

PLAN DE ESTUDIOS (PE): Ingeniería en Tecnologías de la Información

AREA: Modelado de Sistemas

ASIGNATURA: Programación Orientada a Objetos I

CÓDIGO: ITIM-011

CRÉDITOS: 5

FECHA: 10 de diciembre de 2012



1. DATOS GENERALES

Nivel Educativo:	Licenciatura
Nombre del Plan de Estudios:	Licenciatura en Ingeniería en Tecnologías de la Información
Modalidad Académica:	Presencial
Nombre de la Asignatura:	Programación Orientada a Objetos I
Ubicación:	Básico
Correlación:	
Asignaturas Precedentes:	Introducción a la programación
Asignaturas Consecuentes:	Programación Orientada a Objetos II
Conocimientos, habilidades, actitudes y valores previos:	<p>Conocimientos</p> <ul style="list-style-type: none"> • De expresión escrita, razonamiento lógico. • De cultura nacional y regional. • De estructuras de control de programación. • De Modelado Orientado a Objetos. • De Programación Orientada a Objetos. • De herramientas de modelado. <p>Habilidades</p> <ul style="list-style-type: none"> • Creatividad e iniciativa en la búsqueda de soluciones a problemas concretos. • De análisis e inteligencia práctica. • De observación, iniciativa y disposición al trabajo en equipo. • Para la abstracción y modelado de sistemas. • Para la lectura y comprensión de textos. • Para generar diagramas de Clase, Casos de Uso y Secuencia. <p>Actitudes y valores</p> <ul style="list-style-type: none"> • Perseverancia y compromiso social. • Para adquirir nuevos conocimientos y realizar innovaciones. • Inclinación hacia la tecnología. • Al autoconocimiento permanente. • De respeto y empatía con las personas. • De honestidad y responsabilidad. • De liderazgo y humanismo. • De participación activa. • Apertura al diálogo, comprensión y tolerancia hacia la diversidad cultural.

2. CARGA HORARIA DEL ESTUDIANTE (Ver matriz 1)

Concepto	Horas por periodo		Total de horas por periodo	Número de créditos
	Teoría	Práctica		
Horas teoría y práctica (16 horas = 1 crédito)	48	32	80	5
Total	48	32	80	5

3. REVISIONES Y ACTUALIZACIONES

Autores:	Luz A. Sánchez Gálvez Mario Anzures García Miguel Rodríguez Hernández María Elena Flores de Ita Leticia Mendoza Alonso Meliza Contreras González Rafael De la Rosa Flores Ana Patricia Cervantes Márquez Luis Enrique Colmenares Guillen Juan Manuel González Calleros	José Andrés Vázquez Flores Beatriz Beltrán Martínez Hilda Castillo Zacatelco Mario Rossainz López Manuel Martín Ortíz Pedro Bello López Laura Cuayahuitl Romero Marco Antonio Soriano Ulloa Darnes Vilariño Ayala
Fecha de diseño:	10 de Diciembre de 2012	
Fecha de la última actualización:	10 de Diciembre de 2012	
Fecha de aprobación por parte de la academia de área	6 de Febrero de 2013	
Fecha de aprobación por parte de CDESC-UA		
Fecha de revisión del Secretario Académico		
Revisores:	Luz A. Sánchez Gálvez Mario Anzures García Miguel Rodríguez Hernández Leticia Mendoza Alonso Meliza Contreras González Rafael De la Rosa Flores Ana Patricia Cervantes Márquez Juan Manuel González Calleros Abraham Sánchez López	José Andrés Vázquez Flores Beatriz Beltrán Martínez Hilda Castillo Zacatelco Mario Rossainz López Pedro Bello López Laura Cuayahuitl Romero Marco Antonio Soriano Ulloa Darnes Vilariño Ayala Manuel Martín Ortíz
Sinopsis de la revisión y/o actualización:	Es una asignatura de nueva creación	



4. PERFIL DESEABLE DEL PROFESOR (A) PARA IMPARTIR LA ASIGNATURA:

Disciplina profesional:	Ciencias de la computación Ingeniería en Computación Ciencias Físico Matemáticas Tecnologías de la Información Sistemas Computacionales
Nivel académico:	Maestría
Experiencia docente:	Mínima 2 años
Experiencia profesional:	Mínima 2 años

