

**PLAN DE ESTUDIOS (PE): Licenciatura en Ciencias de la Computación /
Ingeniería en Ciencias de la Computación**

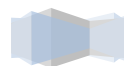
AREA: Ciencias de la Computación / Ingeniería en Computación

ASIGNATURA: Programación II

CÓDIGO: CCOM-015

CRÉDITOS: 5

FECHA: 10 de junio de 2012



1. DATOS GENERALES

Nivel Educativo:	Licenciatura
Nombre del Plan de Estudios:	Licenciatura en Ciencias de la Computación / Ingeniería en Ciencias de la Computación
Modalidad Académica:	Presencial
Nombre de la Asignatura:	Programación II
Ubicación:	Nivel Básico
Correlación:	
Asignaturas Precedentes:	Programación I
Asignaturas Consecuentes:	Estructuras de Datos, Graficación
Conocimientos, habilidades, actitudes y valores previos:	<p>Conocimientos de:</p> <ul style="list-style-type: none"> • La estructura de control y su aplicación en problemas. • Las técnicas y herramientas de solución de problemas a través de algoritmos. • Las diversas formas de resolver un problema para elegir la más adecuada. • Las formas de representar un algoritmo. <p>Habilidades</p> <ul style="list-style-type: none"> • Identificar los elementos que conforman un algoritmo. • Reconocer las herramientas de representación de algoritmos. • Capacidad de aplicar los conocimientos adquiridos de algoritmia en un lenguaje estructurado de programación. • Creatividad en la búsqueda de soluciones a problemas concretos. • Capacidad creativa y analítica, inteligencia práctica, perseverancia y compromiso social. • Capacidad de observación, iniciativa y disposición al trabajo en equipo. <p>Actitudes y valores</p> <ul style="list-style-type: none"> • De trabajo colaborativo.



	<ul style="list-style-type: none"> • Una actitud favorable para adquirir nuevos conocimientos y realizar innovaciones. • De respeto y empatía con las personas. • De Honestidad y responsabilidad. • De liderazgo y humanismo. • Actitud participativa.
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

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 <i>Actividades bajo la conducción del docente como clases teóricas, prácticas de laboratorio, talleres, cursos por internet, seminarios, etc.</i> (16 horas = 1 crédito)	48	32	80	5
Total	48	32	80	5

3. REVISIONES Y ACTUALIZACIONES

Autores:	Beatriz Beltrán Martínez Graciano Cruz Almanza Hilda Castillo Zacatelco Carmen Cerón Garnica José Andrés Vázquez Flores Erica Vera Cervantes Judith Pérez Marcial Laura Cuayahuitl Romero	Darnes Vilariño Ayala Marco Antonio Soriano Ulloa Rafael De la Rosa Flores Meliza Contreras González Miguel Rodríguez Hernández Pedro Bello López José Luis Meza León Yolanda Moyao Martínez
Fecha de diseño:	10 de Junio de 2009	
Fecha de la última actualización:	10 de Junio de 2012	
Fecha de aprobación por parte de la academia de área	29 de enero de 2013	
Fecha de aprobación por parte de CDESCUA	15-febrero-2013	
Fecha de revisión del Secretario Académico	18-febrero-2013	
Revisores:	Ana P. Cervantes Márquez Beatriz Beltrán Martínez Carmen Cerón Garnica	María Luz A. Sánchez Galvez Mario Rossainz López Mario Anzures García



	Darnes Vilariño Ayala Graciano Cruz Almanza Hilda Castillo Zacatelco José Andrés Vázquez Flores Laura Cuayahuitl Romero Leticia Mendoza Alonso Marco Antonio Soriano Ulloa	Meliza Contreras González Miguel Rodríguez Hernández Mireya Tovar Vidal Pedro Bello López Rafael De la Rosa Flores Yalú Galicia Hernández Yolanda Moyao Martínez
Sinopsis de la revisión y/o actualización:	Se considero la necesidad de agregar la especificación sintáctica de las estructuras de datos abstractas, pila, cola y listas ligadas así como distintas herramientas para el modelado.	

