

Programación 4

CURSO 2018 – PRÁCTICO 4

Parte 1: Generación parcial de código

Ejercicio 1 (básico, imprescindible)

- a) Generar un esqueleto del código de las clases a partir de la estructura del Modelo de Dominio del ejercicio 4 del Práctico 3.
- b) Generar un esqueleto del método `main()` a partir del comportamiento contenido en el Diagrama de Secuencia del Sistema de la solución de la parte a) de dicho ejercicio.

Ejercicio 2 (básico, imprescindible)

- a) Generar un esqueleto del código de las clases a partir de la estructura del Modelo de Dominio del ejercicio 5 del Práctico 3.
- b) Generar un esqueleto del método `main()` a partir del comportamiento contenido en el Diagrama de Secuencia del Sistema de la solución de la parte a) de dicho ejercicio.
- c) Discutir según diferentes alternativas de Diagramas de Secuencia del Sistema para el caso de uso descrito en la parte a) de dicho ejercicio.

Ejercicio 3 (avanzado, imprescindible)

- a) Generar un esqueleto del código de las clases a partir de la estructura del Modelo de Dominio de la solución del ejercicio 14 del Práctico 2.
- b) Generar un esqueleto del método `main()` a partir del comportamiento contenido en los Diagramas de Secuencia del Sistema de la solución del ejercicio 8 del Práctico 3.

Parte 2: Conceptos básicos de Orientación a Objetos

Ejercicio 1 (medio, imprescindible)

- a) Definir y explicar la utilidad de los conceptos interfaz y realización.
- b) Identificar clases, interfaces y realizaciones en las siguientes realidades.
 - i) Un estudio de mercado realizado por una marca de lavarropas determinó la construcción de un modelo del mencionado electrodoméstico, el cual debe tener un conjunto de controles determinado C. La compañía fabricante decidió construir una cantidad de lavarropas, de los cuales una cierta cantidad son fabricados con componentes de procedencia P1 y el resto con componentes de procedencia P2. A la vista de los usuarios finales, todos los lavarropas son iguales.
 - ii) Durante el desarrollo de un determinado software, se ha detectado la necesidad de contar con un módulo de manejo de colecciones de elementos. Las operaciones necesarias son: crear, agregar, eliminar, chequeo de vacío, buscar un elemento y operaciones de iteración sobre la colección. Se cuenta con tres implementaciones eficientes de estructuras de datos para colección: lista enlazada, árbol binario y tabla de dispersión. El módulo pretende poder manejar todos los tipos de colecciones sin importar realmente su implementación.
- c) ¿Qué pasaría si no contáramos con los conceptos interfaz y realización en las realidades planteadas en la parte b?