**Experiment 1.2**

**Student Name: Vivek Branch: BE-CSE**

**UID: Section/Group:**

**Date of performance: Subject name: Data Structures**

**AIM:**

Write a program to demonstrate the use of linear and binary search to find a given element in an array.

**PROGRAM CODE:**

#include <iostream>

using namespace std;

int main()

{

*// Declaring variables and taking array as input from the user*

    int size;

    int position, fact = 0, n;

    cout << "\n";

    cout << "Enter number of elements - ";

    cin >> size;

    int a[size], arr[size];

    cout << "\nEnter Elements of the array: " << endl;

    for (int i = 0; i < size; i++)

    {

        cin >> a[i];

    }

*// Element to search in the array*

    int f;

    cout << "Enter element to search: ";

    cin >> f;

*//performing Linear Search*

    cout << "\n------------LINEAR SEARCH------------" << endl;

    for (int j = 0; j < size; j++)

    {

        if (a[j] == f)

        {

            fact = 1;

            position = j;

        }

    }

    if (fact == 1)

    {

        cout << "\nElement is found at index " << position << endl;

        cout << "\n";

    }

    else

    {

        cout << "\nElement not found in the array.";

        cout << "\n";

    }

*//performing Binary Search*

    cout << "\n------------BINARY SEARCH------------" << endl;

    {

        int low, high, mid;

        int temp;

*//Sorting array because Binary search can only be implemented on sorted array.*

        for (int j = 0; j < size; j++)

        {

            for (int i = 0; i < size; i++)

            {

                if (a[i] > a[i + 1])

                {

                    temp = a[i];

                    a[i] = a[i + 1];

                    a[i + 1] = temp;

                }

            }

        }

        cout << "\nSorted Array : " << endl;

        for (int i = 0; i < size; i++)

            cout << a[i] << " ";

        cout << "\n\n";

*//Searching element using Binary Search*

        high = 0;

        low = size - 1;

        mid = ((high + low) / 2);

        while (high <= low)

        {

            if (a[mid] < f)

            {

                high = mid + 1;

            }

            else if (a[mid] == f)

            {

                cout << "Number Found at index " << mid;

                cout << "\n\n";

                break;

            }

            else

            {

                low = mid - 1;

            }

            mid = (high + low) / 2;

        }

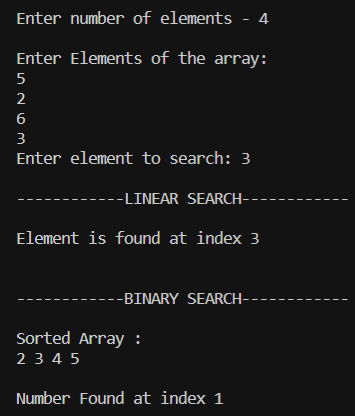
        if (high > low)

            cout << "Element not found in the array.";

    }

}

**OUTPUT:**

****