5. OBJETIVOS:

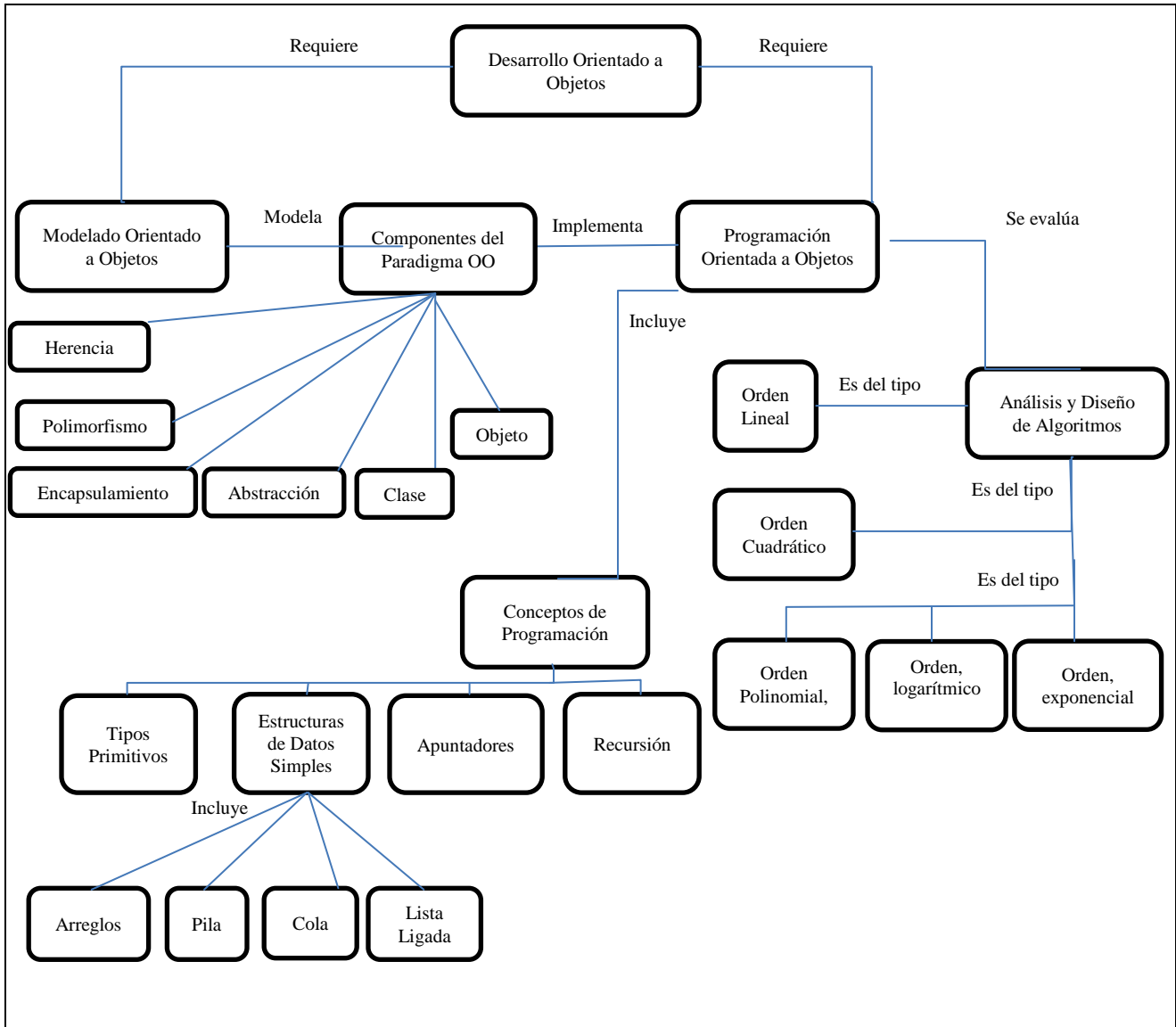
5.1 General: Construir programas en un Lenguaje Orientado a Objetos para resolver problemas a partir del modelo desarrollado, utilizando estructuras de datos básicas.

5.2 Específicos: El alumno será capaz de:

- Identificar los conceptos básicos de la Programación Orientada a Objetos para aplicarlos a la solución de problemas.
- Analizar y aplicar las operaciones sobre arreglos, cadenas, apuntadores y flujos para la solución de problemas.
- Aplicar la técnica de recursión para la solución de problemas.
- Identificar y aplicar estructuras de datos para la organización de la información en la solución de problemas.
- Conocer las reglas y métodos básicos de análisis y diseño de algoritmos en la solución de problemas.



