

<p style="text-align: center;">UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA</p>  <p style="text-align: center;">UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA</p> <p style="text-align: center;">PROGRAMA ANALÍTICO DE LA UNIDAD DE APRENDIZAJE:</p> <p style="text-align: center;">INTRODUCCIÓN A LA PROGRAMACIÓN</p>	DES:	INGENIERÍA
	Programa Educativo	Ingeniería en Ciencias de la Computación
	Tipo de materia (Obli/Opta):	Obligatoria
	Clave de la materia:	CI174
	Semestre:	1
	Área en plan de estudios (G, E):	Ciencias de la Ingeniería
	Total, de horas por semana:	4
	<i>Teoría: Presencial o Virtual</i>	3
	<i>Laboratorio o Taller:</i>	1
	<i>Prácticas:</i>	0
	<i>Trabajo extra-clase:</i>	0
	Créditos Totales:	4
	Total, de horas semestre (x 16 sem):	64
	Fecha de actualización:	Febrero 2023
Prerrequisito (s):	Ninguno	
Realizado por:	Comité de Rediseño Curricular	

PROPÓSITO DEL CURSO

La programación de aplicaciones de software es una actividad fundamental en la actualidad. La correcta comprensión de las estructuras de datos y los principios de aplicación de algoritmos para la solución de problemas por medio de las ciencias de la computación es fundamental para establecer las bases de conocimiento necesarias para la comprensión de conceptos mas avanzados.

El propósito del curso es proporcionar conceptos y herramientas de aplicación por las cuales una computadora resuelve un problema concreto siguiendo una secuencia de instrucciones en un determinado lenguaje de programación.

COMPETENCIAS (Tipo Y Nombre de la competencias que nutre la materia y a las que contribuye)	DOMINIOS COGNITIVOS (Objetos de estudio, temas y subtemas)	RESULTADOS DE APRENDIZAJE. (Por objeto de estudio).
<p>El curso promueve las siguientes competencias:</p> <p>BÁSICAS:</p> <p>SOLUCION DE PROBLEMAS</p> <ul style="list-style-type: none"> • Aplica las diferentes técnicas de observación para la solución de problemas. • Distingue los diversostipos de sistemas • Aplica la tecnología a la solución de 	<p>I. CONCEPTOS FUNDAMENTALES DE PROGRAMACION.</p> <p>1.1. Computadoras y programas 1.2. Diagramas de flujo 1.3. Pseudocódigo</p>	<p>Analiza y explica el comportamiento de programas simples</p> <p>Identifica y utiliza herramientas para la representación de programas</p>

<p>problemáticas</p> <ul style="list-style-type: none"> • Aplica el enfoque sistémico en diversos contextos • Crea soluciones innovadoras y utiliza formas no convencionales en la solución de problemas 		
<p>PROFESIONALES:</p> <p>CIENCIAS FUNDAMENTALES DE LA INGENIERÍA</p> <ul style="list-style-type: none"> • Utiliza las matemáticas como herramientas para solución de problemas en ingeniería 	<p>II. VARIABLES Y TIPOS DE DATOS</p> <p>2.1 Concepto de variable y constante 2.2 Tipos de datos y casting 2.3 Inicialización de variables y asignación 2.4 Expresiones matemáticas y lógicas</p>	<p>Distingue entre variable y constante y su aplicación</p> <p>Identifica y describe los tipos de datos básicos.</p> <p>Distingue entre las diferentes formas de inicialización y asignación de valores a variables</p> <p>Escribe programas donde utiliza adecuadamente expresiones matemáticas y lógicas en el contexto de las variables y constantes</p>
<p>ESPECIFICAS:</p> <p>FUNDAMENTOS DE CIENCIAS DE LA COMPUTACIÓN</p> <ul style="list-style-type: none"> • Diseña y aplica algoritmos, estructuras y representación de datos para soluciones computacionales • Distingue los fundamentos de un sistema computacional • Aplica las bases de los lenguajes de programación para generar aplicaciones óptimas • Aplica los tópicos de paradigmas de programación: estructurado, orientado a objetos. Lógico, funcional, entre otros para la creación de aplicaciones óptimas 	<p>III. ESTRUCTURAS DE CONTROL</p> <p>3.1 Estructuras condicionales 3.1.1 Sentencias If-else 3.1.2 Sentencias case-switch 3.2 Estructuras iterativas 3.2.1 Ciclos while 3.2.2 Ciclos for</p>	<p>Analiza y explica el comportamiento de las estructuras básicas de control</p> <p>Escribe programas simples seleccionando las estructuras de control adecuadas para cada caso</p>