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

Disciplina profesional:	Ciencias de la Computación
Nivel académico:	Maestría
Experiencia docente:	Mínima 2 años
Experiencia profesional:	Mínima 1 año

5. OBJETIVOS:

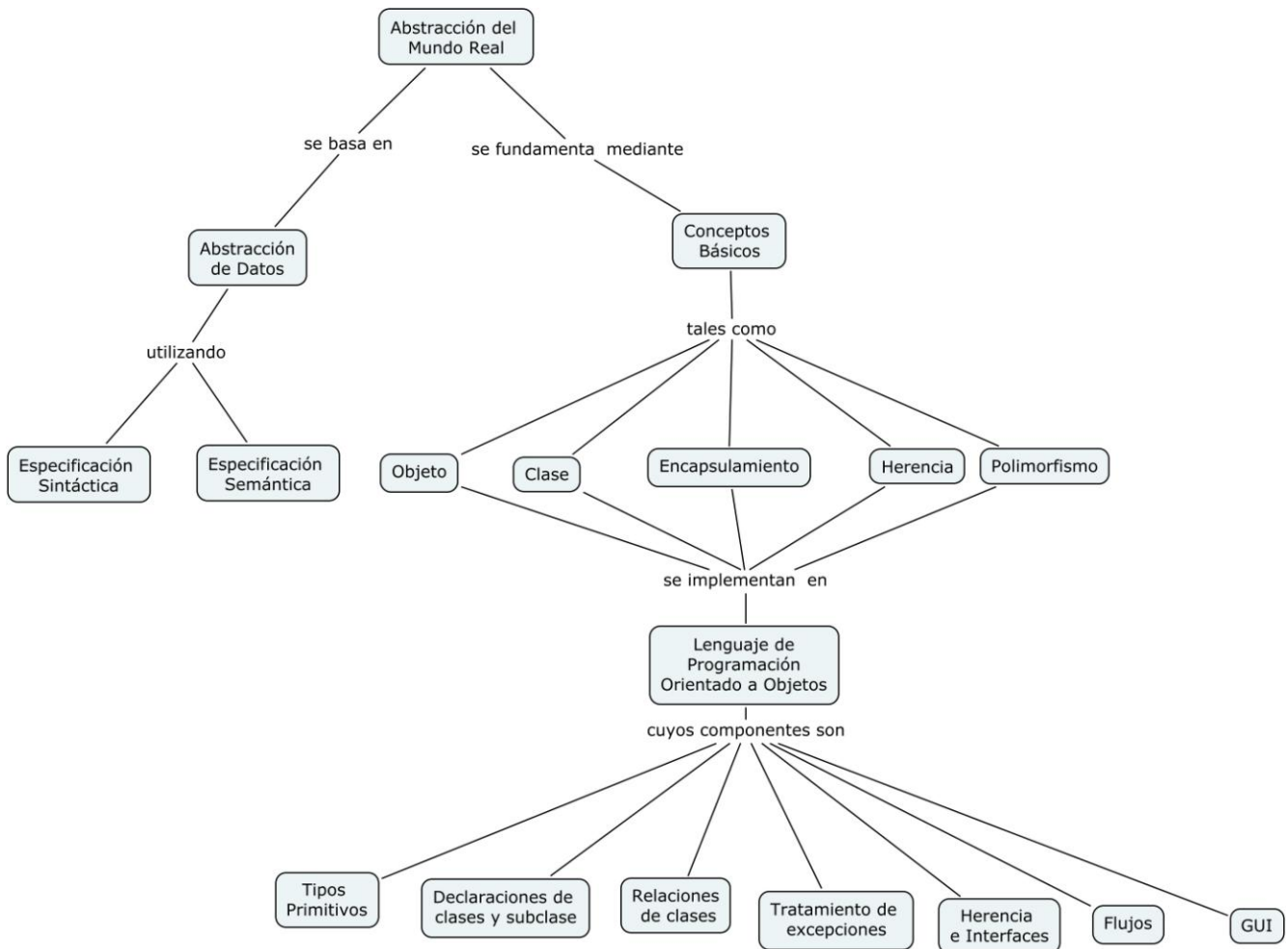
5.1. General: El estudiante obtendrá los conocimientos y habilidades para resolver problemas, utilizando el paradigma orientada a objetos, aplicándolo en un lenguaje de programación orientado a objetos, a través del trabajo en equipo de manera colaborativa, responsable, honesta y comprometido con su aprendizaje.

