



POLIMORFISMO Y REUTILIZACIÓN

PARADIGMAS DE PROGRAMACIÓN II

Carlos Rojas Sánchez

Licenciatura en Informática

Universidad del Mar

1. Concepto del polimorfismo

Concepto del polimorfismo

Concepto del polimorfismo

El polimorfismo es el modo en que los lenguajes OO implementan el concepto de polisemia del mundo real: Un único nombre para muchos significados, según el contexto.

Concepto del polimorfismo

- Capacidad de una entidad de referenciar distintos elementos en distintos instantes de tiempo.
- El polimorfismo nos permite programar de manera general en lugar de programar de manera específica.

Concepto del polimorfismo

Hay cuatro técnicas, cada una de las cuales permite una forma distinta de reutilización de software, que facilita a su vez el desarrollo rápido, la confianza y la facilidad de uso y mantenimiento.

- Sobrecarga
- Sobreescritura
- Variables polimórficas
- Genericidad

Sobrecarga (Overloading, Polimorfismo ad-hoc): un solo nombre de método y muchas implementaciones distintas. Las funciones sobrecargadas normalmente se distinguen en tiempo de compilación por tener distintos parámetros de entrada y/o salida.

Sobreescritura (Overriding, Polimorfismo de inclusión): Tipo especial de sobrecarga que ocurre dentro de relaciones de herencia. En este caso la signatura es la misma (refinamiento o reemplazo del método del padre) pero los métodos se encuentran en dos clases distintas relacionadas mediante herencia.

Variables polimórficas (Polimorfismo de asignación): variable que se declara como de un tipo pero que referencia en realidad un valor de un tipo distinto. Cuando una variable polimórfica se utiliza como argumento, la función resultante se dice que exhibe un polimorfismo puro.

Genericidad (plantillas o templates): forma de crear herramientas de propósito general (clases, métodos) y especializarlas para situaciones específicas.

Tipos de polimorfismo

- Sobrecarga

Factura::imprimir()

Factura::imprimir(int numCopias)

ListaCompra::imprimir()

- Sobreescritura

Cuenta::abonarInteres()

CuentaJoven::abonarInteres()

- Variables polimórficas

Cuenta *pc=new CuentaJoven();

- Genericidad

Lista<Cliente> Lista<Articulo> Lista<Alumno>

Lista<Habitacion>

Sobrecarga (Overloading, polimorfismo ad-hoc)

- Un mismo nombre de mensaje asociado a varias implementaciones.
- La sobrecarga se realiza en tiempo de compilación (enlace estático) en función de la signature completa del mensaje.

Sobrecarga (Overloading, polimorfismo ad-hoc)

Dos tipos de sobrecarga:

1. Basada en ámbito: Métodos con diferentes ámbitos de definición, independientemente de sus signatures de tipo. Permitido en todos los lenguajes OO.
 - Un mismo método puede ser utilizado en dos o más clases.
 - P. ej. Sobrecarga de operadores como funciones miembro.

Sobrecarga (Overloading, polimorfismo ad-hoc)

Dos tipos de sobrecarga:

2. Basada en signatura: Métodos con diferentes signaturas de tipo en el mismo ámbito de definición. No permitido en todos los lenguajes OO.
 - P. ej. Cualquier conjunto de funciones no miembro (en el ámbito de definición global) que comparten nombre.
 - Dos o más métodos en la misma clase pueden tener el mismo nombre siempre que tengan distinta signatura de tipos.

Sobrecarga basada en ámbito

- Distintos ámbitos implican que el mismo nombre de método puede aparecer en ellos sin ambigüedad ni pérdida de precisión.
- La sobrecarga por ámbito no requiere que las funciones asociadas con un nombre sobrecargado tengan ninguna similitud semántica, ni la misma signatura de tipo.