**Count of smaller elements**

Show Topic Tags   

Given an sorted array of size n. Find number of elements which are less than or equal to given element.

**Input:**  
The first line of input contains an integer T denoting the number of test cases. Then T test cases follow. Each test case contains an integer n denoting the size of the array. Then the next line contains n space separated integers forming the array.

**Output:**  
Print the number of elements which are less than or equal to given element.

**Constraints:**  
1<=T<=10^5  
1<=n<=10^5  
1<=a[i]<=10^5  
  
**Example:  
Input:**  
2  
6  
1 2 4 5 8 10  
9  
7  
1 2 2 2 5 7 9  
2

**Output:**  
5  
4

\*\*For More Examples Use Expected Output\*\*

<http://practice.geeksforgeeks.org/problems/count-of-smaller-elements/0>

#include <iostream>

#include <stdio.h>

using namespace std;

int binarySearchCount(int arr[], int n, int key)

{

int left = 0, right = n;

int mid;

while (left < right)

{

mid = left + (right-left)/2;

// Check if key is present in array

if (arr[mid] == key)

{

// If duplicates are present it returns

// the position of last element

while (arr[mid+1] == key && mid+1<n)

mid++;

break;

}

// If key is smaller, ignore right half

else if (arr[mid] > key)

right = mid;

// If key is greater, ignore left half

else

left = mid + 1;

}

// If key is not found in array then it will be

// before mid

while (arr[mid] > key)

mid--;

// Return mid + 1 because of 0-based indexing

// of array

return mid + 1;

}

int main() {

int t;

scanf("%d", &t);

while(t-- > 0) {

int n;

scanf("%d", &n);

int a[n];

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

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

}

int key;

scanf("%d", &key);

cout << binarySearchCount(a,n,key) << endl;

}

//system("pause");

return 0;

}