

Programación y Estructuras de Datos

## **SEMINARIO C++**

**Guía rápida**

# SEMINARIO - SESIÓN 2

## "C++ paso a paso" - Capítulo 3 : "Constructor y destructor"

- **3.1 Sobrecarga de funciones**

- Teclea y compila el **ejemplo 3.2** para ver cómo funciona la sobrecarga.
- Fíjate en la anotación de la pág. 33 y el **ejemplo 3.3** (líneas 5 y 10) : los tipos de salida son diferentes, pero la sobrecarga no es correcta, por igualdad de parámetros.
- Teclea y compila el **ejemplo 3.3** y comprueba resultado (error sobrecarga) .

- **3.2 Constructor**

- Fíjate en el comentario de los varios constructores gracias a la sobrecarga ( pág. 35 )

- **3.3 Constructor por defecto**

- Fíjate en el **ejemplo 3.4** , y en cómo la línea 3 añade el uso del constructor por defecto (pág.35)
- De la sesión anterior, tendrás ya tecleado el fichero de clase **tcoordenada.h** . Es el momento de añadirle la línea 3 del **ejemplo 3.4**, para declarar el constructor por defecto en tu clase (pág. 35).
- Igualmente, tendrás ya tecleado el fichero de métodos de clase **tcoordenada.cc** . Es el momento de añadirle las líneas del **ejemplo 3.5**, para implementar el constructor por defecto en tu clase (pág. 35).
- Teclea y compila el **ejemplo 3.6** y comprueba resultado del MAIN propuesto.

- **3.4 Otros constructores**

- Fíjate en el **ejemplo 3.7** , y en cómo la línea 4 añade el uso del constructor alternativo con parámetros (pág.36).
- Para el fichero de clase **tcoordenada.h** : es el momento de añadirle la línea 4 del **ejemplo 3.7**, para declarar el constructor con parámetros en tu clase (pág. 36).
- Para el fichero **tcoordenada.cc** : es el momento de añadirle las líneas del **ejemplo 3.8**, para implementar el constructor con parámetros en tu clase (pág. 36).
- Añade al último MAIN propuesto las líneas de la pág. 37, para probar a utilizar ambos constructores .

- **3.5 Constructor de Copia**

- Fíjate en las 3 formas de invocar a un CONSTRUCTOR (pág.37). Repara, concretamente, en la NOTA 4 de pie de página, donde se dice que el paso de parámetros por referencia (&) evita la invocación del constructor de copia innecesariamente.
- Para el fichero de clase **tcoordenada.h** : es el momento de añadirle la línea 5 del **ejemplo 3.9**, para declarar el constructor de copia en tu clase (pág. 39).
- Para el fichero **tcoordenada.cc** : es el momento de añadirle las líneas del **ejemplo 3.10**, para implementar el constructor de copia en tu clase (pág. 39).

- Añade al último MAIN propuesto las líneas correspondientes al objeto p3 del ejemplo 3.11, para probar el constructor de copia. Compila y prueba.
- Fíjate en tema del parámetro de entrada en el constructor de copia: la necesidad de que se pase por referencia y el peligro asociado de bucle infinito (pág. 41).
- Para el fichero de clase **tcoordenada.h** : retoca la línea 5 del **ejemplo 3.12**, para declarar el constructor de copia incorrectamente por valor (pág. 41).
- Para el fichero **tcoordenada.cc** : retoca la línea 1 del **ejemplo 3.13**, para declarar el constructor de copia incorrectamente por valor (pág. 41).
- Recompila el MAIN propuesto anteriormente para probar el constructor de copia. Compila y verifica los errores generados. Comprueba que el error auténtico es sólo el de la línea 2 (salida **ejemplo 3.13** , pág.42).

### • 3.6 Constructor en la parte Privada

- Fíjate en la posibilidad de traspasar el constructor de copia a la parte privada, y su posible utilidad (pág.42).
- Para el fichero de clase **tcoordenada.h** : es el momento de añadir la línea 17 del **ejemplo 3.14**, para declarar el constructor de copia en la parte privada clase (pág. 39). Hay que eliminarlo de la parte pública también.
- Recompila el MAIN propuesto anteriormente con este cambio. Comprueba el error (salida **ejemplo 3.14** , pág.43).

### • 3.7 Destructor

- Fíjate en el carácter único del destructor de clase (pág. 43).
- Para el fichero de clase **tcoordenada.h** : retoca con la línea 6 del **ejemplo 3.15**, para declarar el destructor (pág. 44).
- Para el fichero **tcoordenada.cc** : retoca con las líneas del **ejemplo 3.16**, para declarar el destructor (pág. 44).

### • 3.8 Forma canónica de una clase

- Revisa el **punto 7.1** del **Capítulo 7** del libro, donde se explica en profundidad.

### • 3.10 Ejercicios de PROGRAMACIÓN / 3.12 Respuesta a ejercicios de PROGRAMACIÓN

- El **ejemplo 3.10.3** propone la clase **TCalendario**.
- La **solución 3.12.2** resuelve inicialmente lo que se pide de la clase **TCalendario**.