

Programación Orientada a Objetos

Práctica N° 4 – 2021

Para cada una de los siguientes problemas plantee una solución identificando los **objetos que intervienen**, su **estado**, su **comportamiento** y las **relaciones** entre ellos.

1. Definir las clases e implementarlas para un sistema de alquiler de videos.
2. Definir las clases e implementarlas para un sistema de alquiler de autos.

3. Sistema de siembra

Una cooperativa de agricultores requiere un sistema que le aconseje cuáles son los cereales que puede sembrar en un determinado lote y qué lotes pueden ser utilizados para sembrar un determinado cereal. Para poder sembrar un cereal en un lote, éste debe contener determinados minerales. Los cereales se clasifican en granos de cosecha gruesa (como girasol, maíz,...), granos de cosecha fina (como trigo, avena,..) y pasturas (como alfalfa, trébol subterráneo,..). Un lote satisface los requerimientos de un cereal si contiene todos los minerales que requiere ese cereal. Para el caso de las pasturas, poseen la restricción adicional de no poder sembrarse en lotes menores a una superficie de 50 hectáreas.

Adicionalmente, los lotes son clasificados como “especiales” cuando contienen ciertos minerales de interés primario para la cooperativa, y “comunes” cuando contienen sólo minerales secundarios en la composición de la tierra. La cooperativa debe poder consultar si un lote es común o especial.

4. Congreso de Ciencias de la Computación

El comité organizador de un Congreso de Ciencias de la Computación necesita un sistema que le ayude a asignar evaluadores de los trabajos enviados al congreso. Para esto cuenta con un conjunto de evaluadores, calificados como “generales” cuando conocen temas generales como algoritmos, lenguajes de programación, etc., o como evaluadores “expertos” cuando conocen temas específicos como agentes, visualización, redes de comunicación, etc. El sistema debe designar a un evaluador para cada uno de los distintos trabajos enviados al congreso, existiendo diferentes categorías de trabajos: artículo, resumen y poster. Cada trabajo posee un conjunto de palabras claves que definen los temas tratados en el trabajo.

Para asignar un evaluador a un trabajo, el sistema debe asegurarse que el evaluador es apto para evaluar el trabajo. Para determinar si un evaluador es apto para evaluar un trabajo, éste debe haber declarado conocimiento en todos los temas del trabajo, especificados en sus palabras claves. Para el caso de los posters, basta que el evaluador tenga conocimiento en uno de los temas del poster o sea en una de las palabras clave.

Implementar una solución orientada a objetos para este problema que permita determinar 1) qué evaluador puede asignarse a un determinado trabajo, 2) qué trabajos pueden asignarse un evaluador particular, y 3) cuántos trabajos tiene un evaluador determinado y 4) determinar si un evaluador dado es general o experto.

Programación Orientada a Objetos

Práctica N° 4 – 2021

5. Abstraer el proceso de alquiler a partir de los ejercicios 1 y 2 de esta práctica, definiendo e implementando las clases y los métodos necesarios. Implementar cómo quedan ambos ejercicios en función de la abstracción realizada.
6. Abstraer el comportamiento común de los ejercicios 3 y 4 de esta práctica, definiendo e implementando las clases y los métodos necesarios. Implementar cómo quedan ambos ejercicios en función de la abstracción realizada.

7. Cadena de fábricas de muebles

Definir las clases e implementarlas para el siguiente sistema de una cadena de fábricas de muebles. Cada fábrica recibe pedidos de muebles, los cuales especifican el mueble solicitado, el costo estimado, el material con el que se confeccionarán (melamina, roble, pino, etc.), una fecha máxima para su entrega, el lugar a donde debe ser entregado y la fecha en la cual se realizó el pedido. Al recibir un pedido, una fábrica puede decidir si lo va a realizar o no de acuerdo a diferentes políticas. Por ejemplo, una fábrica puede decidir atender sólo pedidos cuyo material sea “pino”. Las fábricas tienen un convenio de cooperación, mediante el cual pueden atender pedidos delegados por alguna de las otras fábricas. Por lo tanto, para los pedidos que la fábrica decide no atender, puede delegarlos a otra fábrica, si cumple con determinada política de delegación, o simplemente rechazarlos.

Por otro lado, al momento de decidir qué mueble realizar primero, cada fábrica utiliza un conjunto de criterios propios; por ejemplo: tratar primero los muebles más caros, los que tienen una fecha de entrega más próxima, etc.

8. Pila

Implementar en Java una colección para almacenar una pila de elementos. A esta colección se le pueden agregar elementos utilizando el método `push(Object o)`. Para retirar elementos de la colección se utiliza el método `pop()`, que retorna el último elemento agregado a la colección y lo elimina de la misma. Es posible consultar el tope de la pila sin eliminarlo utilizando el método `top()`. La mencionada anteriormente es la única forma de consultar y retirar elementos de la colección, es decir, no se pueden obtener ni consultar elementos de otra posición que no sea el tope de la pila. Definir también un método para conocer el tamaño de la pila.

9. Mazo de cartas

Implementar un juego de cartas en el cual se dispone de un mazo y se reparten, en cada ronda, dos cartas por jugador. Un jugador gana la mano cuando la suma de sus cartas es mayor que la de su oponente. El jugador que gana coloca todas las cartas (las suyas y las del oponente) en su pila de cartas ganadas. En caso de empate, se reparte una carta más a cada jugador hasta que uno de los dos gane. Implementar los mazos como pilas de cartas.