

## Ejercicios Tema 5. Plantillas

**EJERCICIO 1.** La media aritmética se define como la suma de un conjunto de valores del mismo tipo dividido entre la cantidad de elementos en el conjunto. Cree una función tipo plantilla que permita encontrar la media aritmética de un conjunto de N elementos alojados en un arreglo con el siguiente prototipo:

```
template<typename T> T mediaAritmeticaElementos(T a[], int n);
```

Valide el ejercicio encontrando la media aritmética de 5 datos tipo int, 3 datos tipo float y 2 tipo double.

**EJERCICIO 2.** Diseñe e implemente una clase llamada Vector2D que permita operar con datos tipo float. La clase Vector2D debe permitir la suma entre objetos Vector2D y la división con enteros (escalares). Si la clase ha sido completamente desarrollada, pruebe la funcionalidad de encontrar la media aritmética de un arreglo de 5 objetos de la clase Vector2D.

**EJERCICIO 3.** La varianza se define como la suma de las diferencias de las desviaciones al cuadrado dividido entre la cantidad N de elementos. Esto significa:

$$\sigma^2 = \sum_{i=1}^n (x[i] - \bar{x}) \cdot (x[i] - \bar{x})$$

Donde  $\bar{x}$  es la media de los datos. Cree una función tipo plantilla que permita encontrar la varianza de un conjunto de N elementos alojados en un arreglo con la siguiente firma:

```
template<typename T> T varianzaElementos(T a[], T media, int n);
```

Valide el ejercicio encontrando la suma de 5 datos tipo int, 3 datos tipo float y 2 tipo double

**EJERCICIO 4.** Cree una función tipo plantilla que intercambie dos objetos de la misma clase, pero no modifique sus direcciones de memoria. La función no devuelve nada. Pruebe la función intercambiando 2 flotantes, 2 strings y por último 2 Vector2D.

**EJERCICIO 5.** Cree una función tipo plantilla que permita encontrar la dirección de memoria de 2 objetos de entrada, en principio de diferentes clases, y a través de punteros de entrada, devolver estas dos direcciones<sup>1</sup>. Usar 2 typename como plantillas y un total de 4 parámetros de entrada dos variables y dos punteros.

**EJERCICIO 6.** Defina 2 o más clases no relacionadas que contengan un mismo atributo estático llamado factor\_tolerancia. Crear una función tipo plantilla que imprima en pantalla el doble de un factor de tolerancia y, de ser posible, la mitad del mismo. Ejemplo: Defina una clase Entero, cuyo factor de tolerancia sea igual a 0.5, en la función plantilla el resultado debe ser 1 y 0.25.

**EJERCICIO 7.** Crear una función tipo plantilla que funcione para diferentes clases que tengan definida la función toString(). La función debe imprimir "Desde la función plantilla: " y a continuación

el `toString()` de la clase. Por ejemplo, cree dos clases diferentes A y B, y cree una función tipo plantilla que reciba el parámetro `template T` y utilice el `toString()`.

**Desafío.** Crear una función tipo plantilla en C++ que permita encontrar los N valores más pequeños de un arreglo de M elementos tipo T.