Trabajo Practico N ° 2-5

Paradigmas de Programación

**Trabajo Practico 4**

**Ejercicio 30.**

Responda y justifique cada una de las preguntas en caso de ser necesario:

1. ¿Cuál la clave en el proceso de análisis y diseño orientado a objetos, ya que mediante ella podemos llegar a armar un conjunto de clases?

2. ¿Cómo definiría al polimorfismo según sus palabras?

Poli (mucahas) Morfismo (formas) ,

3. ¿Concepto que se define como la declaración de un objeto?

4. ¿Cuál es el pilar de la POO que consiste en unir en una clase las características y comportamientos abstraídos de un objeto?

Encapsulamiento: es conjunto donde los métodos y atributos pueden ser privados o públicos protegiendo la manipulación de la información que no permite la modificación de la misma fuera de la clases protegiendo así los datos

5. ¿Encapsulamiento: Significa reunir a todos los elementos que pueden considerarse pertenecientes a una misma entidad, al mismo nivel de abstracción? ¿Por qué?

6. ¿Cuál es el pilar de la POO que consiste en captar las características, comportamientos e identidad que distinguen a un objeto?

Abstracción: mediante la abstracción podemos sacar de un objeto los métodos y atributos restringiendo lo necesario para la clase restringiendo lo necesario para el software

7. ¿Cuál es el pilar de la POO que solamente aplica sobre los métodos y que se refiere a que el mismo método puede ser usado para diferentes fines según se necesite?

Herencia:

**Ejercicio 31.**

Describa cómo los lenguajes orientados a objetos PUROS brindan polimorfismo y son a la vez fuertemente tipados.

**Ejercicio 32.**

Muestre algún ejemplo en donde la clasificación con herencia múltiple permita un modelado más natural que la lograda con herencia simple. Investigue acerca de cuales son los lenguajes que soportan herencia múltiple. Analice bajo qué circunstancias la herencia múltiple puede presentar problemas. ¿Qué mecanismos pueden implementarse para evitarlos? Describir cada uno de ellos.

**Ejercicio 33.**

**Ejercicio 34.**

Defina y diferencie (si existe) la herencia de clases con interfaces en el lenguaje Java. ¿Podríamos tener solo herencia de clases como forma de herencia múltiple?

**Ejercicio 35.**

1. Explique las diferencias entre sobreescritura (Overriding) y sobrecarga (overloading). ¿Qué conceptos trabajan en cada uno de ellos o están relacionados?

2. Que concepto visto en clase nos ayuda a solucionar en parte trabajar con sobrecarga de métodos.

**Trabajo Practico 5**

**Ejercicio 8.**

Explique qué restricciones debería tener un lenguaje orientado a objetos para que su chequeo de tipo pueda realizarse estáticamente. ¿Qué desventajas tiene aplicar tales restricciones?

**Ejercicio 9.**

¿Qué diferencias ve entre el sistema de tipos de Java y el C#? Realice un análisis comparativo entre estos lenguajes utilizando los criterios y/o características vistos en la materia.

**Ejercicio 10.**

Explique porque en los lenguajes orientados a objetos es necesario que la ligadura entre un mensaje (llamada) y el método correspondiente se resuelva dinámicamente.

**Ejercicio 11.**

¿Qué consideraciones deberían tomarse para modelar adecuadamente un lenguaje de programación orientado a objetos que tiene tipado dinámico como Python?