6. REPRESENTACIÓN GRÁFICA DE LA ASIGNATURA:



7. CONTENIDO

Unidad 1	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
Introducción a la Programación Orientada a Objetos	Identificar los conceptos básicos de la Programación Orientada a Objetos	1.1 Estructura general de un programa	Sprankle.,M., Hubbard, J. (2012) Problem Solving & Programming Concepts, Ninth Edition, Prentice Hall Jacobson, I., Rumbaugh, J. & Booch, G. 2005. El Lenguaje Unificado de Modelado. Pearson, Addison Wesley.	López, L. (2011) Programación Estructurada y orientada a objetos. Un enfoque algorítmico. (3er. Ed.), Alfaomega. Goodrich, M. T., Tamassia, R. and Mount, D. (2011). Data Structures & Algorithm in C++. 2ª. Ed. John Wiley and Sons, Inc..
		1.2 Tipos primitivos		
		1.3 Clase		
		1.4 Objeto		
		1.5 Método		
		1.6 Herencia		
		1.7 Polimorfismo		

Unidad 2	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
Tipos de datos simples y flujos	Analizar y aplicar las operaciones sobre arreglos, cadenas, apuntadores y flujos para la solución de problemas.	2.1 Arreglos unidimensionales	Sprankle.,M., Hubbard, J. (2012) Problem Solving & Programming Concepts, Ninth Edition, Prentice Hall Goodrich, M. T., Tamassia, R. and Mount, D. (2011). Data Structures & Algorithm in C++. 2ª. Ed. John Wiley and Sons, Inc..	López, L. (2011) Programación Estructurada y orientada a objetos. Un enfoque algorítmico. (3ª. Ed.), Alfaomega.
		2.2 Manejo de cadenas		
		2.3 Arreglos bidimensionales		
		2.4 Apuntadores		
		2.4.1 Gestión dinámica de memoria		
		2.5 Flujos		



Unidad 3	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
Recursión	Aplicar la técnica de recursión para la solución de problemas.	3.1 Definición de recursión	Goodrich, M. T., Tamassia, R. and Mount, D. (2011). Data Structures & Algorithm in C++. 2ª. Ed. John Wiley and Sons, Inc.. López, L. (2011) Programación Estructurada y orientada a objetos. Un enfoque algorítmico. (3er. Ed.), Alfaomega.	López, L. (2011) Programación Estructurada y orientada a objetos. Un enfoque algorítmico. (3er. Ed.), Alfaomega. Jacobson, I., Rumbaugh, J. & Booch, G. 2005. El Lenguaje Unificado de Modelado. Pearson, Addison Wesley. Weiss, M.A. (2006), Data Structures and Algorithm Analysis in C++, 3rd Edition, Prentice Hall.
		3.2 Tipos de recursión		
		3.2.1 Simple		
		3.2.2 Múltiple		
		3.2.3 Anidada		
		3.2.4 Indirecta		
3.3 Comparación entre métodos iterativos y recursivos.				

Unidad 4	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
Tipo de Datos Abstractos	Identificar y aplicar estructuras de datos para la organización de la información en la solución de problemas.	4.1 TDA Pila	Goodrich, M. T., Tamassia, R. and Mount, D. (2011). Data Structures & Algorithm in C++. 2ª. Ed. John Wiley and Sons, Inc.. Larry R. Nyhoff (2005). TADs Estructuras de Datos y resolución de Problemas con C++ (2ª ed). México: Pearson.	Jacobson, I., Rumbaugh, J. & Booch, G. 2005. El Lenguaje Unificado de Modelado. Pearson, Addison Wesley. Wesley. Weiss, M.A. (2006), Data Structures and Algorithm Analysis in C++, 3rd Edition, Prentice Hall.
		4.1.1 Operaciones push y pop		
		4.1.2 Aplicaciones		
		4.2 TDA Cola		
		4.2.1 Operaciones inserción y eliminación		
		4.2.2 TDA Cola circular		
		4.2.3 Aplicaciones		
		4.3 TDA Listas ligadas (Descripción, Operaciones, Aplicaciones)		
		4.3.1 Simples		
		4.3.2 Dobles		
		4.3.3 Circulares		

