

SINSTITUTO UNIVERSITARIO PUEBLA	HOJA:	1	DE	4
---------------------------------	-------	---	----	---

<b>ASIGNATURA: PROGRAMACIÓN II</b>	
<b>PROGRAMA ACADÉMICO: INGENIERIA EN MECATRÓNICA</b>	
<b>TIPO EDUCATIVO: INGENIERIA</b>	<b>MODALIDAD: MIXTA</b>
<b>SERIACIÓN: IM04</b>	<b>CLAVE DE LA ASIGNATURA: IM10</b>
<b>CICLO: SEGUNDO CUATRIMESTRE</b>	

<b>HORAS CON DOCENTE</b>	<b>HORAS INDEPENDIENTES</b>	<b>TOTAL DE HORAS</b>	<b>CRÉDITOS</b>
<b>60</b>	<b>68</b>	<b>128</b>	<b>8</b>

**TOTAL DE HORAS EN EL PERÍODO:** \_\_\_\_\_ 60 \_\_\_\_\_

**OBJETIVO GENERAL DE LA ASIGNATURA:**

Comprender la importancia del lenguaje ensamblador en el área de las ciencias computacionales a través del conocimiento de la arquitectura y el conjunto de instrucciones con que operan los microprocesadores de la familia Intel, y el desarrollo de programas en lenguaje ensamblador de los microprocesadores de Intel.

**VÍNCULOS DE LA ASIGNATURA CON LOS OBJETIVOS GENERALES DEL CURRÍCULUM:**

EL estudio de la Asignatura de Programación en el aspecto del Lenguaje del Ensamblador en las áreas de las Ciencias Computacionales que le ayudara al alumno para el desarrollo de programas en lenguaje ensamblador de los microprocesadores de Intel.

**PERFIL DEL DOCENTE REQUERIDO:**

Profesor con maestría y carrera en algún área de computación y con experiencia en el uso del lenguaje ensamblador para procesadores de la familia Intel.

JESUS ADRIAN BALLESTEROS XICOTENCATL  
**NOMBRE Y FIRMA DEL RESPONSABLE DEL PROGRAMA ACADÉMICO**  
01 DE MARZO DE 2007  
**FECHA DE ELABORACIÓN**

**ASIGNATURA: PROGRAMACIÓN II  
DEL PROGRAMA ACADÉMICO: INGENIERIA EN MECATRÓNICA**

HORAS ESTIMADAS	TEMAS Y SUBTEMAS	OBJETIVOS DE LOS TEMAS
5	<b>1. INTRODUCCIÓN AL LENGUAJE ENSAMBLADOR.</b> 1.1 Evolución de los lenguajes. 1.2 Aplicaciones del lenguaje ensamblador. 1.3 La familia de microprocesadores de Intel.	Definir y analizar la evolución de los lenguajes, como también sus aplicaciones y la familia de microprocesadores.
5	<b>2. ARQUITECTURA Y FUNCIONAMIENTO DE UN MICROPROCESADOR.</b> 2.1 Registros y banderas. 2.2 Unidades funcionales básicas. 2.3 Canales de datos, direcciones y control. 2.4 Proceso de arranque de una microcomputadora.	Definir los conceptos del funcionamiento de un microprocesador.
5	<b>3. ORGANIZACIÓN DE MEMORIA PRINCIPAL.</b> 3.1 Organización de un programa en memoria principal. 3.2 Tipos de direccionamiento.	Definir los conceptos de organización de memoria principal.
5	<b>4. INSTRUCCIONES BÁSICAS.</b> 4.1 Transferencia de datos. 4.2 Suma y resta. 4.3 Operaciones lógicas. 4.4 Corrimientos y rotaciones. 4.5 Multiplicación y división (auto estudio).	Analizar las instrucciones básicas de transferencia de datos, operaciones lógicas y Corrimientos y rotaciones
5	<b>5. USO DE UN PROGRAMA DEBUGGER.</b> 5.1 Observación y modificación de datos. 5.2 Ejecución paso a paso. 5.3 Uso de breakpoints.	Definir los conceptos básicos del uso de un programa debugger, así como la observación y modificación de datos.
5	<b>6. CONTROL DE SECUENCIA.</b> 6.1 Instrucciones para comparar operandos.	Definir y analizar los controles de secuencia, así como también Programación estructurada.

