**Program for Binary Search**

**// C program to implement recursive Binary Search**

**#include <stdio.h>**

**// A recursive binary search function. It returns**

**// location of x in given array arr[l..r] is present,**

**// otherwise -1**

**int binarySearch(int arr[], int l, int r, int x)**

**{**

**if (r >= l)**

**{**

**int mid = l + (r - l) / 2;**

**// If the element is present at the middle**

**// itself**

**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, l, mid - 1, x);**

**// Else the element can only be present**

**// in right subarray**

**return binarySearch(arr, mid + 1, r, x);**

**}**

**// We reach here when element is not**

**// present in array**

**return -1;**

**}**

**int main(void)**

**{**

**int i, size = 0;**

**int x = 0;**

**printf("Enter size of array: ");**

**scanf("%d",&size);**

**int arr[size];**

**printf("Enter array elements: ");**

**for(i = 0; i < size; i++){**

**scanf("%d",&arr[i]);**

**}**

**printf("Enter element to be searched in array: ");**

**scanf("%d",&x);**

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

**(result == -1)**

**? printf("Element is not present in array")**

**: printf("Element is present at index %d", result);**

**return 0;**

**}**

**Output:**

