**DATA STRUCTURES LAB EXPERIMENTS**

**Experiment Number:** 10

Sanskruti Samant

USN: 2GI17EC113

DIV: B

**Title:** Implementation of Binary Search in C++.

**Problem Statement:**

Write a C++ program to perform binary search.

**Algorithm:**

1. Create the array of size specified by the user.
2. Read the array elements in ascending order.
3. Read the element ‘X’ to be searched from the user.
4. Compare ‘X’ with the middle element of the sorted array.
5. If x matches with middle element, we return the mid index.
6. Else If ‘X’ is greater than the mid element, then ‘X’ can only lie in right half subarray after the mid element. So we recur for right half.
7. Else (x is smaller) recur for the left half.

**Code:**

// Binary Search using Recursion

#include <iostream>

using namespace std;

// If found, location of x in the array is returned.

// otherwise -1 is returned.

int binarySearch(int arr[], int first, int last, int x)

{

if (last >= first) {

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

// If the element is present at the middle of the array

if (arr[mid] == x)

return mid;

// If element is smaller than mid, then

// it can only be present in left subarray

if (arr[mid] > x)

return binarySearch(arr, first, mid - 1, x);

// Else the element can only be present

// in right subarray

return binarySearch(arr, mid + 1, last, x);

}

// If the element is not found.

return -1;

}

int main()

{

int arr[20],n,x,i;

cout<<"How many elements?";

cin>>n;

cout<<"\nEnter elements of the array\n";

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

cin>>arr[i];

cout<<"\nEnter element to search:";

cin>>x;

int result = binarySearch(arr, 0, n - 1, x);

if (result == -1)

cout << "Element is not present in array.";

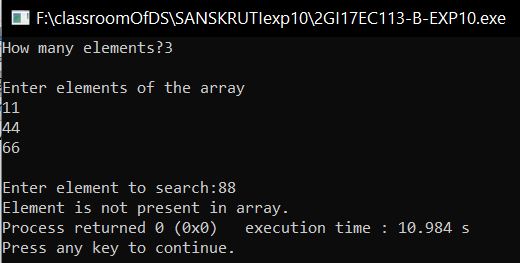
else

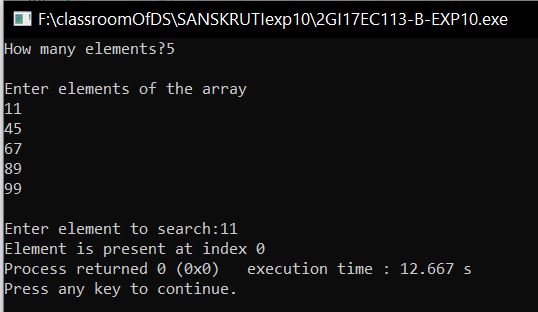
cout << "Element is present at index " << result;

return 0;

}

**Sample Input/ Output:**

****

****

**Analysis:**

* Binary search is an efficient algorithm for finding an item from a sorted list of items.
* It works by repeatedly dividing in half the portion of the list that could contain the item, until you've narrowed down the possible locations to just one. But it has disadvantages.
* It is implemented using recursion so it requires more stack space and to perform binary search the array has to be sorted otherwise the linear search is the only option.