HORAS ESTIMADAS	TEMAS Y SUBTEMAS	OBJETIVOS DE LOS TEMAS
	<p>6.2 Saltos condicionales e incondicionales.</p> <p>6.3 Definición y llamada de procedimientos.</p> <p>6.4 Instrucciones para el manejo del stack.</p> <p>6.5 Programación estructurada (auto estudio).</p>	
4	<p><b>7. PROGRAMAS EN LENGUAJE ENSAMBLADOR.</b></p> <p>7.1 Estructura general de un programa.</p> <p>7.2 Proceso de generación de un programa ejecutable.</p>	<p>Analizar los programas en lenguaje ensamblador. Así como también la estructura general de un programa.</p>
4	<p><b>8. DEFINICIÓN DE DATOS.</b></p> <p>8.1 Definición y uso de variables.</p> <p>8.2 Definición y uso de constantes.</p>	<p>Conocer las variables y constantes, como cuando se les emplea.</p>
4	<p><b>9. INTERRUPCIONES.</b></p> <p>9.1 Cómo funciona una interrupción.</p> <p>9.2 Usando interrupciones de software.</p> <p>9.3 Interrupciones para interactuar con el usuario.</p>	<p>Definir cuando funciona una interrupción, así como usando Software.</p>
4	<p><b>10. MANEJO DE STRINGS.</b></p> <p>10.1 Instrucciones.</p> <p>10.2 Prefijos de repetición.</p>	<p>Definir el uso del manejo de Strings, así como también los prefijos de repetición.</p>
4	<p><b>11. MANEJO DE VIDEO.</b></p> <p>11.1 Pantalla de texto.</p> <p>11.2 Pantalla de gráficas MCGA (autoestudio).</p> <p>11.3 Pantalla de gráficas VGA.</p>	<p>Conocer el manejo de video de una pantalla de texto, gráficas GA y VGA.</p>
5	<p><b>12. INTERFASE CON UN LENGUAJE DE ALTO NIVEL.</b></p> <p>12.1 Convenciones para el desarrollo de rutinas externas.</p> <p>12.2 Uso de parámetros.</p> <p>12.3 Compartir rutinas y datos.</p>	<p>Aprender las interfaces con un lenguaje de alto nivel, así como convenciones para el desarrollo de rutinas externas.</p>
5	<p><b>13. PROGRAMAS RESIDENTES.</b></p> <p>13.1 Características.</p> <p>13.1 Interrupciones para implementar programas residentes.</p> <p>13.2 Reglas para la creación de programas residentes.</p>	<p>Conocer los programas residentes, así como también sus características.</p>

**ASIGNATURA: PROGRAMACIÓN II****DEL PROGRAMA ACADÉMICO: INGENIERIA EN MECATRÓNICA****EXPERIENCIA DE APRENDIZAJE (METODOLOGÍA):**

El profesor deberá exponer en clase el marco teórico que sustenta cada tema, ejemplificar y desarrollar rutinas de programación en lenguaje ensamblador en las que se apliquen los conceptos expuestos.

El alumno deberá desarrollar las siguientes actividades:

- Realizar auto estudios.
- Leer artículos.
- Resolver tareas escritas.
- Realizar tareas de investigación.
- Desarrollar programas.
- Presentar exámenes rápidos (actividad sugerida para evaluar los auto estudios).
- Presentar tres exámenes parciales.
- Presentar un examen final.

**BIBLIOGRAFÍA (AUTOR, TÍTULO, EDITORIAL, EDICIÓN):**

1. Michael Thorne. Computer organization and assembly language programming.  
Benjamin Cummings

**LIBRO (S) DE CONSULTA**

1. Nabajyoti Barkakati. Microsoft macro assembler bible. Howard W. Sams & Company

2. William C. Runnion. Structured programming in assembly language for the IBM PC and PS/2. PWS Publishing Company

**RECURSOS DIDÁCTICOS:**

**Proyector de acetatos**

**Cañón**

**Computadora**

**NORMAS Y PROCEDIMIENTO DE EVALUACIÓN:**

3 EXÁMENES PARCIALES	45 %
TAREAS Y EXÁMENES RÁPIDOS	7 %
PROGRAMAS:	
INTERRUPCIONES	4 %
PANTALLA DE GRÁFICAS	5 %
INTERFASE	4 %
EXAMEN FINAL	35 %