

Elementos videntes

Diseñar un algoritmo que cumpla la siguiente especificación:

```
{0 ≤ n ≤ 100.000}  
proc mayorVidente(int v[], int n, E/S bool existe, E/S int p)  
{ existe = (∃q : 0 ≤ q < n : esVidente(v, q))  
  existe → p = max r : 0 ≤ r < n ∧ esVidente(v, r) : r }
```

donde:

$$esVidente(v, k) = (v[k] = \sum i : k < i < n : v[i])$$

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 100.000 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 *existe* sea falso.

"Si p": En caso de que *existe* sea cierto.

Entrada de ejemplo

```
4  
10  
3 5 8 25 12 14 5 7 0 2  
7  
6 2 3 1 0 8 12  
3  
1 4 0  
8  
9 45 5 20 10 1 6 3
```

Salida de ejemplo

```
Si 5  
No  
Si 2  
Si 4
```