**Practical-6(a)**

**Q. Program for binary search using recursion.**

**Code:-**

//Harsh Bamotra AC-1216

//Program for binary search using recursion

#include <iostream>

using namespace std;

//defining function for binary search

int binary(int s , int high , int low , int arr[])

{

if(high>=1) //checking for empty array

{

int mid=low+(high-1)/2;

if(arr[mid]==s) //searching in the mid index

{ //and if found then returning the index

return mid;

}

else if(arr[mid]>s) //checking in the lower part

{ //of the array

return binary(s , mid-1 , low , arr );

}

else //checking in the upper part

{ //of the array

return binary(s , high , mid+1 , arr);

}

}

return -1; //returning -1 to check for exception

}

int main()

{

int n , s; //defining variables

cout << "Enter the number of elements::";

cin >> n; //taking number of elements

int arr[n];

cout << "Enter the elements in ascending order::" << endl;

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

{

cin >> arr[i]; //initializing the elements in the array

}

cout << "The array you entered::";

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

{

cout << arr[i] << " "; //printing the array

}

cout << endl << "Enter the element you want to search::";

cin >> s; //taking the search element

int r=binary(s , n-1 , 0 , arr);

if(r==-1)

{

cout << "Element not found !!";

}

else //printing the final result

{

cout << "Element found at index::" << r;

}

return 0;

}

**Output:-**

**Normal case**

Graphical user interface, application

Description automatically generated

**Exception case**

A picture containing text, screenshot, computer, monitor

Description automatically generated