

<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 ANALITICO DE LA UNIDAD DE APRENDIZAJE:</p> <p style="text-align: center;">LENGUAJE ENSAMBLADOR</p>	DES:	Ingeniería
	Programa(s) Educativo(s):	Ingeniería en Ciencias de la Computación
	Tipo de materia:	Obligatoria
	Clave de la materia:	IA578
	Semestre:	5°
	Área en plan de estudios:	Ingeniería Aplicada
	Créditos	4
	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:	64
	Total de horas semestre	64
Fecha de actualización:	Febrero 2023	
Prerrequisito (s):	Arquitectura de Computadoras	

PROPÓSITO DEL CURSO

El curso proporciona las herramientas fundamentales para que el estudiante sea capaz de desarrollar programas de bajo nivel, aplicando algunas de las operaciones básicas que los sistemas de software desarrollan para la gestión y control de los recursos computacionales. Los conocimientos y habilidades adquiridos en este curso, son antecedentes para que los estudiantes aborden en cursos posteriores que niveles abstractos de la organización de las computadoras y los sistemas operativos componentes elementales de cualquier sistema computacional.

COMPETENCIAS (Tipo y nombre de las competencias) DOMINIOS	DOMINIOS COGNITIVOS (Objetos de aprendizaje, temas y subtemas)	RESULTADOS DE APRENDIZAJE
<p>El curso promueve las siguientes competencias:</p> <p>Competencias Básicas:</p> <ul style="list-style-type: none"> • Solución de problemas. • Trabajo en equipo y liderazgo. • Comunicación. <p>Competencias Profesionales:</p> <ul style="list-style-type: none"> • Proyectos de Ingeniería • Ingeniería de Proceso 	<p>UNIDAD I: Introducción a la arquitectura de computadoras</p> <p>1.1. Procesador 1.2. Memoria 1.3. Registros 1.4. Unidad Aritmético-Lógica 1.5. Unidad de Control 1.6. Movimiento de datos 1.7. Manipulación de datos</p>	<p>Identifica conceptos del modelo de computación implícito en lenguajes de programación de bajo nivel</p> <p>Ilustra a través de pseudocódigo y uso de mnemónicos la implementación conceptual de programas de bajo nivel.</p>
	<p>UNIDAD II: ELEMENTOS DE LA PROGRAMACIÓN EN LENGUAJE ENSAMBLADOR</p>	<p>Describe el lenguaje de más bajo nivel que puede ser utilizado para asignar</p>

	<p>2.1. Juego de instrucciones: datos, aritmético-lógicas, control de flujo del programa, etc.</p> <p>2.2. Estructura de Archivos COM</p> <p>2.3. Estructura de Archivos EXE</p> <p>2.4. Herramientas para depuración de Programas</p>	<p>tareas al procesador.</p> <p>Distingue las estructuras de programación disponibles en el lenguaje ensamblador.</p> <p>Identifica el nivel de soporte de herramientas de depuración de programas desarrollados en lenguaje ensamblador</p>
	<p>UNIDAD III: PROGRAMACIÓN BASE DE ENTRADA-SALIDA</p> <p>3.1. Servicio del DOS para dispositivo de entrada</p> <p>3.2. Servicios del DOS para dispositivo de salida</p>	<p>Diseña e implementa programas que le permitan comprender algunas de las tareas básicas desarrolladas por sistemas operativos monousuario</p>
	<p>UNIDAD IV: MODULARIDAD</p> <p>4.1. Procedimientos</p> <p>4.2. Macros</p> <p>4.3. Multimódulos</p>	<p>Modela soluciones y escribe código modular que da claridad al programa e introduce el concepto de reutilización de software.</p>
	<p>UNIDAD V: PROGRAMACION AVANZADA DE ENTRADASALIDA</p> <p>5.1. Servicios del BIOS para dispositivo de entrada</p> <p>5.2. Servicios del BIOS para dispositivo de salida</p> <p>5.3. Área de datos del BIOS para dispositivo de entrada.</p> <p>5.4. Video mapeado en memoria.</p>	<p>Implementa programas de comunicación con los dispositivos de entrada-salida, con instrucciones que toman control de eventos a nivel hardware</p>
	<p>UNIDAD VI: TÓPICOS AVANZADOS</p> <p>6.1. Ensamblador y lenguajes de alto nivel</p> <p>6.2. Programación residente</p>	<p>Desarrolla programas para comunicar eventos de bajo nivel con programas desarrollados en lenguajes de alto nivel.</p> <p>Ilustra y analiza las características de programas que se cargan de manera permanente en la memoria de una computadora.</p>

