

Vectores ordenados

El problema

Un vector está ordenado crecientemente cuando cada elemento es menor o igual que todos los que le suceden. Por ejemplo, (5, 5, 10, 10, 20) está ordenado crecientemente, mientras que (5, 5, 10, 20, 10) no lo está.

Debes desarrollar un algoritmo **eficiente** que, dado un vector de enteros, determine si el vector está o no ordenado crecientemente.

Programa de prueba

Cada caso de prueba consiste en dos líneas, la primera con el número de elementos del vector, y la segunda con los elementos del vector en sí (el programa soportará la lectura de vectores de, como máximo, un millón de elementos). Para cada caso de prueba el programa imprimirá SI si el vector está ordenado crecientemente, y NO en caso contrario. La entrada finaliza con una línea con -1. A continuación se muestra un ejemplo de entrada/salida:

Entrada	Salida
5 5 5 10 10 20	SI NO
5 5 5 10 20 10	SI NO
8 5 9 10 10 40 80 160 320	
4 2 4 12 6	
-1	

Se proporciona, además, algunos archivos con casos de prueba: 0.in contiene una primera batería de casos de prueba, y 0.out los resultados esperados, y 1.in contiene otra batería de casos de prueba (correspondiente a vectores muy grandes), y 1.out la salida esperada.

Trabajo a desarrollar

Tu trabajo consiste en implementar el algoritmo como cuerpo de la función `esta_ordenado`. Dicha función recibe el vector almacenado en las primeras `n` posiciones del array `a`. Deberá devolver `true` si el vector está ordenado crecientemente, y `false` en caso contrario.

El algoritmo debe ser eficiente. En particular, debería ser capaz de procesar, en un PC estándar, vectores del orden del millón de elementos en, a lo sumo, unos pocos segundos.