

Programación orientada a objetos. Segundo parcial 1999.

Lunes 7 de junio de 1999.

1. Supongan que entre las instancias de una clase `Alpha` y las de otra clase `Beta` existe una relación de agregación. ¿Cómo implementarían en Smalltalk ambas clases para que funcionara la agregación? ¿Cómo las implementarían si la relación fuera una composición en lugar de una agregación? Justifiquen debidamente sus respuestas.
2. Supongan que existe en Smalltalk una clase `Gamma` a la que hay que agregar un nuevo método `greek`. El programador Armando Lío propone tres soluciones:
 - a. agregar el método `greek` a la clase `Gamma`;
 - b. crear una subclase de `Gamma` llamada `Epsilon` y agregar a la clase `Epsilon` el método `greek`;
 - c. crear una subclase `Delta` y otra `Omega` ambas con el método `greek`; mientras en `Delta` se delega la ejecución a una instancia de `Omega`, en `Omega` está programado el método.

Analice cada solución del Sr. Lío suponiendo que todas son posibles y pueden tener sentido. ¿Cuál elegirían ustedes? ¿Por qué? ¿Bajo qué condiciones o supuestos pueden llegar a elegir otra? Programen las soluciones propuestas por el Sr. Lío en Smalltalk sintiendo en cuenta el contenido del método `greek`.
3. Se suele afirmar que en Smalltalk todo es un objeto. La clase de un objeto es también un objeto de la clase `Behavior`. Los mensajes que se envían a un objeto se llaman métodos de instancia y los que se envían a la clase de un objeto se llaman métodos de clase. Pero los métodos de clase de cualquier clase son métodos de instancia de la clase `Behavior`. Supongan que existen los siguientes métodos de Smalltalk:

Selector	Clase	Descripción
<code>class</code>	<code>Object</code>	Retorna la clase del receptor.
<code>subclass</code>	<code>Behavior</code>	Retorna la superclase del receptor.
<code>sourceCodeAt: aSymbol</code>	<code>Behavior</code>	Retorna una <code>String</code> con el código fuente del método llamado <code>aSymbol</code> en el receptor o <code>nil</code> si el receptor no implementa el método.

Programen en Smalltalk un método de instancia al que puedan responder todos los objetos, que retorne el código fuente de un nombre de un método. Elijan ustedes la clase en la que implementarlo y la signatura del método.