

<p align="center">UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA</p>  <p align="center">Clave: 08MSU0017H</p> <p align="center">FACULTAD DE INGENIERÍA</p>  <p align="center">Clave: 08USU4053W</p> <p align="center">PROGRAMA ANALÍTICO DE LA UNIDAD DE APRENDIZAJE: PROGRAMACIÓN II</p>	DES:	Ingeniería
	Programa(s) Educativo(s):	Ingeniería de Software
	Tipo de materia:	Obligatoria
	Clave de la materia:	IS0603
	Cuatrimestre:	6
	Área en plan de estudios:	Específica
	Créditos	5.4
	Total de horas por semana:	4 horas
	<i>Teoría: Virtual</i>	4 horas
	<i>Práctica</i>	
	<i>Taller:</i>	
	<i>Laboratorio:</i>	
	<i>Prácticas complementarias:</i>	
	<i>Trabajo extra clase:</i>	4 horas
	Total de horas por cuatrimestre:	96 horas
Fecha de actualización:	Octubre de 2015	
<i>Materia requisito:</i>	IS0503 - Programación I	

PROPÓSITO DEL CURSO:

Introducir los principios y la práctica del desarrollo de software utilizando el paradigma orientado a objetos, mediante la aplicación y práctica con un lenguaje de programación de éste paradigma, para la solución de problemas algorítmicos desde un enfoque distinto.

COMPETENCIAS	DOMINIOS COGNITIVOS.	RESULTADOS DE APRENDIZAJE.
<p>Competencias específicas</p> <ul style="list-style-type: none"> ● Sistemas computacionales: Aplica los fundamentos de las Ciencias de la Computación que dan soporte al diseño y construcción de software mediante un proceso metodológico atendiendo a los estándares internacionales. ● Diseño y Modelado de Software: Transforma los requerimientos del cliente en una especificación formal y documentada, diseñando y modelando soluciones de 	<ol style="list-style-type: none"> 1. Introducción al Paradigma Orientado a Objetos <ol style="list-style-type: none"> 1.1 Clases 1.2 Objetos 1.3 Características del paradigma 1.4 Modificadores de acceso public, private y protected 2. Métodos <ol style="list-style-type: none"> 2.1 Declaración de métodos 2.2 Concepto de método. 2.3 Atributos const y static. 2.4 Llamadas a métodos y uso de argumentos. 2.5 Tipos de métodos. 2.6 Autorreferencia 	<ul style="list-style-type: none"> ✓ Identifica los principios del paradigma orientado a objetos para su implementación en el desarrollo de proyectos. ✓ Diseña interfaces de comunicación humano-computadora tomando en cuenta las

<p>técnicas y metodologías que responden a estándares internacionales.</p> <ul style="list-style-type: none"> ● Calidad de Software: Selecciona las técnicas adecuadas para asegurar la calidad y seguridad del software durante su planeación, diseño, construcción y mantenimiento mediante la aplicación de metodologías, técnicas y estándares internacionales, que aplicados sistemáticamente garanticen los requerimientos del cliente. ● Ingeniería del Proceso de Software: Adquiere las herramientas y metodologías para llevar a cabo las etapas de la ingeniería del proceso de software para construir o mejorar proyectos mediante métodos y procedimientos para lograr su propósito. 	<p>2.7 Forma de pasar argumentos.</p> <p>3. Constructores y destructores</p> <p>3.1. Conceptos de métodos constructor y destructor.</p> <p>3.2. Declaración de métodos constructor y destructor.</p> <p>3.3. Aplicaciones de constructores y destructores.</p> <p>3.4. Tipos de constructores y destructores.</p> <p>4. Herencia</p> <p>5.1 Introducción a la herencia.</p> <p>5.2 Herencia simple.</p> <p>5.3 Herencia múltiple.</p> <p>5.4 Clase base y clase derivada.</p> <p>5.5 Parte protegida.</p> <p>5.6 Redefinición de los miembros de las clases derivadas.</p> <p>5.7 Clases virtuales y visibilidad.</p> <p>5.8 Constructores y destructores en clases derivadas.</p> <p>5.9 Aplicaciones.</p> <p>5. Polimorfismo y reutilización</p> <p>5.1. Concepto del polimorfismo.</p> <p>5.2. Clases abstractas.</p> <p>5.3. Definición de una interfaz.</p> <p>5.4. Implementación de la definición de una interfaz.</p> <p>5.5. Reutilización de la definición de una interfaz.</p> <p>5.6. Definición y creación de paquetes / librería.</p> <p>5.7. Reutilización de las clases de un paquete / librería.</p> <p>5.8. Clases genéricas (Plantillas).</p> <p>6. Excepciones</p> <p>6.1. Propagación</p> <p>6.2. Gestión de Excepciones</p>	<p>necesidades de pertenencia de las personas.</p> <ul style="list-style-type: none"> ✓ Utiliza la notación formal de diseño describiendo de manera precisa lo que el proyecto de software debe hacer. ✓ Utiliza métricas en la toma de decisiones durante el proceso de ingeniería de software. ✓ Implementa métodos y estrategias coadyuvantes a la construcción de software seguro, garantizando su calidad.
--	---	--

OBJETOS DE APRENDIZAJE	METODOLOGÍA (Estrategias, secuencias, recursos didácticos)	EVIDENCIAS DE APRENDIZAJE.
<ol style="list-style-type: none"> 1. Introducción al Paradigma Orientado a Objetos 2. Métodos 3. Constructores y destructores 4. Herencia 5. Polimorfismo y reutilización 6. Excepciones 	<ol style="list-style-type: none"> 1. Foro de Bienvenida, Documentos de lectura, Videos. 2. Videos y documentos de lectura. 3. Videos y documentos de lectura. 4. Videos y documentos de lectura. 5. Videos y documentos de lectura. 6. Videos y documentos de lectura. 	<ol style="list-style-type: none"> 1. Elaboración de resúmenes, tareas y cuestionarios. 2. Elaboración de tareas y ejercicios en línea. 3. Elaboración de tareas y ejercicios en línea. 4. Elaboración de tareas y ejercicios en línea. 5. Elaboración de tareas y ejercicios en línea. 6. Elaboración de tareas en línea.
FUENTES DE INFORMACIÓN (Bibliografía, direcciones electrónicas)		EVALUACIÓN DE LOS APRENDIZAJES (Criterios e instrumentos)
<p>Peter Coad, Object Oriented Programming, Yourdon Press Computer Series, 1993.</p> <p>Robert Lafore, Object Oriented Programming in C++, 2001</p>		<p>Se toma en cuenta para integrar la calificación total:</p> <p>Introducción al Paradigma Orientado a Objeto</p> <ul style="list-style-type: none"> ● Cuestionario sobre Clases y Objetos: 5% ● Cuestionario sobre el Paradigma: 10% ● Tarea sobre modificadores de acceso: 10% <p>Métodos</p> <ul style="list-style-type: none"> ● Ejercicios con métodos: 10% ● Ejercicios con métodos auto referenciados: 5% <p>Constructores y Destructores</p> <ul style="list-style-type: none"> ● Tarea de investigación de Constructores y Destructores: 10% ● Ejercicios con Constructores y Destructores: 10%

