

## AEDD - Guía Práctica 11: Búsqueda y Ordenamiento

Se solicita codificar una función en C++ para cada uno de los siguientes ejercicios. En todos los casos considerar los parámetros de las funciones y sus valores de retorno.

### Ejercicios propuestos:

1. En un arreglo de 1500 elementos se guardan números enteros ordenados ascendentemente. Defina la función `cantidadRepetidos()` que recibe el vector y retorna la cantidad de números que se repiten más de 10 veces. ✓✓
2. La función llamada `elimOrdenar` recibe como parámetros un arreglo A de 500 números enteros, el tamaño lógico TLA del arreglo y un valor entero X. El arreglo A está desordenado. La función deberá devolver el vector A actualizado de la siguiente forma: ✓✓
  - Se deben eliminar todos los valores que sean múltiplos de X.
  - El vector debe quedar ordenado ascendentemente.
  - La función deberá retornar la cantidad de elementos eliminados de A.
3. La función `Uno()` recibe dos arreglos de 500 números enteros (A y B), los tamaños lógicos de cada uno (TA y TB,  $TB \leq TA$ ) y un valor booleano (C). El arreglo A y el arreglo B están desordenados. La función `Uno()` debe retornar un vector V y su tamaño lógico TV, ordenado ascendentemente si  $C == \text{True}$  (descendentemente en otro caso) conteniendo solamente los valores de B que se encuentran en A, sin repeticiones. Los vectores A y B pueden contener elementos repetidos. ✓✓