Unidad 5	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
Análisis y Diseño de Algoritmos	Conocer las reglas y métodos básicos de análisis y diseño de algoritmos en la solución de problemas.	5. Análisis de Algoritmos: Complejidad	Data Structures and Algorithm Analysis in C++, 3rd Edition, Prentice Hall.	Jacobson, I., Rumbaugh, J. & Booch, G. 2005. El Lenguaje Unificado de Modelado. Pearson, Addison Wesley. Wesley.Weiss, M.A. (2006),
		5.1 Orden Lineal		
		5.2 Orden Cuadrático		
		5.3 Orden Polinomial		
		5.4 Orden logarítmico		
5.5 Orden exponencial				

8. CONTRIBUCIÓN DEL PROGRAMA DE ASIGNATURA AL PERFIL DE EGRESO

Asignatura	Perfil de egreso (anotar en las siguientes tres columnas, cómo contribuye la asignatura al perfil de egreso)		
	Conocimientos	Habilidades	Actitudes y valores
Programación Orientada a Objetos I	De la gestión, planeación y mejora de modelado y programación orientada a objetos para proyectos en TIs. De la metodología del paradigma orientado a objetos para el desarrollo de aplicaciones.	Para modelar en el paradigma orientado a objetos sistemas integrales para organizaciones. Para aplicar las técnicas y herramientas de la programación orientada a objetos para la implementación de los modelos y diseños de cómputo.	Considerará los principios éticos, el compromiso social, el cuidado del medio ambiente y el respeto a las normas.

9. Describa cómo el eje o los ejes transversales contribuyen al desarrollo de la asignatura

Eje (s) transversales	Contribución con la asignatura
Formación Humana y Social	Promover el trabajo en equipo para resolver problemas de la vida real utilizando modelado y programación orientada a objetos.
Desarrollo de Habilidades en el uso de las Tecnologías de la Información y la Comunicación	Promover el uso de herramientas tecnológicas para modelar soluciones a problemas reales a través del paradigma orientado a objetos.
Desarrollo de Habilidades del Pensamiento Complejo	Contribuir al proceso de representación de un problema con diagramas de modelado.
Lengua Extranjera	Lectura de bibliografía en el área de modelado y programación de sistemas en el POO.
Innovación y Talento Universitario	Representación de la solución de problemas del mundo real a través del modelado OO.

10. ORIENTACIÓN DIDÁCTICO-PEDAGÓGICA

Estrategias y Técnicas de aprendizaje-enseñanza	Recursos didácticos
<p>Estrategias de aprendizaje:</p> <ul style="list-style-type: none"> • Lectura y comprensión, • Reflexión, • Comparación, • Resumen. <p>Estrategias de enseñanza:</p> <ul style="list-style-type: none"> • ABP, • Aprendizaje activo, • Aprendizaje cooperativo, • Aprendizaje colaborativo, • Basado en el descubrimiento. <p>Ambientes de aprendizaje:</p> <ul style="list-style-type: none"> • Aula, • Laboratorio, • Simuladores. <p>Actividades y experiencias de aprendizaje:</p> <ul style="list-style-type: none"> • Visita a empresas. 	<p>Técnicas</p> <ul style="list-style-type: none"> • grupales, • de debate, • del diálogo, • de problemas, • de estudio de casos, • cuadros sinópticos, • mapas conceptuales, • para el análisis, • comparación, • síntesis, • mapas mentales, • lluvia de ideas, • analogías, • portafolio, • exposición. <p>Herramientas</p> <ul style="list-style-type: none"> • Compiladores • Herramientas y Bibliotecas para el desarrollo de Aplicaciones de Software • Herramientas de Modelado de Aplicaciones de Software Orientadas a Objetos e Interfaces Gráficas de Usuario • Herramientas para el trabajo y desarrollo de software en Grupos

11. CRITERIOS DE EVALUACIÓN

Criterios	Porcentaje
▪ Exámenes	35%
▪ Tareas	10%
▪ Prácticas de laboratorio	20%
▪ Proyecto final	35%
Total	100%

12. REQUISITOS DE ACREDITACIÓN

Estar inscrito como alumno en la Unidad Académica en la BUAP
Asistir como mínimo al 80% de las sesiones
La calificación mínima para considerar un curso acreditado será de 6
Cumplir con las actividades académicas y cargas de estudio asignadas que señale el PE

13. Anexar (copia del acta de la Academia y de la CDESC- UA con el Vo. Bo. del Secretario Académico)

