

Estudiantes

<12-11006> - <Gabriel Gimenez>

<11-10103> - <Guillermo Betancourt>

Respuestas

Los lenguajes de programación orientados a objetos que poseen herencia simple están limitados a incorporar comportamientos de un solo ancestro al momento de definir una clase.

Si consideramos que la herencia multinivel disponible, pues podríamos tener algo del estilo $A \rightarrow B \rightarrow C$ donde A y B son ancestros de C, y C obteniendo comportamiento de 2 ancestros, por lo que la respuesta sería No.

Si por lo contrario consideramos solo 1 nivel en la herencia simple, lo mas que podríamos tener, sería una clase con interfaces asociadas, por lo tanto la clase que hereda estaría limitada a incorporar comportamientos de un solo ancestro, pero pudiendo incorporar metodos, no implementados de sus interfaces.

Lenguajes de POO con un sistemas de tipos estático (C++, Java, C#) no tienen la posibilidades de elegir la implementación de un método a tiempo de ejecución (despacho dinámico).

No, Java es un lenguaje con un sistema de tipos estáticos, y podemos tener tanto despacho dinámico como despacho estático.

La introspección y reflexibilidad son conceptos que se manejan en la POO pero no guardan ninguna relación entre sí.

Si, no todo lenguaje con introspeccion y reflexividad debe ser OO

En un lenguaje con un sistema de tipos dinámico la sobrecarga de métodos es innata y representa una comodidad dado que permite implementar un mismo método para distintos tipos.

Si, es innata cuando se refiere a los tipos de los parametros, pero no respecto a la aridad de las funciones que tambien podria ser sobrecargada.

En los lenguajes POO existen los términos interfaz, módulo, clase abstracta, rol, etc; definidos como objetos que pueden encapsular definiciones de clases o implementaciones concretas de métodos.

No, Pueden encapsular definiciones de clases, mas no implementaciones concretas de metodos.

*Los métodos virtuales permiten asociar, al momento de compilar, una implementación de un método sobrecargado con una llamada del mismo; eliminando el **overhead** del despacho dinámico.*

No, las llamadas de los metodos sobrecargados se resuelven a tiempo de ejecucion.

Cuando un lenguaje de POO tiene herencia simple no ocurre el problema del diamante pero de igual forma pueden existir llamadas ambiguas de métodos, dado que incorporar interfaces, módulos, protocolos, etc, no evita colisión de nombres.

No, las interfaces estan justamente para cumplirlas, y no poseen implementacion concreta, por lo que no hay ambigüedad, siempre se llama a la función de la clase.

El paso de mensaje es un término que se maneja en modelos concurrentes, también de POO y es equivalente a la llamada de una función.

Si, equivale a la llamada a un metodo de una clase desde el exterior.

Sin importar la herencia del lenguaje de POO, una clase podría tener más de un ancestro.

No, por lo expresado en la respuesta 1, y no considerando las interfaces como ancestros.