



# UNIVERSIDAD AUTÓNOMA DE SINALOA

## FACULTAD DE INGENIERÍA MOCHIS LICENCIATURA EN INGENIERÍA EN SOFTWARE

### PROGRAMA DE ESTUDIO



1. DATOS DE IDENTIFICACIÓN			
UNIDAD DE APRENDIZAJE O MÓDULO:	PROGRAMACIÓN ORIENTADA A OBJETOS		
Clave:			
Ubicación:	Semestre IV	Área: Profesionalizante	
Horas y créditos:	Teóricas: 30	Prácticas: 50	Estudio Independiente: 80
	Total de horas: 160		Créditos: 10
Competencia(s) del perfil de egreso al que aporta:	<p>CG1. Desarrolla su potencial intelectual para generar el conocimiento necesario en la resolución de problemas y retos, tanto de su vida individual y como parte de una comunidad, con sentido de pertinencia, identidad y empatía.</p> <p>CG3. Ejerce su conocimiento ponderando los valores éticos para brindar mayores beneficios a la comunidad, con respeto a la ley y los códigos que dirigen su desempeño.</p> <p>CG7. Cultiva el compañerismo, el trabajo en equipo y la coordinación de esfuerzos bajo la aspiración de mejorar las tareas académicas, los entornos laborales y la convivencia social en beneficio para la consecución de metas que impactan en las formas de entablar y mantener relaciones humanas positivas.</p> <p>CE2. Desarrolla habilidades de abstracción y la expresión de formalismos, además de proporcionar conocimientos específicos fundamentales para la informática y la computación.</p> <p>CE11. Construye las piezas de software que hacen posible el funcionamiento de las computadoras en diferentes niveles operativos.</p> <p>CE20. Desarrolla software para diferentes tipos de aplicaciones utilizando técnicas, metodologías y paradigmas de programación en el contexto de los ciclos de vida del software y técnicas de gestión de proyectos cuidando conservar los atributos de calidad requeridos.</p>		
Unidades de aprendizaje relacionadas:	Algoritmos y lógica computacional, programación i, programación ii, estructura de datos, lenguajes de programación, desarrollo de aplicaciones web		
Responsable(s) de elaborar el programa:	Dr. Juan Francisco Figueroa Pérez		Fecha: 31/01/2025
Responsable(s) de actualizar el programa:			Fecha:
2. PROPÓSITO			
Identificar y aplicar los diferentes elementos de la programación orientada a objetos para el desarrollo de software			
3. SABERES			
Teóricos:	Conoce e identifica los conceptos básicos de la programación orientada a objetos. Conoce e identifica los elementos básicos del diseño e implementación de clases. Conoce e identifica elementos avanzados del diseño de sistemas orientados a objetos.		
Prácticos:	Desarrolla programas de software aplicando términos de clases de objetos y abstracciones en forma de herencia y polimorfismo. Elabora programas que utilizan elementos básicos del diseño e implementación de clases. Elabora prototipos de sistemas que utilizan elementos		



