeof(arr[0]);
23      int x = 10;
24      int result = binarySearch(arr, 0, n - 1, x);
25      (result == -1)
26          ? printf("Element is not present in array")
27          : printf("Element is present at index %d", result);
28      return 0;
```

```c
        int arr[] = { 2, 3, 4, 10, 40 };
        int n = sizeof(arr) / sizeof(arr[0]);
        int x = 2;
        int result = binarySearch(arr, 0, n - 1, x);
        (result == -1)
            ? printf("Element is not present in array")
            : printf("Element is present at index %d", result);
        return 0;
```

```c
20  {
21      int arr[] = { 2, 3, 4, 10, 40 };
22      int n = sizeof(arr) / sizeof(arr[0]);
23      int x = 200;
24      int result = binarySearch(arr, 0, n - 1, x);
25      (result == -1)
26          ? printf("Element is not present in array")
27          : printf("Element is present at index %d", result);
28      return 0;
```