



Libro/a:

Tema: 3

Fecha: / /

Programación III - Primer Parcial

Un cliente, requiere un sistema modelado e implementado observando los principios de la POO para registrar y administrar la venta de libros en una librería.

Los datos pertinentes a dicho objeto *Libro* son:

- *ISBN*: (*International Standard Book Number*) Número entero de 13 dígitos decimales.
- *Signatura Topográfica*: Cadena de caracteres. Código alfanumérico que indica la ubicación en una biblioteca de una obra.
- *Costo*: Número real (punto flotante de precisión simple).

Se pide:

A) Código fuente del archivo de clase *CLibro.cs* conteniendo:

- 1) Adecuada declaración de las variables miembro.
- 2) Un método *setSignaTopog(codigo)* que permita asignar el valor a la variable miembro *Signatura Topográfica*.
- 3) Un método constructor, que recibiendo como argumento un dato *numISBN* (número entero de 13 dígitos decimales), establezca valor inicial para la variable miembro *ISBN*.
- 4) Un método *setCosto(monto)*, que recibiendo como argumento un valor *monto*, número real (punto flotante de precisión simple), establezca valor para la variable miembro *Costo*.
- 5) Un método *darPrecio()*, que sin argumento alguno, devuelva como número real (punto flotante de precisión simple), el importe del *Precio* a abonar, producto de adicionarle a su *Costo*, la comisión del vendedor (15%).
- 6) Un método *darDatos()*, que sin argumento alguno, devuelva una cadena de caracteres que concatene los valores presentes en las variables miembro *ISBN*, *Signatura Topográfica*, *Costo* más el valor calculado de *Precio* a abonar por el libro (emplear método anterior).
- 7) Un método *esMasBaratoQue(otroLibro)*, que recibiendo como argumento, una referencia a otra instancia de *CLibro*, devuelva como dato lógico, el valor *true*, cuando la instancia invocante abone un *Precio* menor que la recibida como argumento y, el valor *false*, en cualquier otro caso.
- 8) Un método sobrecargado *darPrecio(descuento)*, que recibiendo como argumento un valor de *descuento*, número real (punto flotante de precisión simple del intervalo [0.0 a 100.0]), devuelva como número real (punto flotante de precisión simple), el *Precio* a abonar por el libro, aplicándole el descuento indicado (obviamente, sólo si *descuento* es positivo).

B) Código fuente del archivo de clase *CEjecutora.cs* conteniendo el método *main()* de una aplicación de consola, que permita:

- 1) Solicitar iterativamente y registrar, los datos de un conjunto de extensión desconocida de libros; finalizando dicho proceso de carga con el ingreso de un número de *ISBN* de valor igual a 0 (cero), para cuyo caso no deberán solicitarse ni registrarse los datos restantes *Signatura Topográfica* y *Costo*.
- 2) Informar *ISBN – Signatura Topográfica – Costo – Precio* del libro más costoso ingresado (último de los ingresados en caso de existir más de uno), como así también, el *Precio* con el descuento a estudiantes (40%) aplicado. Emplear el método del apartado 1-7.
- 3) En caso de no haberse ingresado libro válido alguno, en lugar de lo indicado en el ítem anterior, informar la leyenda “No se ingresaron libros.”.

Criterios:

Se evaluará sólo lo presentado por escrito, en tinta y sin tachaduras ni correcciones de ningún tipo.

- Adicionalmente, se calificará:
 - La eficacia en la funcionalidad de la aplicación.
 - Las técnicas, metodologías y estrategias de programación aplicadas.
 - La prolijidad general y la claridad conceptual en el desarrollo.
 - La eficiencia en el empleo y la gestión de los recursos de memoria.
 - La eficiencia en el empleo de estructuras de control y gestión del flujo de ejecución.
 - La correcta promoción de tipos, que debe ser estricta y explícita.
 - La observancia de los principios de la P.O.O. estudiados.
 - El cumplimiento de los tiempos y pautas de examinación.
 - Importante: No se requiere validación de los datos de entrada.