

Control 2

Gaspariformes

El objetivo de este control es especificar un algoritmo para resolver un problema sencillo, y diseñar sistemáticamente una implementación a partir de dicha especificación.

El problema

Dado un vector **int** $a[n]$, con $n > 0$, se llama *suma parcial* a toda suma de la forma $a[0] + \dots + a[i]$, con $0 \leq i < n$. El vector se llama *gaspariforme* si todas sus sumas parciales son **positivas**, excepto la última, que debe ser 0. Diseña un algoritmo que, dados a y n , decida si a es o no *gaspariforme*.

Trabajo a realizar

Para realizar el control se proporciona un archivo `control2.cpp` que contiene un programa que lee por la entrada estándar vectores, y decide si dichos vectores son o no *gaspariformes* invocando a la función `es_gaspariforme`.

A continuación, se muestra un ejemplo de entrada procesable por este programa, y de salida producida (suponiendo una implementación adecuada de `es_gaspariforme`) (los vectores se introducen indicando, como primer valor, el número de elementos, y, a continuación, enumerando los distintos elementos; el final se indica indicando 0 como tamaño del vector):

Entrada	Salida
4 1 3 5 -9	SI
1 1	NO
1 0	SI
4 1 -3 5 3	NO
0	

Tu trabajo consiste en:

- Especificar formalmente el algoritmo que decide si el vector **int** $a[n]$ es o no *gaspariforme*, rellenando los huecos correspondientes en los comentarios que rodean a `es_gaspariforme`
- 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_gaspariforme` a partir del código derivado.
- Entregar `control2.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_gaspariforme`. 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
Σ	SUMA
Π	PROD

Por su parte, la 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:

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