Lab Report

Course Code: CSE135
Course Name: Data Structure Lab

Lab No: 02

Submitted to
Dr. Mohammad Shamsul Arefin
Professor
Department of CSE
Daffodil International University

Submitted By
Name: Md Shefat Al Mahmud
ID: 213-15-4364
Section: PC-C
Department of CSE
Daffodil International University

Questions:

1. Binary Search.
2. Insertion.
3. Deletion.
4. Frequency count & Number of occurrences of each item.

Code: (Ideone Link: https://ideone.com/ej3rtL)

```
#include <bits/stdc++.h>
using namespace std;

#define nl      cout << "\n";
#define MAX_INT 100002

/*-----------------------------------------------------------------
--------*/


int binarysearch(int a[], int l, int r, int x)
{
    while (l <= r) {
        int m = l + (r - l) / 2;
        if (a[m] == x) return m;
        if (a[m] < x) l = m + 1;
        else r = m - 1;
    }
    return -1;
}

int* insertion(int a[], int n , int pos, int x)
{
    for(int i = n+1 ; i >= pos ; i--){
        a[i] = a[i-1];
    }
    a[pos -1] = x;
    return a;
}
int deletion(int a[], int n , int pos)
{
    for(int i = pos; i < n ; i++){
```

```
        a[i] = a[i+1];
    }
    return n;
}


void solve()
{
    cout << "Enter Array Size: ";
    int n;
    cin >> n;
    int a[n+10];
    //Array Input
    cout << "Enter Array Elements: ";
    for(int i = 0 ; i < n ; i++){
        cin >> a[i];
    }

    //  Task #01
    ///Binary Search
    cout << "Binary Search:";nl;
    cout << "Enter a number to search: ";
    int x;
    cin >> x; // item to search
    int out = binarysearch(a, 0, n-1, x);
    if(out == -1) cout << "Item not found";
    else cout << "Item found at index: " << out;
    nl;nl;

    //  Task #02
    ///Insertion
    cout << "Insertion:";nl;
    cout << "Enter the value to insert: ";
    int val, pos;
    cin >> val ;
    cout << "Enter the position to insert: ";
```

```
cin >> pos;
cout << "After inserting: ";
insertion(a, n , pos , val);
for(int i = 0 ; i <n ; i++){
    cout << a[i] << ' ';
}
cout << a[n];
nl;nl;

//  Task #03
///Deletion
cout << "Deletion:";nl;
cout << "Enter the location to delete: ";
int loc;
cin >> loc;
deletion(a, n ,loc);

cout << "After deleting: ";
for(int i = 0 ; i <n-1 ; i++){
    cout << a[i] << ' ';
}
cout << a[n-1];
nl;nl;



//  Task #04
///Frequency count & Number of occurences of each item
cout << "Number Frequency count:";nl;
cout << "Value - Occurences";nl;
int freq[MAX_INT] = {0}, mx= 0;
for(int i = 0 ; i< n ; i++){
    freq[a[i]]++;
    if(a[i] > mx) mx = a[i];
}
for(int i = 1 ; i <= mx ; i++){
    if(freq[i] > 0){
```

```
            cout << setw(3) << i << setw(10) << freq[i];
            nl;
        }
    }
}


int main()
{
    solve();
    return 0;
}
```

Sample Input & Output:

Enter Array Size: 10

Enter Array Elements: 1 2 3 4 5 6 7 8 9 10

Binary Search:

Enter a number to search: 4

Item found at index: 3

Insertion:

Enter the value to insert: 4

Enter the position to insert: 2

After inserting: 1 4 2 3 4 5 6 7 8 9 10

Deletion:

Enter the location to delete: 6

After deleting: 1 4 2 3 4 5 7 8 9 10

Number Frequency count:

Value - Occurences

1	**1**
2	**1**
3	**1**
4	**2**
5	**1**
7	**1**
8	**1**
9	**1**
10	**1**

Screenshots: