PARCIALITO 9

Desarrollar **individualmente** los temas indicados, respetando las consignas específicas si las hubiera.

Se valorarán la **claridad de las explicaciones**, la **brevedad y precisión** (sin omisión de información), la **aplicación en ejemplos** diferentes a los vistos en clase, y el **uso de recursos visuales** (cuadros, diagramas, mapas conceptuales, etc).

## 

# 

# TEMAS

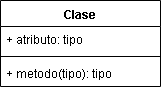
## CURSADA

### Modelado orientado a objetos

1. Explique las vistas del sistema acorde al modelado orientado a objetos. ¿Por qué se dice que estas vistas están regidas por una arquitectura basada en casos de uso?  
   *La respuesta no puede exceder las 6 (seis) líneas, considerando una tipografía Arial tamaño 10 sin formato adicional.*

Las vistas son proyecciones de los modelos que muestran determinados aspectos de un sistema, se pueden separar respecto al uso de datos, es decir, el uso de los usuarios sobre estas.

### Conceptos básicos del paradigma orientado a objetos

1. Defina los siguientes elementos e indique, si es posible, su sintaxis en UML:
   1. Clase Seria como el molde en el que se basan los objetos al instanciarse
   2. Objeto Es la instancia de la clase 
   3. Atributo Son los o el conjunto de valores que pueden tener los objetos de la clase
   4. Operación
   5. Método Es la funcionalidad que pueden tener los objetos de la clase

### Clasificadores

1. ¿Qué es un clasificador? Ejemplifique.

Un clasificador es un mecanismo que describe características estructurales y de comportamiento. Por ejemplo el clasificador abstract indica que la clase no se puede instanciar, solamente se puede heredar.

### Relaciones

1. Complete el siguiente cuadro:

|  |  |  |
| --- | --- | --- |
| Relación | Sintaxis UML | Características |
| Asociación |  | Relación semántica entre dos clases |
|  |  |  |
|  |  |  |

### Herencia

1. ¿Qué es la herencia? ¿Qué implica el principio de sustitución entre clases?

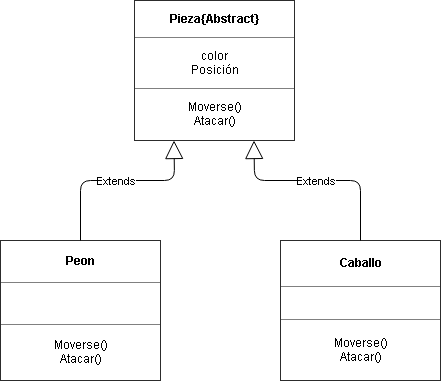
La herencia es la capacidad que tienen las clases de tomar tanto atributos como métodos de sus clases padres. El principio de sustitución entre clases implica que se pueden cambiar los métodos que se heredan de las clases padres a conveniencia de las clases hijas

### Polimorfismo

Crear un diagrama de clases en donde se pueda ver una relación de polimorfismo para las piezas del juego de ajedrez. Explicar en un párrafo, de no más de 5 líneas en formato arial 10, dicha relación.



***El ajedrez es un juego de mesa en el que se enfrentan dos jugadores, cada uno posee 16 piezas blancas o negras (según la elección del jugador) de formas diversas que pueden mover, según ciertas reglas, sobre un tablero dividido en 64 cuadros; gana el jugador que consigue dar mate al rey de su contrincante.***

Básicamente el polimorfismo está en que tanto el peón como el caballo se mueven y atacan pero de manera completamente distinta ya que el peón se mueve de un cuadro a la vez hacia adelante y ataca a los cuadros de adelante que sean diagonales a él, mientras que el caballo se mueve y ataca en ele y así mismo pasa con las demás piezas.

### Encapsulamiento

1. ¿Qué relación existe entre el encapsulamiento y los modificadores de acceso de los atributos definidos en una clase? Ejemplifique e indique sintaxis UML para cada caso.

La relación entre el encapsulamiento y los modificadores de acceso es que uno complementa el otro, ya que el encapsulamiento se utiliza para ocultar atributos y métodos y el cómo se hacen a la persona que lo está usando y sin los modificadores de acceso, cualquiera podría ver los elementos previamente mencionados. Por Ejemplo, para encapsular los atributos por lo general se les pone el modificador de acceso private, y se lo llama desde un método público tanto para obtener su información con un getter como para escribirla con un setter

### Diagrama de clases

1. Realizar un diagrama de clases para un sistema bancario (con clases abreviadas) en donde se puedan observar las distintas relaciones entre clasificadores.

## PROMOCIÓN

### Relación

1. Establezca relaciones entre los distintos pilares del paradigma orientado a objetos

Para que el paradigma orientado a objetos funcione, se debe poder realizar vistas sobre los modelos, para centrarse en lo importante, después de esto, se busca crear diagramas de clases con todas las funcionalidades, atributos, relaciones y demás para así ayudarle a entender al programador como debe construir el sistema y que tiene que hacer cada una de las clases, quien hereda de quien, en donde se tiene que aplicar el encapsulamiento, el polimorfismo y demás.