
La última posición

X31870_es

Disponemos de una lista ordenada de n números x_0, x_1, \dots, x_{n-1} y de un número z tal que $x_0 \leq z < x_{n-1}$. Se pide una función `effi_last_pos` con un código muy eficiente que calcule la última posición i tal que $x_i \leq z$. La función debe estar convenientemente documentada y debe utilizarse para completar el siguiente programa.

```
#include <iostream>
#include <vector>
using namespace std;

///////////////////////////////
//  

// documentation and code of effi_last_pos
// function must be here
//  

///////////////////////////////

//gets vector v from input channel
void read_vector(vector<int>& v) {
    int n = v.size();
    for (int i = 0; i < n; ++i) cin >> v[i];
}

int main() {
    int n;
    cin >> n;
    vector<int> v(n);
    read_vector(v);
    int z;
    while (cin >> z)
        cout << effi_last_pos(v, z) << endl;
}
```

Puntos examen: 1.750000 **Parte automática:** 40.000000%

Entrada

La entrada tiene tres partes. En la primera aparece un número entero n mayor que uno. Luego hay una lista de n enteros ordenados de menor a mayor x_0, x_1, \dots, x_{n-1} . Por último, encontramos una secuencia de números enteros. Cada número z de la última secuencia es tal que $x_0 \leq z$ y $z < x_{n-1}$.

Salida

Para cada número z de la secuencia, una línea con la última posición i donde el valor de la lista x_i no supere a z , es decir $x_i \leq z$.

Ejemplo de entrada 1

10
11 23 34 55 55 55 55 55 55 55 65
55
15
48

Ejemplo de salida 1

802

Ejemplo de entrada 2

```
5
123 345 1071 1100 1278
1099
345
```

Ejemplo de salida 2

2
1

Ejemplo de entrada 3

Ejemplo de salida 3

28
0 0 0 0 0 0 0 0 1

Observación

Las implementaciones de la función `effi_last_pos` que puedan tener un tiempo de ejecución proporcional a n serán invalidadas.

Información del problema

Autor : Pro1

Generación : 2020-05-25 16:09:25

© Jutge.org, 2006–2020.

<https://jutge.org>