**Modelos Finales - Introducción a Objetos**

///////////////////////////////////////////////////////////////////////////////////////////////////////

FINAL 1

1 - Definir variable de instancia. Ejemplificar.

- Definir variable de clase. Ejemplificar.

- Tipos de métodos, desarrollar.

2 - Modelado de patrones. Modele UML del patrón Observer y dé un ejemplo (que no esté en las teorías y prácticas) de su posible uso, explicando cómo hará el patrón para resolver el problema.

3 - Colecciones - Características, comportamiento, tipos de colecciones. Explique tres operaciones que tengan en común todas las colecciones.

4 - Proponga un enunciado creativo que contenga 6 clases como mínimo y por cada clase 4 atributos y 4 responsabilidades o métodos. Tener en cuenta el uso de herencia simple, composición y también redefinir un método de una super clase.

///////////////////////////////////////////////////////////////////////////////////////////////////////

FINAL 2

Fue un final accesible. Tomó lo siguiente, parafraseando:

- Pseudovariables. Constantes y no constantes.

- Definir herencia simple y múltiple.

- Definir patrones de diseño. Enumerar 3. Dar un ejemplo de uso de cada uno. Y dar 5 características que deben cumplir los patrones de diseño.

- No tomaron código (increíble). Había un último ejercicio que decía: Desarrolle un enunciado de ejercicio para resolver modelando con UML. Debe tener 6 clases mínimo, con 4 métodos únicos y 4 atributos únicos cada uno. O sea, tenías que armar un enunciadito sobre el ejercicio, y después implementarlo en UML.

Se tomaron preguntas de finales que ya se venían tomando. Y, a mi entender, se simplificó un poco la complejidad.

Quizás alguna más que me esté olvidando. El ejercicio del UML / Enunciado, obviamente era el que más puntos valía.

///////////////////////////////////////////////////////////////////////////////////////////////////////

FINAL 3

Teoría:

- ¿Qué es un programa orientado a objetos?

- ¿Qué elementos hay en un programa orientado a objetos?

- Características de la programación orientada a objetos.

- Herencia. ¿Qué relación define la herencia entre objetos? ¿Qué tipos de herencia existen? Ventajas y desventajas.

- Pseudovariables constantes y no constantes, ejemplos.

- Definir patrones. Dar 5 características que deben cumplir. Desarrollar 3 patrones.

Práctica:

Una empresa tiene que liquidar sueldos. La empresa tiene un máximo de vendedores en su plantel. Los vendedores pueden ser fijos o a comisión. Los fijos cobran un básico + $100 por cada año de antigüedad. Los de comisión cobran un básico + un porcentaje del precio de cada artículo que vendieron. Cada vendedor conoce los artículos que vendió. Los artículos pueden ser nacionales o importados. Los importados tienen un recargo del 21% por impuestos.

Implementar métodos para: agregar un vendedor si no se sobrepasó el máximo de vendedores en el plantel, devolver el total de artículos vendidos por todos los vendedores, obtener el total de sueldos a pagar.

///////////////////////////////////////////////////////////////////////////////////////////////////////

FINAL 4

1) ¿Que es un programa orientado a objetos? Definir todos los componentes

¿Cuáles son los principios básicos de la POO?

2) Definición de patrones. Dar 5 características...Escribir 3 ejemplos de patrones, y que funcionalidad tiene cada uno.

Tomar uno de los patrones y realizar un ejemplo en UML.

3) Definir self/super. Ejemplo.

Definir variable de instancia. Ejemplo.

Definir variable de clase. Ejemplo.

4) Realizar un diagrama UML con 8 clases como mínimo. Cada clase tiene q tener 4 características únicas, y 4 comportamientos únicos. O sea 4 métodos únicos.

El ejercicio no puede ser parecido a los de la teoría o práctica.