

Última posición de equilibrio

Implementa una función que resuelva el siguiente problema:

```
{true}
proc equilibrio(vector<int> v) dev int p
{-1 ≤ p < v.size() ∧ numUnos(v, p + 1) = numCeros(v, p + 1)
 ∧ ∀k : p < k < v.size() : numUnos(v, k + 1) ≠ numCeros(v, k + 1)}
```

donde $\text{numUnos}(v, j) = \#i : 0 \leq i < j : v[i] = 1$ y $\text{numCeros}(v, j) = \#i : 0 \leq i < j : v[i] = 0$.

Entrada

La entrada comienza con el número de casos de prueba. Cada caso de prueba tiene dos líneas. En la primera línea se indica el tamaño del vector, en la segunda los valores separados por blancos.

Salida

Para cada caso de prueba se escribe en una línea el valor de p pedido.

Entrada de ejemplo

```
5
8
2 1 1 0 2 3 0 2
6
0 1 0 1 1 1
5
1 1 1 1 1
4
2 3 2 3
1
0
```

Salida de ejemplo

```
7
3
-1
3
-1
```

Nota

Este ejercicio debe verse en el contexto de la asignatura de Estructura de Datos y Algoritmos (EDA), FDI-UCM 2017/2018 (prof. Clara Maria Segura Diaz). Por tanto *no* vale cualquier solución, sino sólo aquellas que utilicen los conceptos de EDA. Es muy posible que se den aclaraciones adicionales en clase a este respecto.