	IV. FUNCIONES 4.1 Definición y declaración de funciones 4.2 Alcance de la función y paso de parámetros 4.3 Concepto de recursión	Explica los tipos de función y su aplicación Escribe programas simples haciendo uso de funciones simples Describe el concepto de recursión y da ejemplos de su uso
	V. ESTRUCTURAS BASICAS DE DATOS. 5.1. Arreglos 5.1.1. Vectores 5.1.2. Matrices 5.2. Cadenas 5.3. Tipos de datos compuestos (estructuras o registros) 5.4 . Apuntadores	Enuncia el concepto de arreglo, cadenas y tipos de datos compuestos Escribe programas simples haciendo uso de estructuras básicas de datos como arreglos, cadenas y tipos de datos compuestos
	VI. DISPOSITIVOS DE ENTRADA Y SALIDA. 6.1. Dispositivos de entrada 6.2. Dispositivos de salida 6.3. Archivos	Distingue los diferentes mecanismos de entrada y salida para un programa de computo Escribe programas simples haciendo uso de dispositivos de entrada y salida incluyendo archivos

OBJETO DE ESTUDIO	METODOLOGIA (Estrategias, secuencias, recursos didácticos)	EVIDENCIAS DE APRENDIZAJE.
I. CONCEPTOS FUNDAMENTALES DE PROGRAMACION II. VARIABLES Y TIPOS DE DATOS III. ESTRUCTURAS DE CONTROL IV. FUNCIONES V. ESTRUCTURAS BASICAS DE DATOS VI. DISPOSITIVOS DE ENTRADA Y SALIDA	Aprendizaje interactivo (exposición del profesor) Grupo de discusión. Auto aprendizaje (búsqueda y análisis de información) Inductivo <ul style="list-style-type: none"> • Observación • Comparación Deductivo <ul style="list-style-type: none"> • Aplicación • Comprobación • Demostración 	<ul style="list-style-type: none"> • Tareas y ejercicios. • Escritura de programas de computo. • Exámenes escritos.

	<p>Sintético</p> <ul style="list-style-type: none"> • Recapitulación • Definición • Resumen • Esquemas • Modelos matemáticos • Conclusión. <p>Material de Apoyo didáctico: Recursos</p> <ul style="list-style-type: none"> • Manual de Instrucción • Materiales gráficos: artículos, libros, diccionarios, etc. • Cañón • Pintarrones • Equipo de computo 	
--	--	--

FUENTES DE INFORMACIÓN (Bibliografía, Direcciones electrónicas)	EVALUACIÓN DE LOS APRENDIZAJES (Criterios e instrumentos)
<p>1. Concepts of Programming Languages. Robert W. Sebesta. Cualquier edición.</p>	<p>INSTRUMENTOS:</p> <ul style="list-style-type: none"> • Exámenes • Solución de ejercicios (aplicación de conocimientos) • Lista de cotejo (Respeto y participación al trabajo dentro del salón de clase, interés por la asignatura) <p>CRITERIOS DE DESEMPEÑO:</p> <ul style="list-style-type: none"> • Los programas desarrollados deberán estar completos. • Las tareas se deberán entregar en tiempo y forma <p style="margin-left: 40px;">• Exámenes escritos:</p> <p>Se realizan 3 exámenes escritos durante el semestre y las fechas se establecen por la secretaría académica</p> <p>Se toma en cuenta para integrar calificaciones parciales:</p> <p>Exámenes 60% Tareas y practicas 40%</p> <p>Fecha de exámenes parciales: 1º. Parcial: 2º. Parcial: 3^{er} Parcial:</p>

LAS ACTIVIDADES NO REALIZADAS EN TIEMPO Y FORMA SE CALIFICAN CON CERO.

Nota: para acreditar el curso se deberá tener calificación aprobatoria tanto en la teoría como en las prácticas.

Cronograma del Avance Programático

S e m a n a s

Unidades de aprendizaje	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
I. Conceptos fundamentales de programación																
II. Variables y tipos de datos																
III. Estructuras de control																
IV. Funciones																
V. Estructuras básicas de datos																
VI. Dispositivos de entrada y salida																