

Vector divisible en pares e impares

Dado un vector de naturales v y un entero n ($0 \leq n \leq 10.000$) se desea implementar un algoritmo de coste lineal en n que satisfaga la siguiente especificación:

```
{0 ≤ n ≤ 10000}
proc paresImpares(int v[], int n, E/S bool existe, E/S int p)
{ existe = (∃q : 0 ≤ q ≤ n : pares(v, 0, q) ∧ impares(v, q, n)) ∧
  existe → (0 ≤ p ≤ n ∧ pares(v, 0, p) ∧ impares(v, p, n)) }
```

donde:

$$\text{pares}(v, s, t) = \forall j : s \leq j < t : v[j] \% 2 = 0$$
$$\text{impares}(v, s, t) = \forall i : s \leq i < t : v[i] \% 2 \neq 0$$

Entrada

La primera línea contiene un número que indica el número de casos de prueba que aparecen a continuación.

Cada caso de prueba se compone de dos líneas. La primera de ellas tiene un único entero con el número de elementos del vector (como mucho 10.000 elementos), mientras que la segunda línea contiene la lista con el contenido del vector.

Salida

Por cada caso de prueba aparecerá una línea independiente con la siguiente información:

"No" : En caso de que la posición no exista.

"Si p": En caso de que la posición exista y sea p.

Entrada de ejemplo

```
2
10
2 4 6 8 10 1 3 5 7 9
5
6 2 3 1 0
```

Salida de ejemplo

```
Si 5
No
```

Nota

Este ejercicio debe verse en el contexto de la asignatura de Estructura de Datos y Algoritmos (EDA), FDI-UCM 2016/2017 (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.