

Control 4

Números k-emparejados

El objetivo de este control es desarrollar sistemáticamente un algoritmo iterativo para resolver eficientemente un problema.

El problema

Dos números se llaman *k-emparejados* cuando el valor absoluto de su diferencia es exactamente *k* (es decir, *u* y *v* están *k-emparejados* cuando $|u - v| = k$). Por ejemplo, 7 y 14 están *7-emparejados*. Observa que todo número está 0-emparejado consigo mismo.

Diseña un algoritmo eficiente que, dado un vector estrictamente creciente de enteros $\text{int } a[n]$ ($n \geq 0$) y un número $k \geq 0$, determine el número de parejas de números en *a* que están *k-emparejados*.

Trabajo a realizar

Para realizar el control se proporciona un archivo `control4.cpp` que contiene un programa que lee por la entrada estándar casos de prueba que consisten en dos líneas:

- En la primera aparecen el valor de *n* y el valor de *k*, por este orden
- En la segunda aparecen, en orden, todos los componentes del vector

La entrada termina con una línea que empieza por -1. Cada vez que lee un caso de prueba, el programa invoca a una función `num_k_emparejados` que determina el número de parejas de números *k-emparejados*, y escribe dicho número por la salida estándar.

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

Entrada	Salida
5 3	2
1 4 5 6 8	1
4 2	3
1 4 5 6	
3 0	
1 2 3	
-1	

Tu trabajo consiste en:

- Especificar formalmente el algoritmo que determina la cantidad de parejas de números *k-emparejados* en *a*, rellenando los huecos correspondientes en los comentarios que rodean a `num_k_emparejados`
- 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 `num_k_emparejados` a partir del código derivado.
- Entregar `control4.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 `num_k_emparejados`. 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:

PARATODO i: 0<=i<n: (EXISTE j: 0<=j<n:

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