

¿Está bien dividido el vector?

Nos dan un vector de números enteros con n elementos y una cierta posición p del vector. Debemos comprobar si todas las componentes del vector en las posiciones $[0..p]$ (incluida la p) son menores (estrictamente) que todas las componentes en las posiciones $[p+1..n-1]$.

Si $p=n-1$ el resultado debe ser cierto.

Requisitos de implementación.

Se implementará una función que reciba un vector con los valores y el valor p y que devuelva un valor booleano que indique si el vector está bien dividido.

El orden de complejidad del algoritmo debe ser lineal respecto al número de elementos del vector.

Entrada

La entrada comienza con un valor entero que indica el número de casos de prueba. Cada caso de prueba consta de dos líneas. La primera indica el número de elementos del vector y la posición que debería separar los valores menores de los mayores. La segunda línea contiene los valores del vector.

El número de valores de entrada n cumple: $1 \leq n \leq 300.000$ y la posición p cumple $0 \leq p \leq n-1$. Los valores del vector son números enteros que pueden almacenarse en una variable de tipo `int`.

Salida

Para cada caso de prueba se escribe en una línea *SI* si la posición dada separa el vector en la forma apropiada y *NO* si no lo hace.

Entrada de ejemplo

```
6
15 5
5 7 3 4 7 3 8 9 8 10 8 9 8 9 10
4 0
4 7 5 6
4 0
8 7 5 6
4 3
4 3 6 2
5 2
5 3 4 6 4
6 3
-8 -5 -7 -4 -2 -3 -1
```

Salida de ejemplo

```
SI
SI
NO
SI
NO
SI
```