

---

## Vector xulo

X83303\_ca

---

Per resoldre aquest problema heu de completar el codi que trobareu al final de l'enunciat. Hi heu de reemplaçar cada ??? amb una expressió de codi. **No canvieu res més.** Descarregueu-vos de la web del problema el fitxer code.cc amb el codi a completar (clicqueu el botó ".CPP" corresponent), editeu-lo i envieu-lo al jutge. També us facilitem un fitxer main.cc per ajudar-vos a provar la vostra solució, que **no** heu d'enviar al jutge.

En aquest problema, diem que un vector de  $n$  nombres enters  $v[0..n-1]$  és *xulo* si  $n \geq 2$ ,  $v[0] > v[n-1]$ , i existeix un índex  $j$  entre 0 i  $n-2$  que satisfà:

- $v[0] \leq \dots \leq v[j-1] \leq v[j]$ ,
- $v[j+1] \leq v[j+2] \leq \dots \leq v[n-1]$ .

Per exemple, el vector [12, 12, 15, 20, 1, 3, 3, 5, 9] és xulo (amb  $j = 3$ ).

Implementeu una funció *eficient*

```
int search(int x, const vector<int>& v);
```

que retorni la posició de la *primera* ocurrència de  $x$  en un vector xulo  $v$ . Si  $x$  no pertany a  $v$ , retorna un -1.

## Precondició

El vector  $v$  és xulo.

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

```
int position(const vector<int>& v, int e, int d) {
    if (e+1 == d) return e;
    int m = (e+d)/2;
    if (???) return position(v, m, d);
    else return position(v, e, m);
}
```

```
int search(int x, const vector<int>& v, int e, int d) {
    if (e > d) return -1;
    if (e == d) return (v[e] == x ? e : -1);
    int m = ???
    if (???) return search(x, v, ???, d);
    else return search(x, v, e, ???);
}
```

```
int search(int x, const vector<int>& v) {
    int n = v.size();
```

```
int j = position(v, 0, n-1);  
int p = search(x, v, 0, j);  
if (p ≠ -1) return p;  
return search(x, v, j+1, n-1);  
}
```

## Informació del problema

Autor : Enric Rodríguez

Generació : 2020-04-21 06:29:06

© *Jutge.org*, 2006–2020.

<https://jutge.org>