

### Control 3

#### Números sobrados

El objetivo de este control es desarrollar sistemáticamente un algoritmo iterativo para resolver un problema sobre estructura de números enteros.

#### El problema

Un número entero no negativo  $n$  se dice *sobrado* cuando la secuencia de sus dígitos (desde el menos significativo al más significativo) es estrictamente creciente. Por ejemplo, 4321 es un número *sobrado*, pero 3421 no lo es. Diseña un algoritmo que, dado un número  $n$ , decida si es sobrado o no.

#### Trabajo a realizar

Para realizar el control se proporciona un archivo `control3.cpp` que contiene un programa que lee por la entrada estándar: (i) el número  $k$  de casos que desean ejecutarse; (ii)  $k$  números enteros. Cada vez que lee uno de estos  $k$  números enteros, decide si es o no *sobrado* invocando a la función `es_sobrado`, e imprime el resultado (SI o NO) por pantalla.

A continuación, se muestra un ejemplo de entrada procesable por este programa, y de salida producida (suponiendo una implementación adecuada de `es_sobrado`):

Entrada	Salida
4	SI
4321	NO
3421	SI
54	NO
15	

Tu trabajo consiste en:

- Especificar formalmente el algoritmo que decide si el número  $n$  es o no *sobrado*, rellenando los huecos correspondientes en los comentarios que rodean a `es_sobrado`
- Diseñar **sistemáticamente** una implementación C++ para dicho algoritmo. Debes describir el diseño en el comentario habilitado a tal fin en el archivo proporcionado, siguiendo las reglas de diseño presentadas en clase.
- Completar la función `es_sobrado` a partir del código derivado.
- Entregar `control3.cpp` a través del juez en línea de la asignatura.

#### Importante:

- Únicamente se evaluarán aquellas entregas que superen satisfactoriamente los casos de prueba del juez.
- No modificar el código proporcionado. Únicamente deben responderse a los distintos apartados, en el interior de los comentarios, e implementar la función `es_sobrado`. Para escribir las fórmulas de la especificación en texto plano, pueden utilizarse los siguientes convenios:

Símbolo	Representación en texto plano
$\forall$	PARA_TODO
$\exists$	EXISTE
$\Sigma$	SUMA
$\Pi$	PROD

Por su parte, las conectivas lógicas y operadores de relación no expresables directamente en texto plano, pueden escribirse utilizando la notación de C++.

Ejemplo:

$$\forall i: 0 \leq i < n: (\exists j: 0 \leq j < n: (((\Pi k: 0 \leq k < j: a[k]) = (\Sigma r: j \leq r < i: a[r])) \vee a[j] = 0))$$

Escrita en texto plano:

$$\text{PARATODO } i: 0 \leq i < n: (\text{EXISTE } j: 0 \leq j < n: (((\text{PROD } k: 0 \leq k < j: a[k]) = (\text{SUMA } r: j \leq r < i: a[r])) \vee a[j] = 0))$$