



**Green University of Bangladesh**  
**Department of Computer Science and Engineering(CSE)**  
**Faculty of Sciences and Engineering**  
**Semester: (Spring, Year:2021), B.Sc. in CSE (Day)**

**LAB REPORT NO 01**  
**Course Title: Algorithms Lab**  
**Course Code: 206          Section: DB**

**Lab Experiment Name:** 1. Linear Search  
2. Binary Search  
3. Bubble Sort

**S tudent Details**

Name		ID
1.	Shamim Ahmed	201902067

**Lab Date** : 22/06/2021  
**Submission Date** : 22/06/2021  
**Course Teacher's Name** : **Monoshi Kumar Roy**

[For Teachers use only: **Don't Write Anything inside this box**]

**Lab Report Status Marks:** .....

**Signature:**.....

## 1.Linear Search

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int k, i, n;
```

```
    printf("Enter the number of array\n");
```

```
    scanf("%d", &n);
```

```
    printf("Enter %d int\n", n);
```

```
    int item[n];
```

```
    for (i = 0; i < n; i++)
```

```
    {
```

```
        scanf("%d", &item[i]);
```

```
    }
```

```
    printf("Enter a number you want to search\n");
```

```
    scanf("%d", &k);
```

```
    for (i = 0; i < n; i++)
```

```
    {
```

```
        if (item[i] == k)
```

```
        {
```

```
            printf("%d is at %d.\n", k, i+1);
```

```
            break;
```

```
        }
```

```
    }
```

```
    if (i == n)
```

```
        printf("%d isn't present in the array.\n", k);
```

```
    return 0;
```

```
}
```

The screenshot shows the Code::Blocks IDE with a C program for linear search. The code is as follows:

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int k, i, n;
6
7     printf("Enter the number of array\n");
8     scanf("%d", &n);
9
10    printf("Enter %d int\n", n);
11    int item[n];
12
13    for (i = 0; i < n; i++)
14    {
15        scanf("%d", &item[i]);
16    }
17    printf("Enter a number you want to search\n");
18    scanf("%d", &k);
19
20    for (i = 0; i < n; i++)
21    {
22        if (item[i] == k)
23        {
24            printf("%d is at %d.\n", k, i+1);
25            break;
26        }
27    }
28    if (i == n)
29        printf("%d isn't present in the array.\n", k);
30
31    return 0;
32 }
33
```

The output window shows the execution of the program:

```
/home/shamim/Desktop/Algo Lab 1 in class
Enter the number of array
3
Enter 3 int
2 5 7
Enter a number you want to search
7
7 is at 3.

Process returned 0 (0x0)   execution time : 19.604 s
Press ENTER to continue.
```

## 2. Binary Search

```
#include <stdio.h>
```

```
int main()
{
    int n, find;
    printf("Enter number of elements\n");
    scanf("%d", &n);

    int item[n];

    printf("Enter %d int\n", n);

    for (int i = 0; i < n; i++)
    {
        scanf("%d", &item[i]);
    }

    printf("Enter the Item to search \n");
    scanf("%d", &find);

    int first = 0;
    int last = n - 1;
    int mid = (first+last)/2;

    while (first <= last)
    {
        if (item[mid] < find)
            first = mid + 1;
        else if (item[mid] == find)
        {
            printf("%d is at position %d.\n", find, mid+1);
            break;
        }
        else
            last = mid- 1;

        mid = (first + last)/2;
    }
    if (first > last)
        printf("%d isn't present in the Array.\n", find);

    return 0;
}
```

The screenshot shows a code editor window titled 'algo lab 1 class 1 code 3.c - Code::Blocks 20.03'. The code is a binary search function. The terminal window on the right shows the execution: 'Enter number of elements', '3', 'Enter 3 int', '4 6 8', 'Enter the Item to search', '6', '6 is at position 2.', 'Process returned 0 (0x0) execution time : 15.931 s', 'Press ENTER to continue.'

```
11 printf("Enter %d int\n", n);
12
13 for (int i = 0; i < n; i++)
14 {
15     scanf("%d", &item[i]);
16 }
17
18 printf("Enter the Item to search \n");
19 scanf("%d", &find);
20
21 int first = 0;
22 int last = n - 1;
23 int mid = (first+last)/2;
24
25 while (first <= last)
26 {
27     if (item[mid] < find)
28         first = mid + 1;
29     else if (item[mid] == find)
30     {
31         printf("%d is at position %d.\n", find, mid+1);
32         break;
33     }
34     else
35         last = mid - 1;
36
37     mid = (first + last)/2;
38 }
39 if (first > last)
40     printf("%d isn't present in the Array.\n", find);
41
42 return 0;
43 }
```

### 3. Bubble Sort

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int n, i, swap;
```

```
    printf("Enter array num\n");
```

```
    scanf("%d", &n);
```

```
    int item[n];
```

```
    printf("Enter %d int\n", n);
```

```
    for (i = 0; i < n; i++)
```

```
        scanf("%d", &item[i]);
```

```
    for (i = 0 ; i < n-1 ; i++)
```

```
    {
```

```
        for (int j = 0 ; j < n-i -1; j++)
```

```
        {
```

```
            if (item[j] > item[j+1])
```

```
            {
```

```
                swap    = item[j];
```

```
                item[j]  = item[j+1];
```

```
                item[j+1] = swap;
```

```
            }
```

```
        }
```

```
    }
```

```
    printf("Sorted list :\n");
```

```
    for (i = 0; i < n; i++)
```

```
        printf("%d ", item[i]);
```

```
    return 0;
```

```
}
```

algo lab 1 class 1 code 2.c - Code::Blocks 20.03

File Edit View Search Project Build Debug Tools Plugins Settings Help

Start here | Algo Lab 1 in class.c | algo lab 1 class 1 code 2.c | algo lab 1 class 1 code 3.c

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int n, i, swap;
6
7     printf("Enter array num\n");
8     scanf("%d", &n);
9     int item[n];
10
11     printf("Enter %d int\n", n);
12
13     for (i = 0; i < n; i++)
14         scanf("%d", &item[i]);
15
16     for (i = 0; i < n-1; i++)
17     {
18         for (int j = 0; j < n-i-1; j++)
19         {
20             if (item[j] > item[j+1])
21             {
22                 swap = item[j];
23                 item[j] = item[j+1];
24                 item[j+1] = swap;
25             }
26         }
27     }
28
29     printf("Sorted list :\n");
30
31     for (i = 0; i < n; i++)
32         printf("%d ", item[i]);
33 }
```

/home/shamim/Desktop/algo lab 1 class 1 code 2

Enter array num  
5  
Enter 5 int  
2 5 7 1 3  
Sorted list :  
1 2 3 5 7  
Process returned 0 (0x0) execution time : 17.882 s  
Press ENTER to continue.

logs & others