5.2. Específicos: El estudiante será capaz de:

- Identificar los conceptos básicos del paradigma orientado a objetos.
- Construir abstracciones del mundo real al paradigma orientado a objetos.
- Identificar los elementos que conforman un lenguaje de programación orientado a objetos.
- Analizar las diversas formas de resolver un problema para elegir la más adecuada.

6. REPRESENTACIÓN GRÁFICA DE LA ASIGNATURA:





7. CONTENIDO

Unidad 1	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
Conceptos básicos del paradigma Orientado a Objetos	Identificar los conceptos básicos del paradigma orientado a objetos.	1.1 Conceptos relacionados con la abstracción de datos.	James Martin, James J. (2007), <i>Análisis y Diseño Orientado a Objetos</i> (3ª. Ed.), Prentice Hall	Miguel Katrib Mora (1997), <i>Programación Orientada a objetos en C++, X VIEW</i>
		1.2 Clase y Objeto		
		1.3 Encapsulamiento		
		1.4 Herencia y Polimorfismo		
		1.4.1 Polimorfismo genérico (programación)		



Unidad 1	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
		genérica)	Estructura de datos y organización de archivos, Prentice Hall	

Unidad 2	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
Abstracción del mundo real al paradigma Orientado a Objetos	Construir abstracciones del mundo real al paradigma orientado a objetos.	2.1 Principios básicos de modelado de objetos.	Rumbaugh, James, Blaha, Michael, Premerlani, William,(2001), Modelado y Diseño Orientado a Objetos, España: Pearson Educación	Grady Booch, Díaz de Santos (1994). Análisis y Diseño Orientado a Objetos con Aplicaciones. (2da. Ed.), México: Addison-Wesley.
		2.2 CRC		
		2.3 Modelar objetos y atributos.		
		2.4 Modelar clases y métodos.		
		2.5 Modelar relaciones entre clases:		
		2.5.1. Dependencia o instanciación.		
		2.5.2. Asociación		
2.5.3. Agregación				
		2.5.3. Generalización		

Unidad 3	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
Lenguaje de Programación Orientado a Objetos	Identificar los elementos que conforman un lenguaje de programación orientado a objetos	3.1 Tipos de datos primitivos	Joyanes, L., Zahonero, I. (2002), Programación en Java 2: Algoritmos, Estructuras de Datos y	Deitel, H. M. y Deitel, P. J. (2004), JAVA Como programar (5ª. Ed.) Pearson Educación.
		3.2 Declaración de clases y subclases		
		3.3 Declaración de herencia e interfaces		
		3.4 Relaciones entre clases		
			Dcker Hirsfield	



Unidad 3	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
		3.5 Clases y métodos genéricos	<p>Programación Orientada a Objetos. (1ª. Ed), McGraw Hill.</p> <p>Arnold Gusting Holmes.(2001), El Lenguaje de Programación JAVA. (3ª. Ed.),Addison Wesley.</p> <p>Paul S. Wang. (2000), Java con programación orientada a objetos y aplicaciones en la WWW, Thompson Learning</p>	<p>(2000), Programación con JAVA. Thomson Learning</p> <p>Arnow Weiss,(2000), Introducción a la programación JAVA. Addison Wesley.</p>