	avanzados del diseño orientado a objetos.
Actitudinales:	Comunicación oral y escrita en la propia lengua. Habilidades de gestión de la información (habilidad para buscar y analizar información proveniente de diversas fuentes). Capacidad de crítica y autocrítica. Trabajo en equipo. Compromiso ético. Capacidad para aprender a trabajar de forma autónoma.
<b>4. CONTENIDOS</b>	
<p>1.- Programación Estructurada</p> <ul style="list-style-type: none"><li>1.1 Estructura básica de un programa</li><li>1.2 Condiciones</li><li>1.3 Ciclos</li><li>1.4 Procedimientos y funciones</li></ul> <p>2.- Clases y objetos</p> <ul style="list-style-type: none"><li>2.1 Entidades, clases, objetos, atributos y métodos</li><li>2.2 Constructores y destructores</li><li>2.3 Visibilidad</li><li>2.4 Sobrecarga</li></ul> <p>3.- Herencia y polimorfismo</p> <ul style="list-style-type: none"><li>3.1 Herencia</li><li>3.2 Polimorfismo</li><li>3.3 Interfaces</li></ul>	
<b>5. ACTIVIDADES PARA DESARROLLAR LAS COMPETENCIAS</b>	
<p><i>Actividades del docente:</i></p> <ul style="list-style-type: none"><li>● Selecciona un tema y guía la conducción de las preguntas que los jóvenes abordarán en la clase: ¿Qué? ¿Quién? ¿Cómo? ¿Cuándo? ¿Por qué? ¿Para qué? Sobre la estructura básica de un programa, condiciones, ciclos, procedimientos y funciones</li><li>● Selecciona un tema y guía la conducción de las preguntas que los jóvenes abordarán en la clase: ¿Qué? ¿Quién? ¿Cómo? ¿Cuándo? ¿Por qué? ¿Para qué? Sobre entidades, clases, objetos, atributos y métodos, constructores y destructores, visibilidad y sobrecarga</li><li>● Explica los diferentes elementos relacionados con entidades, clases, objetos, atributos y métodos, constructores y destructores, visibilidad y sobrecarga</li><li>● Elabora práctica integradora sobre entidades, clases, objetos, atributos y métodos, constructores y destructores, visibilidad y sobrecarga</li><li>● Selecciona un tema y guía la conducción de las preguntas que los jóvenes abordarán en la clase: ¿Qué? ¿Quién? ¿Cómo? ¿Cuándo? ¿Por qué? ¿Para qué? Sobre herencia, polimorfismo e interfaces</li><li>● Explica los diferentes elementos de la herencia, polimorfismo e interfaces</li><li>● Explica la en que consiste la practica integradora sobre herencia, polimorfismo e interfaces</li></ul>	
<p><i>Actividades del estudiante:</i></p> <ul style="list-style-type: none"><li>❖ Dan respuesta a las preguntas haciendo referencias a datos, ideas y detalles expresados en las lectura.</li><li>❖ Revisa elementos relacionados con entidades, clases, objetos, atributos y métodos, constructores y destructores, visibilidad, sobrecarga, herencia, polimorfismo e interfaces</li><li>❖ Elabora práctica integradora sobre entidades, clases, objetos, atributos y métodos, constructores y</li></ul>	



# UNIVERSIDAD AUTÓNOMA DE SINALOA

## FACULTAD DE INGENIERÍA MOCHIS LICENCIATURA EN INGENIERÍA EN SOFTWARE



### PROGRAMA DE ESTUDIO

destructores, visibilidad y sobrecarga, herencia, polimorfismo e interfaces				
6. EVALUACIÓN DE LAS COMPETENCIAS				
6.1. Criterios de desempeño		6.2 Portafolio de evidencias		
Lecturas obligadas y estudio individual, resolución de prácticas, problemas o ejercicios propuestos por profesor, preparación de trabajos individuales, exposiciones de trabajos en clase, portafolio de evidencia y heteroevaluación.		Reportes de lectura, evaluación de ejercicios prácticos, y práctica final integradora		
6.3. Calificación y acreditación:				
Parcial: Evaluaciones parciales y ejercicios de programación		Final: Entrega de práctica final integradora		
7. RECURSOS DIDÁCTICOS				
Computadora, pintarrón, cañon, diapositivas, videos, exposiciones y ejemplos de código				
8. FUENTES DE INFORMACIÓN				
Bibliografía básica				
Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
Deitel & Deitel	Como programar en Java	Pearson	2012	
Booch Grady	Object-Oriented Analysis and Design with Applications	Pearson	2007	
Joyanes Aguilar Luis	Pascal	Mc Graw Hill	2009	
Joyanes Aguilar Luis	Programación Orientada a Objetos	Mc Graw Hill	1996	
Texeira Steve, Pacheco Xavier	Delphi 5	Inprise Cor	1999	
Bibliografía complementaria				
Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible
9. PERFIL DEL DOCENTE				
Licenciado en informática, licenciado en sistemas computacionales, licenciado en ingeniería de software, ingeniero en sistemas computacionales, ingeniero en computación				