



Orientación a Objetos 2 – Práctica 5

Ejercicio 1: Lea el pattern Template Method y ...

(Puede leer el patrón desde el archivo `templateMethod.pdf`) .

1. Responda:

- ¿Dónde se define el esqueleto del algoritmo?
- ¿Se puede redefinir el esqueleto?
- ¿Qué es lo que se puede redefinir?
- ¿Qué es un hook method?

2. Busque un ejemplo de aplicación del pattern en:

- Clase `#Magnitude` categoría `#comparing`
- Clase `#Collection` categoría `#accessing`

3. Busque dos ejemplos más de uso del patrón en Smalltalk.

4. Para cada uno de los cuatro ejemplos de los dos ejercicios anteriores, realice un diagrama de clase indicando:

- La clase abstracta.
- El template method.
- Al menos 2 clases concretas.
- Los hook methods.

Ejercicio 2: Template hasta en la sopa (o el café)

Mucha gente no puede vivir sin su café o su té matinal, ni hablar del chocolatada de la tarde. El proceso de preparación de cada uno es:

- ¿Cómo preparar un café? Calentar agua, poner café en la taza, agregar azúcar y leche, verter el agua en una taza.
- ¿Cómo preparar un té? Calentar agua, poner el saquito de té en la taza, agregar azúcar y limón, verter el agua en una taza
- ¿Cómo preparar una chocolatada? Calentar la leche, poner cacao en la taza, agregar azúcar, verter la leche en una taza.

1. Defina 3 clases: Café, Té y Chocolatada e implemente en cada una de ellas el método `#preparar`.

2. Vea que el proceso de preparación es muy común para todas. Cree una clase abstracta de la cual extienden las 3 clases definidas anteriormente y factorice en la clase abstracta todo el código que le sea posible.

Ejercicio 3: Fábrica de vehículos

Independientemente de si el vehículo va por aire, agua o tierra, posee básicamente la misma línea de construcción.

- Auto: a un chasis se le incorpora una planta motriz y una dirección.
- Avión: a un fuselaje se le incorporan turbinas y alerones.
- Velero: a un casco se le incorporan los mástiles de las velas y un timón

1. Defina 3 clases: Auto, Avión y Velero e implemente en cada una de ellas el método `#construir`, factorizando en la clase Vehículo el código que sea posible.
2. Indique que otros elementos forman parte de estos vehículos. Por ejemplo: freno, freno de estacionamiento, anclas, etc.
3. Extienda el la solución para incorporar los elementos indicados en el punto anterior.

Ejercicio 4: Juego de Combate con Tácticas

Retomando el ejercicio del Juego de Combate, ahora se desea dotar a los jugadores de tácticas de combate. Las tácticas disponibles son:

- Agresiva: en donde se utilizará un 80 % de golpes y un 20 % de bloqueos elegidos de forma aleatoria.
- Random: en donde la elección del movimiento será aleatorio con la misma probabilidad para los golpes y los bloqueos.
- Defensiva: en donde la cantidad de bloqueos será de 80 % y la cantidad de golpes de 20, elegidos de forma aleatoria. %.
- Golpe-Bloqueo: en esta estrategia el jugador comienza con un golpe y continua con un bloqueo, repitiendo siempre esta secuencia.

Un jugador utiliza una táctica durante todo un combate, pero puede cambiar de táctica entre combates. Por ejemplo, si un jugador pierde al utilizar la táctica random, puede decidir usar la defensiva.

1. Realice un diagrama de clases.
2. Implemente en Smalltalk.
3. Ejecute 100 corridas de todas las tácticas contra el resto y averigüe qué táctica es la más exitosa.

Ejercicio 5: El Encriptador

En un sistema de mensajes instantáneos (como Gtalk), se envían mensajes de una máquina a otra a través de una red. Para asegurar que la información que pasa por la red no es espiada, el sistema utiliza una *conexión segura*. Este tipo de conexión encripta la información antes de enviarla y la desencripta al recibirla. La siguiente figura ilustra un posible diseño para este enunciado.

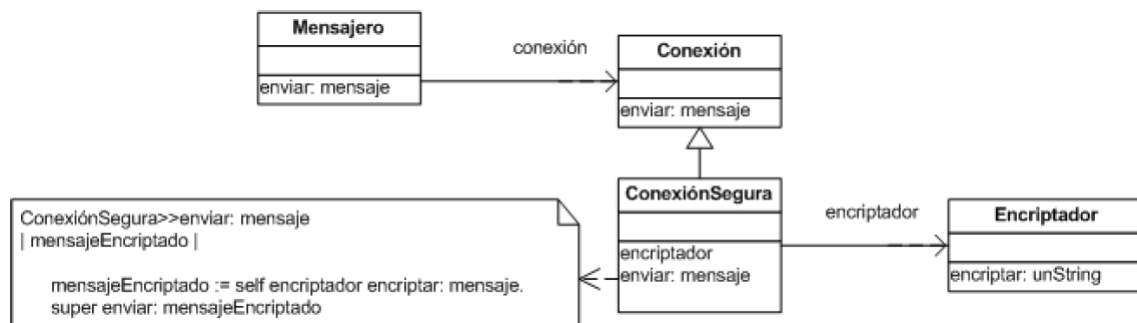


Figura 1: Diseño del encriptador

El encriptador utiliza el algoritmo RSA, pero se desea agregar otros algoritmos (diferentes algoritmos ofrecen distintos niveles de seguridad, overhead en la transmisión, etc).

1. Modifique el diseño para que el objeto Encriptador pueda encriptar mensajes usando los algoritmos Blowfish y RC4, además del ya soportado RSA.
2. Documente mediante un diagrama de clases UML indicando los roles de cada clase.