

Auxiliar I

METODOLOGÍAS DE DISEÑO Y PROGRAMACIÓN CC3002 @ 2009

Ejercicio 1

Analizar las siguientes frases que buscan definir el concepto de clase, detectando semejanzas y errores respecto a la definición correcta.

- a) Una clase define atributos y operaciones de los objetos que son instancia de ellos, es una plantilla a partir de la cual se crean objetos.
- b) Es un concepto que tiene ciertas características, por ejemplo atributos.
- c) La clase es la encargada de describir cómo son los objetos de la misma, es el análogo entre tipo y archivo, que además de describir el formato de estos datos también describe el comportamiento mediante las operaciones y las relaciones con otras clases mediante asociaciones y generalización.
- d) Descriptor de conceptos del mundo real.
- e) Es un descriptor de elementos que tienen mismos atributos, operaciones y asociaciones.

Ejercicio 2

- a) ¿Un datatype es un objeto? ¿y un datavalue? Justificar.
- b) ¿Un datavalue es el valor de un atributo? Justificar.

Ejercicio 3

Analizar el significado de la asignación y del pasaje de parámetros por valor teniendo en cuenta la propiedad de identidad de los objetos.

Ejercicio 4

Considerar definidas las clases A, B y C, las asociaciones R1 entre las clases A y B, y R2 entre las clases B y C. Sean los objetos a, b y c, de clases A, B y C respectivamente. ¿Es posible que existan los siguientes links?

- a) Un link entre a y b
- b) Un link entre b y c
- c) Un link entre a y c
- d) Dos links entre b y c

Ejercicio 5

Considerar definidas las clases A, B, C, D y E, que cumplen $B <: A$, $C <: A$, $D <: A$ y $E <: B$, siendo sus segment descriptors:

SD_A	SD_B	SD_C	SD_D	SD_E
ATT_{at1}	OP_{op3}	$MET_{C::op1}$	ATT_{at2}	$MET_{E::op1}$
OP_{op1}	$MET_{B::op3}$		OP_{op4}	$MET_{E::op2}$
OP_{op2}			$MET_{D::op1}$	
$MET_{A::op2}$			$MET_{D::op4}$	

- Indicar:
 - Clases bases directas de B
 - Clases bases directas de E
 - Subclases indirectas de A
 - Subclases directas de B
 - Clases abstractas
- Definir la propiedad subsumption y explicar informalmente su significado.
- Construir los full descriptors para cada una de las clases.
- Dado un objeto de la clase E ¿de qué clases es instancia directa y de cuáles es instancia indirecta?

Ejercicio 6

Considerar definidas las clases D, E y F, que cumplen $F <: E$, y una operación global:

`void g(E param)`

Considerar además que se cuenta con los objetos d, e y f, instancias directas de las clases D, E y F respectivamente.

Indicar cuáles de las siguientes invocaciones son válidas. Justificar.

- `g(d)`
- `g(e)`
- `g(f)`

Ejercicio 7

- ¿Los términos método y operación denotan el mismo concepto? Justificar.
- Identificar clases, atributos, operaciones y métodos en la siguiente descripción del diseño de una parte de un sistema de gestión de personal.

Se considera una empresa que mantiene información de sus empleados. De cada empleado se almacena su nombre y se cuenta con una operación para obtener el mismo. Los empleados de la empresa pueden ser fijos, de los cuales se conoce su sueldo, o jornaleros, de los cuales se conoce la cantidad de horas que trabajó y el valor de la hora. Todos los meses, al realizarse la liquidación de los sueldos, la empresa calcula el monto total por concepto de sueldos, en base al sueldo para empleados fijos y al producto de la cantidad de horas por el valor de la hora para empleados jornaleros.

- c) ¿Existe alguna clase abstracta en su solución? ¿En qué situaciones debemos definir a una clase como abstracta?
- d) Dar una descripción (pseudocódigo) de la operación `getTotal()` mediante la cual la empresa calcula el monto total de la liquidación de todos sus empleados. Tener en cuenta que en un futuro pueden incorporarse nuevos tipos de empleados, y que la repercusión en la implementación existente debe ser mínima (en particular sobre las operaciones ya implementadas en clases existentes).
- e) Definir el concepto de polimorfismo e identificar su aplicación en el modelo realizado en la parte (b). ¿Qué beneficios le aportó el polimorfismo y cuál otra propiedad le fue de utilidad?

Ejercicio 8

- 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) Usualmente la salida de audio analógica de un DVD se conecta a la entrada de audio de una TV. Sin embargo, alternativamente es posible realizar dicha conexión a la entrada de audio analógica de un home theater. Ambos aparatos procesan la señal de audio proveniente del DVD de diferente forma, pero la señal emitida por el DVD es la misma independientemente del aparato que la procese.
 - iii) Un equipo de audio cuenta con salidas de sonido, a las cuales se accede a través de conexiones que utilizan el tipo de ficha F1. De esta forma se pueden conectar parlantes de tipo P1 o P2 utilizando el tipo de ficha F1.
 - iv) Durante el desarrollo de un determinado software, se ha detectado la necesidad de contar con el manejo de colecciones de elementos. Las operaciones necesarias son: crear una colección, agregar un elemento, eliminar un elemento, chequear si está vacía, buscar un elemento y operaciones de iteración sobre los elementos de la colección. Se cuenta con tres estructuras de datos para colección: lista enlazada, árbol binario y tabla de dispersión. Se pretende que a quien utilice una colección le resulte transparente la estructura de datos subyacente.
- c) ¿Qué pasaría si no contáramos con los conceptos interfaz y realización en las cuatro realidades planteadas?

Ejercicio 9

Considerando el siguiente programa:

```
1  class A {...}
2  class B subclass of A {...}
3  void main() {
4      A b = new B();
5      b.op();
6  }
```

- a) En tiempo de compilación, ¿cómo procede el compilador para determinar si la invocación de la línea 5 es válida?
- b) En tiempo de ejecución, ¿cómo procede el runtime para determinar el método a despachar como consecuencia de la invocación de la línea 5?