



Universidad Autónoma de Sinaloa

Escuela de Ciencias de la Tierra

LICENCIATURA EN INGENIERÍA GEOMÁTICA



PROGRAMA DE ESTUDIOS

1. DATOS DE IDENTIFICACIÓN			
UNIDAD DE APRENDIZAJE	PROGRAMACIÓN E INGENIERÍA DE SOFTWARE II		
Clave:	(pendiente)		
Semestre:	VI semestre		
Eje Curricular:	<input type="checkbox"/> Tronco Común <input checked="" type="checkbox"/> Profesionalizante		
Área:	<input type="checkbox"/> Física-Matemática <input type="checkbox"/> Cs. Sociales y Humanidades <input type="checkbox"/> Idiomas <input type="checkbox"/> Básico Profesional <input checked="" type="checkbox"/> Profesional		
Horas y créditos:	Teóricas: 40	Prácticas: 40	Estudio Independiente: 16
	Total de horas: 96		Créditos: 6
Tipo de curso:	Teórico	Teórico-práctico <input checked="" type="checkbox"/>	Práctico
Competencias del perfil de egreso a la que aporta	Capacidad de aplicación de la formación científica, organización y coordinación de proyectos, que le permitan hacer uso óptimo de las complejas tecnologías de computo modernas, así como desarrollar nuevos modelos, procedimientos y sistemas para la obtención, manejo y presentación de datos geográficos requeridos para la solución de problemas en el aprovechamiento del suelo, de los recursos naturales y la infraestructura del país.		
Unidades de aprendizaje relacionadas	Programación I, Programación e Ingeniería de software I, Programación e Ingeniería de software III, Tratamiento de la Información.		
Responsables de elaborar y/o actualizar el programa:	Dr. Miguel Contreras Montoya, M.C. Gerardo Gálvez Gámez, Lic. Rogelio Prieto Alvarado, M.C. José Guadalupe Quintero Armenta, MC. Edy López Cervantes, MC. Rodrigo García Carlos, Dr. Inés Fernando Vega López, LI. Diego Alonso Gastélum Chavira, MC. Gerardo Beltrán Gutiérrez, Lic. Cosme Adrián López Inda, M.C. Rodrigo García Carlos, Dr. Juan Martín Aguilar Villegas.		
Fecha de:	Elaboración: Febrero del 2012		Actualización: Febrero del 2012
2. PROPÓSITO			
Solucionar problemas aplicando técnicas algorítmicas bajo una metodología Orientada a Objeto, utilizando las Estructuras de Datos en un lenguaje de alto nivel.			
3. SABERES			
Teóricos:	<ul style="list-style-type: none"> • Conocer el concepto general de la metodología de la programación orientada a objetos (POO). • Comprender los diferentes Arreglos, Cadenas y Archivos. • Identificar las diferentes estructuras de datos: Listas simples, listas enlazadas, pilas, colas y árboles, empleadas en la creación de programas. • Conocer los conceptos principales de las estructuras de datos árboles y Gráficos, para la solución de problemas. • Comprender el concepto de relación y dígrafos para la resolución de diferentes problemas. 		

<p>Prácticos:</p>	<ul style="list-style-type: none"> • Determinar la metodología de la programación orientada a objetos (POO). • Desarrollar aplicaciones que mejoren el desempeño de aplicaciones utilizando Arreglos, Cadenas y Archivos. • Aplicar las diferentes estructuras de datos: Listas simples, listas enlazadas, pilas, colas y árboles, para la creación de programas que solucionen problemas utilizando las estructuras. • Manejar los conceptos principales de las estructuras de datos árboles y Gráficos, para la solución de problemas. • Utilizar el concepto de relación y dígrafos para la resolución de diferentes problemas.
<p>Actitudinales:</p>	<ul style="list-style-type: none"> • Valorar el papel de la ciencia y la tecnología en la programación dirigida a objetos. • Disposición al trabajo colectivo. • Cultivar la disciplina de la lectura científica. • Desarrollar la ética profesional. • Reflexividad ante las diferentes propuestas técnicas y metodológicas. • Atención a la actualización profesional.

4. CONTENIDO TEMÁTICO

1. INTRODUCCION A LA PROGRAMACIÓN ORIENTADA A OBJETOS

- 1.1. Conceptos básicos.
- 1.2. El entorno de desarrollo.
- 1.3. Características de laprogramación orientada aobjetos.
 - 1.3.1. Objetos.
 - 1.3.2. Mensajes.
 - 1.3.3. Métodos.
 - 1.3.4. Clases.
 - 1.3.5. Abstracción.
 - 1.3.6. Encapsulamiento.
 - 1.3.7. Herencia.
 - 1.3.8. Polimorfismo.
- 1.4. Estructura de un programa.