Unidad 4	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
Lenguaje de Programación Orientado a Objetos Elementos avanzados.	Analizar las diversas formas de resolver un problema para elegir la más adecuada	4.1 Tratamiento de excepciones	Joyanes, L., Zahonero, I. (2002), Programación en Java 2: Algoritmos, Estructuras de Datos y Programación Orientada a Objetos. (1ª.	Deitel, H. M. y Deitel, P. J. (2004), JAVA Como programar (5ª. Ed.) Pearson Educación.
		4.2 Desarrollo de ejercicios con tipos de datos primitivos y abstractos(Pila, Cola y Lista ligada)		
		4.3. Interfaces Gráficas de Usuario.		
		4.4 Flujos de Entrada/Salida.		
				Dcker Hirsfield (2000), Programación con JAVA. Thomson



Unidad 4	Objetivo Específico	Contenido Temático/Actividades de aprendizaje	Bibliografía	
			Básica	Complementaria
		4.5 Recursividad	Ed), McGraw Hill. (2002).	Learning Arnow Weiss, (2000), Introducción a la programación JAVA. Addison Wesley.

8. CONTRIBUCIÓN DEL PROGRAMA DE ASIGNATURA AL PERFIL DE EGRESO

Asignatura	Perfil de egreso (anotar en las siguientes tres columnas a qué elemento(s) del perfil de egreso contribuye esta asignatura)		
	Conocimientos	Habilidades	Actitudes y valores
Programación II	De los conceptos principales de las ciencias de la computación, así como de los diferentes paradigmas de modelado y programación, para analizar, elaborar, desarrollar proyectos, de software.	Para identificar problemas y proponer soluciones basadas en el uso de algoritmos y computadoras integrando componentes de hardware y software, así como el de aplicar metodologías para el análisis, diseño e implementación de sistemas basados en hardware y software.	Mantendrá una actitud favorable para la actualización permanente en la disciplina. Será un profesional responsable, solidario, crítico, ético y comprometido con la sociedad y con el medio ambiente. Estará preparado para trabajar en equipo, emprender, liderar proyectos e incidir en la transformación de la realidad sustentablemente.

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	Tomar consciencia de que los conocimientos y habilidades adquiridos pueden ser usados para modelar problemas de la vida real.



Desarrollo de Habilidades en el uso de las Tecnologías de la Información y la Comunicación	Interactuar con diferentes plataformas y herramientas que permitan integrar componentes en la solución de problemas.
Desarrollo de Habilidades del Pensamiento Complejo	Modelado de problemas de diversas disciplinas de la ciencia a través de componentes y sus relaciones.
Lengua Extranjera	Lectura de artículos especializados en el área de la programación orientada a objetos.
Innovación y Talento Universitario	Aplicación de los conceptos aprendidos en la solución de problemas de la vida real.
Educación para la Investigación	Capacidad de migrar el modelado de la solución de un problema a diversos lenguajes de programación orientados a objetos.

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: Visita a empresas.</p> <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, 	<p>Plumón y pizarrón, Ejercicios Prácticas de Laboratorio Libros, fotocopias Artículos científicos Antologías Materiales audiovisuales Programas informáticos (CD u on-line) educativos. Aplicaciones Multimedia Páginas Web Webquest correo electrónico chats foros</p>



Estrategias y Técnicas de aprendizaje-enseñanza	Recursos didácticos
<ul style="list-style-type: none"> • portafolio, • exposición. 	

11. CRITERIOS DE EVALUACIÓN

Criterios	Porcentaje
• Exámenes	35%
• Participación en clase	15%
• Tareas	15%
• Prácticas de laboratorio	15%
• Proyecto final	20%
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 CDESCUA con el Vo. Bo. del Secretario Académico)