OBJETO DE APRENDIZAJE	METODOLOGIA (Estrategias, secuencias, recursos didácticos)	EVIDENCIAS DE APRENDIZAJE
<p>Introducción a la arquitectura de computadoras</p> <p>ELEMENTOS DE LA PROGRAMACIÓN EN LENGUAJE ENSAMBLADOR</p> <p>PROGRAMACIÓN BASE DE ENTRADA-SALIDA</p> <p>MODULARIDAD</p> <p>PROGRAMACIÓN AVANZADA DE ENTRADASALIDA</p> <p>TÓPICOS AVANZADOS</p>	<ol style="list-style-type: none"> 1. Encuadre 2. Aprendizaje interactivo (exposición del profesor) 3. Discusión dirigida, uso de mapas conceptuales y diagramación 4. Técnicas P.N.I. 5. Trabajo individual en solución de ejercicios vía desarrollo de programas y herramientas de depuración <ol style="list-style-type: none"> 1. Aprendizaje interactivo (exposición del profesor) 2. Trabajo colaborativo. Consulta de literatura de referencia para el desarrollo de programas 3. Cuadros sinópticos 4. Trabajo individual en solución de ejercicios vía desarrollo de programas y herramientas de depuración <ol style="list-style-type: none"> 1. Aprendizaje interactivo (exposición del profesor) 2. Trabajo colaborativo. Consulta de literatura de referencia para el reforzamiento cognitivo de los tópicos avanzados. 3. Trabajo colaborativo. Solución de ejercicios demostrativos de los diferentes conceptos de los tópicos avanzados. 	<p>Informe por escrito describiendo el origen y evolución de las arquitecturas de computadoras monolíticas</p> <p>Contextualización de la información obtenida en un adecuado marco de referencia</p> <p>Presentaciones ppt</p> <p>Reportes técnicos producto de evaluación de desempeño de programas utilizando herramientas de depuración</p> <p>Reportes técnicos producto de implementación en código utilizando el lenguaje ensamblador</p> <p>Sustento teórico para fundamentar soluciones de código que desarrollan tareas ejecutivas en una arquitectura de computadora.</p> <p>Sustento teórico para extender y fortalecer conocimiento acerca de la implementación de código de bajo y alto nivel en una arquitectura de computadora.</p>

FUENTES DE INFORMACIÓN (Bibliografía, direcciones electrónicas)	EVALUACIÓN DE LOS APRENDIZAJES (Criterios e instrumentos)
<p>OØ Abel, Peter. (1996). <i>Lenguaje Ensamblador y Programación para IBM PC y compatibles</i>. Prentice Hall.</p> <p>OØ Goodfrey J. Terry. (1991). <i>Lenguaje Ensamblador para Microcomputadoras IBM</i>. Prentice Hall.</p> <p>OØ Scragg Greg W. (1992). <i>Computer Organization, A TopDown Approach</i>. Prentice Hall</p>	<p>El curso se evalúa en 3 momentos, las fechas se establecen por la secretaría académica:</p> <p>INSTRUMENTOS:</p> <p>Examen escrito Informes escritos Presentaciones ppt</p>

OØ Brey Barry B. y Del Arco Pérez Jorge Joel. (2001). *Los Microprocesadores de INTEL*. Prentice Hall.

Conocimientos: 40 (aspectos teóricos)
Habilidades: 45% (análisis, argumentación, redacción, uso de tecnología, comunicación, efectiva, , resolución de ejercicios con aplicación metodológica)
Valores y actitudes: 15% (colaboración, orden, lenguaje apropiado, respeto, puntualidad).

CRITERIOS DE DESEMPEÑO:

Los informes por escrito: valoran el nivel de argumentación en relación al hecho que se quiere demostrar. Manejo de lenguaje técnico, coherencia entre párrafos y global, redacción, ortografía y presentación.

Se utiliza una rúbrica para evaluación.

- **Exposición:** presentadas en orden lógico:
 1. Introducción resaltando el objetivo a alcanzar
 2. Desarrollo temático, responder preguntas y aclarar dudas
 3. Concluir.

• **Los trabajos extracurriculares**

Toda actividad complementaria al curso se podrá llevar a cabo en forma individual o por equipo según amerite el tema. Estos se reciben únicamente en tiempo y forma previamente establecidos.

• **Prácticas de Laboratorio**

Las actividades de práctica de laboratorio (programación y/o depuración) valoran el conocimiento aplicado. Se utiliza lista de cotejo para evaluación. Toda práctica debe generar un reporte técnico dentro del cual se evalúa, la revisión de literatura, el establecimiento de hipótesis (en su caso), la experimentación, la discusión de resultados y la argumentación de las conclusiones.

La acreditación del curso:

- Promedio de Calificaciones parciales: 60%
- Prácticas de laboratorio: 40%

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

Unidades de aprendizaje	Semanas																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
UNIDAD I: ARQUITECTURA DEL MICROPROCESADOR 8088	■	■	■														
UNIDAD II: ELEMENTOS DE LA PROGRAMACIÓN EN LENGUAJE ENSAMBLADOR				■	■	■											
UNIDAD III: PROGRAMACIÓN BASE DE ENTRADA- SALIDA							■	■									
UNIDAD IV: MODULARIDAD									■	■	■						
UNIDAD V: PROGRAMACIÓN AVANZADA DE ENTRADA- SALIDA												■	■	■			
UNIDAD VI: TÓPICOS AVANZADOS															■	■	■