



UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERIA ELECTRÓNICA Y TELECOMUNICACIONES
PROGRAMA DE INGENIERIA DE SISTEMAS

ASIGNATURA:	PROGRAMACION ORIENTADA A OBJETOS
CODIGO:	SIS201
MODALIDAD:	PRESENCIAL TEORICO
INTENSIDAD:	4 HORAS TEORICAS.
PREREQUISITOS:	INTROD A LA INFORMATICA, LAB INTROD A LA INFORMATICA
CO-REQUISITOS:	LABORATORIO PROGRAMACIÓN ORIENTADA A OBJETOS
AREA:	INGENIERIA APLICADA
CREDITOS:	3

OBJETIVO GENERAL

Brindar los conceptos fundamentales de la programación orientada a objetos, buscando ofrecer un tratamiento riguroso de la teoría y la práctica haciendo uso de un lenguaje de programación altamente utilizado en el ámbito profesional.

OBJETIVOS ESPECIFICOS

El estudiante al final del curso estará en capacidad de:

- Entender y manejar cada uno de los conceptos fundamentales de la programación orientada a objetos.
- Relacionar los conceptos del paradigma de programación estructurado con el orientado a objetos.
- Desarrollar aplicaciones con un lenguaje orientado a objetos haciendo uso del UML como lenguaje visual de modelado.
- Definir los alcances de una herramienta de programación orientada a objetos.

METODOLOGIA

- El alumno adquirirá los conocimientos básicos a través de clases magistrales acompañadas de ejercicios prácticos.
- El alumno deberá profundizar sus conocimientos con temas complementarios desarrollando talleres prácticos basados en UML y un lenguaje de programación orientado a objetos y trabajos de investigación.

CONTENIDO

1. INTRODUCCIÓN (10 horas)

- 1.1. Complejidad y Calidad de Software
- 1.2. Manejo de complejidad. Técnicas de programación

- Programación No Estructurada
- Programación Procedimental
- Programación Modular
- Programación Orientada a Objetos
- Programación Genérica.
- Tipos Abstractos de dato

- 1.3. El Modelo O.O

- 1.3.1. Abstracción
- 1.3.2. Encapsulamiento
- 1.3.3. Modularidad
- 1.3.4. Jerarquía
- 1.3.5. Polimorfismo

1.4. PARADIGMA O.O.

- 1.4.1. Mensajes y métodos
- 1.4.2. Clases y Objetos
- 1.4.3. Herencia
- 1.4.4. Polimorfismo
- 1.4.5. Sobrecarga de funciones
- 1.4.6. Sobrecarga de operadores

1.5. INTRODUCCION AL MODELADO ORIENTADO A OBJETOS

- 1.5.1. ¿Qué es el modelado y para qué modelamos?
- 1.5.2. EL lenguaje unificado de modelado UML2.0
- 1.5.3. Vistas y diagramas
- 1.5.4. Diagramas de clase: clases y relaciones(asociación, agregación y generalización)
- 1.5.5. Diagramas de secuencia: interacción de objetos

2. CLASES Y METODOS(12 horas)

- 2.3. Clases
- 2.4. Constructores y destructores
- 2.5. Constructores Parametrizados
- 2.6. Miembros de alcance de clase
- 2.7. Funciones Amigas
- 2.8. Funciones insertadas
- 2.9. Diagramas de clases y relaciones

3. OBJETOS, FUNCIONES Y PUNTEROS (10 horas)

- 3.3. Punteros a objetos.
- 3.4. Arreglos de objetos.
- 3.5. Paso de objetos a funciones.
- 3.6. Paso por Referencia.
- 3.7. Diagramas de colaboración: representación de la ejecución del programa orientado a objetos.
- 3.8. Sobrecarga de funciones.
- 3.9. Sobrecarga de operadores.

4. LA ROBUSTEZ DE SOFTWARE Y EL MANEJO DEL ERROR (3 horas)

- 4.3. Conceptos.
- 4.4. Captura y manejo de excepciones.
- 4.5. Lanzamiento de excepciones.
- 4.6. Manejadores de excepciones.

5. HERENCIA (10 horas)

- 5.3. Definición y beneficios
- 5.4. Tipos de herencia: Simple y Múltiple
- 5.5. Clases bases virtuales
- 5.6. Visibilidad de la herencia
- 5.7. Excepciones y herencia

6. POLIMORFISMO (10 horas)

- 6.3. Ligadura temprana y ligadura tardía
- 6.4. Clases Abstractas y métodos abstractos
- 6.5. Funciones virtuales
- 6.6. Constructores y destructores por herencia
- 6.7. Tipo e implementación: interfaces e implementaciones

7. PLANTILLAS (5 horas)

- 7.3. Genericidad.
- 7.4. Conceptos fundamentales.
- 7.5. Plantillas de funciones.
- 7.6. Plantillas de clases.
- 7.7. Plantillas frente a Polimorfismo.
- 7.8. Standard Template Library

EVALUACIONES

Se realizarán tres (3) evaluaciones de la siguiente forma:

NUMERO	%	COMPONENTES
Primer Parcial	35%	Parcial Escrito 70%
		Quices, Talleres, tareas, etc. 30%
Segundo Parcial	35%	Parcial Escrito 70%
		Quices, Talleres, tareas, etc. 30%
Tercer Parcial	30%	Parcial Escrito 70%
		Quices, Talleres, tareas, etc. 30%

Los talleres y tareas en grupo serán evaluados individualmente y deben estar debidamente documentados.

BIBLIOGRAFÍA

- ✓ Object-Oriented Análisis and Design with Applications. Booch, Grady.
- ✓ Joyanes A. Luis. Programación Orientada a Objetos. Segunda Edición. McGraw-Hill. 1998.
- ✓ Budd Timothy. Introducción a la Programación Orientada a Objetos. Addison Wesley Iberoamericana. 1994.
- ✓ Stroustrup Bjarne. The C++ Programming Language. Third Edition. 1999.
- ✓ Hebert Schildt. Turbo C/C++. McGraw -Hill.