2. ARREGLOS,CADENAS Y ARCHIVOS

- 2.1. Arreglos unidimensionales
- 2.2. Arreglos multidimensionales
- 2.3. Arreglos como parámetros
- 2.4. Cadenas
- 2.5. Algoritmos de ordenamiento
 - 2.5.1. Intercambio
 - 2.5.2. Selección
 - 2.5.3. Inserción
 - 2.5.4. Shell

- 2.5.5. Quicksort
- 2.6. Algoritmos de Búsqueda
 - 2.6.1. Secuencial
 - 2.6.2. Binaria
- 2.7. Ordenación de objetos
- 2.8. Ordenación y Búsqueda en Archivos

3. ESTRUCTURAS LINEALES

- 3.1. Pilas
 - 3.1.1. Conceptos
 - 3.1.2. Operaciones en pila
- 3.2. Recursividad
 - 3.2.1. Concepto de Recursividad
 - 3.2.2. Utilización de Recursividad
- 3.3. Colas
 - 3.3.1. Conceptos
 - 3.3.2. Tipos de colas
 - 3.3.2.1. Simples
 - 3.3.2.2. Circulares
 - 3.3.2.3. Dobles
 - 3.3.3. Operaciones
- 3.4. Listas (Conceptos, Operaciones y Recorridos)
 - 3.4.1. Simplemente Enlazadas
 - 3.4.2. Doblemente enlazadas
 - 3.4.3. Circulares
 - 3.4.4. Multilistas
- 3.5. Colecciones en el Lenguaje de Programación

4. ESTRUCTURAS NO LINEALES

- 4.1. Árboles
 - 4.1.1. Conceptos fundamentales
 - 4.1.2. Árboles binarios
 - 4.1.3. Recorridos sobre árboles
 - 4.1.4. Árboles Parcialmente ordenados
 - 4.1.5. Árboles binarios de búsqueda.
 - 4.1.6. Otros tipos de árboles
- 4.2. Grafos
 - 4.2.1. Conceptos fundamentales
 - 4.2.2. Grafos dirigidos y no dirigidos
 - 4.2.3. Recorrido en profundidad y anchura
 - 4.2.4. TDA grafo
 - 4.2.5. Representaciones

5. RELACIONES Y DÍGRAFOS

- 5.1. Introducción
- 5.2. Relaciones y dígrafos
- 5.3. Trayectorias en relaciones y dígrafos

5.4. Propiedades de las relaciones
 5.5. Relaciones de equivalencia

5. ACCIONES ESTRATÉGICAS PARA EL APRENDIZAJE

- Exposición de introducción al tema, así como de los antecedentes y vigencia del mismo en cada unidad.
- Lecturas de investigación para complementar la comprensión del tema.
- Análisis y debate grupal sobre el tema.
- Transferencia de contenidos temáticos mediante los medios electrónicos.
- Planteamiento y solución de problemas concretos.
- Trabajos de investigación y redacción de resúmenes.
- Solución de problemas extra clase.
- Trabajo colectivo de exposición.

6. EVALUACIÓN DEL APRENDIZAJE

6.1. Evidencias de aprendizaje	6.2. Criterios de desempeño	6.3. Calificación y acreditación
1. Resolución de ejercicios en clase en forma individual y grupal 2. Resolución de ejercicios de tarea 3. Debate en torno a las dudas de los estudiantes 4. Resolución de problemas utilizando herramientas de cómputo. 5. Examen.	<ul style="list-style-type: none"> • Comprensión y profundización de conceptos teóricos. • Capacidad de trabajo colectivo intelectual y práctico. • Capacidad de exposición y dominio temático. • Capacidad de análisis, de redacción y síntesis de la investigación bibliográfica. • Planeación y desarrollo metodológico en la solución de problemas. • Capacidad de responder de manera precisa, clara y completa los reactivos de exámenes en forma oral y escrita. 	<ul style="list-style-type: none"> • Asistencia y participación 20% • Tareas 30% • Examen 50%

7. FUENTES DE INFORMACIÓN

Fuentes de Información Básica:

- CAIRO-GUARDATI. *Estructura de datos*, 3ª Edición, McGraw Hill, 2003.
- JOYAMS AGUILAR, LUIS. *Fundamentos de programación, algoritmos, estructura de datos y objetos*, McGraw Hill 2003
- MARTÍ, NARCISSO, et al. *Estructuras de datos y métodos algorítmicos*, Pearson Educación, 2004.
- Heileman. *Estructuras de datos, algoritmos y programación orientada a objetos*. Mc Graw Hill, 2001. 1997.}
- Giménez Cánovas. «*Algoritmos y Estructuras de Datos, Volumen I: Estructuras de Datos*». Diego Marín, 2003.
- Domingo Giménez Canovas, Joaquín Cervera López, Ginés García Mateos y Norberto Marín Pérez. «*Algoritmos y Estructuras de Datos, Volumen II Algoritmos*». Diego Marín, 2003. Graw-Hill. Madrid, España. Cuarta edición.
- SCHILDT HERBERT (2003), *C#. Manual de referencia*. Mc Graw Hill, Primera Edición.
- CEBALLOS, Fco. Javier (2008), *C# Lenguajes y Aplicaciones*, Edit Alfaomega, México D.F , Segunda Edición.

Fuentes de Información Complementaria

- TANENBAUM, ANDREW, M., LANGSMAN, *et al.*, *Estructura de datos con c/c++*, Prentice Hall, 1995.
- Bowman C. Algoritmos y estructuras de datos: aproximación en C. Primera edición en español. Oxford University Press México S.A. de C.V. México. 1999

8. PERFIL DEL PROFESOR

El profesor debe de contar con posgrado en el área de las ciencias de la información con orientación en Programación o en alguna disciplina de la informática. Debe de contar con experiencia docente y en trabajos de investigación o aplicación de los métodos de programación para la generación de información geográfica.