



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE ESTUDIOS SUPERIORES ACATLÁN

PLAN DE ESTUDIOS DE LA LICENCIATURA EN
MATEMÁTICAS APLICADAS Y COMPUTACIÓN

PROGRAMA DE ASIGNATURA



SEMESTRE:3 (TERCERO)

Programación Orientada
a Objetos

CLAVE:

MODALIDAD	CARÁCTER	TIPO	HORAS AL SEMESTRE	HORAS SEMANA	HORAS TEÓRICAS	HORAS PRÁCTICAS	CRÉDITOS
Curso	Obligatoria	Teórico-Práctica	96	6	4	2	10

ETAPA DE FORMACIÓN	Básico
CAMPO DE CONOCIMIENTO	Computación

SERIACIÓN	Indicativa
ASIGNATURA(S) ANTECEDENTE	Programación II
ASIGNATURA(S) SUBSECUENTE(S)	Métodos Numéricos II
Objetivo general: El alumno desarrollará aplicaciones basadas en el análisis y diseño propuesto por el enfoque del paradigma orientado a objetos, independiente del lenguaje de programación a utilizar para su implementación.	

Índice Temático		Horas	
Unidad	Tema	Teóricas	Prácticas
1	Introducción a la programación orientada a objetos	4	2
2	Análisis y diseño con UML	12	6
3	Implementación de clases y objetos	12	6
4	Sobrecarga de funciones, copia y referencia	12	6
5	Herencia	12	6
6	Tópicos abstractos	12	6
Total de horas:		64	32
Suma total de horas:		96	

HORAS		UNIDAD	CONTENIDO
T	P		
4	2	1	INTRODUCCIÓN A LA PROGRAMACIÓN ORIENTADA A OBJETOS Objetivo particular: El alumno identificará los conceptos fundamentales de la Programación Orientado a Objetos. Temas: 1.1 Antecedentes de la programación orientada a objetos 1.2 Conceptos básicos: objeto, clase, abstracción, encapsulamiento, sobrecarga de funciones, herencia y polimorfismo. 1.3 Enfoque orientado a objetos 1.4 Ventajas de la programación y métodos orientados a objetos 1.5 Lenguajes de programación orientados a objetos
12	6	2	ANÁLISIS Y DISEÑO CON UML Objetivo particular: El alumno aplicará el Lenguaje Unificado de Modelado (UML) para el desarrollo de aplicaciones. Temas: 2.1 Análisis orientado a objetos con UML 2.1.1 Estándares recomendados de diagramación 2.2 Diseño orientado a objetos con UML 2.2.1 Diseño de la estructura y comportamiento de los objetos 2.2.2 Estándares recomendados de diagramación
12	6	3	IMPLEMENTACIÓN DE CLASES Y OBJETOS Objetivo particular: El alumno implementará clases que representen objetos del mundo real indicando los componentes y funciones que los conforman, a través de un lenguaje de programación según el paradigma Orientado a Objetos. Temas: 3.1 Implementación de clases 3.2 Creación de objetos 3.3 Protección 3.4 Datos y funciones miembro de una clase 3.5 Constructores e instanciación 3.6 Destructores
12	6	4	SOBRECARGA DE FUNCIONES, COPIA Y REFERENCIA Objetivo particular: El alumno desarrollará aplicaciones con sobrecarga de funciones, paso de objetos como parámetros y devolución de objetos desde funciones. Temas: 4.1 Redefinición de operadores 4.2 Sobrecarga de funciones 4.3 Función amiga 4.4 Asignación y copia de objetos 4.5 Paso de objetos como parámetros en funciones 4.6 Funciones que retornan objetos

12	6	5	HERENCIA Objetivo particular: El alumno ampliará clases ya existentes a nuevos requerimientos con base en el concepto de herencia. Temas: 5.1 Ampliación mediante herencia 5.2 Herencia y protección 5.3 Constructores y herencia 5.4 Funciones virtuales 5.5 Polimorfismo 5.6 Herencia múltiple 5.7 Herencia y composición
12	6	6	TÓPICOS ABSTRACTOS Objetivo Particular: El alumno aplicará conceptos abstractos del paradigma orientado a objetos. Temas: 6.1 Clases abstractas 6.2 Genericidad: clases contenedoras, funciones genéricas, arreglos y diccionarios de datos 6.3 Constantes y objetos compartidos

Referencias básicas:

- Barnes, David J. y Kölling, Michael. (2007). *Programación Orientada a Objetos con Java* (3ed). España: Prentice-Hall.
- Budd, T. (2001). *An introduction to object oriented programming* (3ed). E.U.A.: Addison Wesley.
- Ceballos, Fco. Javier (2007). *Programación Orientada a Objetos Con C++* (4 ed). España: RA-MA.
- James Martin, James J. Odell. (1994). *Análisis y Diseño Orientado a Objetos*. México: Prentice
- Joyanes Aguilar, Luis, Zahonero Martínez, Ignacio. (2010). *Programación en C, C++, Java, UML*. México: McGraw Hill.
- Joyanes Aguilar, Luis, Zahonero Martínez, Ignacio (2011). *Programación en Java. Algoritmos, programación Orientada a Objetos e interfaz gráfica de usuario*. México: McGraw Hill.
- Joyanes Aguilar, Luis. (2006). *Programación en C++. Algoritmos, Estructuras de Datos y Objetos* (2ed). México: McGraw Hill.
- Smith, Jo Ann. (2001). *Programación Orientada a Objetos con C++*. México: Thomsom Learning.
- Schach, Sthepen R. (2005). *Análisis y Diseño Orientado a Objetos con UML y el Proceso Unificado*. México: McGraw Hill Interamericana.

Referencias complementarias:

- Guardati, Silvia. (2007). *Estructura de Datos Orientada a Objetos*. México: Pearson.
- Winblad, A. (1991). *Object oriented software*. E.U.A.: Addison Wesley.
- Fisher, A. (1991). *C.A.S.E. Using software development tools*. E.U.A.: Willey.

Sugerencias didácticas:	Sugerencias de evaluación del aprendizaje:
<p>Analizar y producir textos</p> <p>Utilizar tecnologías multimedia</p> <p>Resolver ejercicios dentro y fuera de clase</p> <p>Estudiar casos</p> <p>Instrumentar técnicas didácticas como exposición audiovisual, exposición oral, interrogatorio, técnicas grupales de trabajo colaborativo, entre otros</p> <p>Trabajo de investigación</p> <p>Realizar visitas de observación</p> <p>Prácticas de campo</p> <p>Usar recursos didácticos en línea.</p> <p>Utilizar algún lenguaje de programación orientado a objetos para realizar sistemas computacionales en el que se involucren los conocimientos adquiridos en la materia.</p> <p>Explicar el entorno del lenguaje a utilizar.</p> <p>Hacer uso del laboratorio de cómputo en la programación de aplicaciones basadas en el paradigma orientado a objetos</p>	<p>Examen final oral o escrito</p> <p>Exámenes parciales</p> <p>Informes de prácticas</p> <p>Informes de investigación</p> <p>Participación en clase</p> <p>Rúbricas</p> <p>Solución de ejercicios</p> <p>Trabajos y tareas</p>

Perfil Profesiográfico: El profesor que imparta la asignatura deberá tener el título de licenciado en Matemáticas Aplicadas y Computación o carrera afin, con experiencia profesional y docente en la materia, contar con actualización en el área y preferentemente tener estudios de posgrado.