

## 1.- DATOS DE LA ASIGNATURA

<p>Nombre de la asignatura: <b>Introducción a la Programación</b></p> <p>Carrera: <b>Ingeniería Electromecánica</b></p> <p>Clave de la asignatura: <b>EMZ - 0522</b></p> <p>Horas teoría-horas práctica-créditos <b>1- 3- 5</b></p>
---

## 2.- HISTORIA DEL PROGRAMA

<b>Lugar y fecha de elaboración o revisión</b>	<b>Participantes</b>	<b>Observaciones (cambios y justificación)</b>
Instituto Tecnológico de Ocotlán del 23 al 27 agosto 2004.	Representante de las academias de ingeniería Electromecánica de los Institutos Tecnológicos.	Reunión Nacional de Evaluación Curricular de la Carrera de Ingeniería Electromecánica
Instituto Tecnológico de Cd. Juárez	Academias de Ingeniería Electromecánica	Análisis y enriquecimiento de las propuestas de los programas diseñados en la reunión nacional de evaluación
Instituto Tecnológico de Acapulco del 14 al 18 febrero 2005	Comité de Consolidación de la carrera de Ingeniería Electromecánica.	Definición de los programas de estudio de la carrera de Ingeniería Electromecánica.

## 3.- UBICACIÓN DE LA ASIGNATURA

a). Relación con otras asignaturas del plan de estudios.

<b>Anteriores</b>		<b>Posteriores</b>	
<b>Asignaturas</b>	<b>Temas</b>	<b>Asignaturas</b>	<b>Temas</b>
		En todas las asignaturas posteriores donde se aplique la programación.	

b). Aportación de la asignatura al perfil del egresado

- Fomenta el análisis lógico de problemas específicos dividiendo el problema en módulos para facilitar su comprensión y solución.
- Proporciona las bases de la programación al egresado, para que este realice aplicaciones dirigidas a resolver problemas de ingeniería.

#### 4.- OBJETIVO(S) GENERAL(ES) DEL CURSO

Proporcionará las herramientas de programación con un lenguaje estructurado para la solución de problemas de ingeniería.

#### 5.- TEMARIO

Unidad	Temas	Subtemas
1	Introducción a la Programación	1.1 Historia de los lenguajes de programación. 1.2 Análisis y comprensión de problemas. 1.3 Diseño de algoritmos. 1.3.1 Conceptos básicos. 1.3.2 Metodología de la programación 1.3.3 Diagramas de flujo. 1.3.4 Pseudocódigos. 1.4 Variables, operadores, tipos de datos y expresiones. 1.4.1 Variables y constantes. 1.4.2 Tipos de datos fundamentales. 1.4.3 Modificadores de datos. 1.4.4 Operadores y precedencia. 1.4.5 Construcción de expresiones. 1.4.6 Sentencias y bloques de sentencias de programa. 1.5 Estructura de un programa.
2	Entradas y Salidas	2.1 Salida con formato. 2.1.1. Especificadores de formato 2.1.2. Secuencias de escape. 2.1.3. Ancho de campo 2.1.4. Especificadores de precisiones. 2.2 Entrada con formato. 2.3 Otras funciones de entrada y salida.
3	Estructuras de control	3.1 Control secuencial.

	selectivas y repetitivas.	<p>3.2 Control selectivo</p> <p>3.2.1 Las sentencias <b>if</b> e <b>if-else</b>.</p> <p>3.2.2 Selección Múltiple ( <b>if</b> anidados y sentencia <b>switch</b>).</p> <p>3.3. Control repetitivo.</p> <p>3.3.1 Sentencia <b>while</b>.</p> <p>3.3.2 Sentencia <b>do-while</b>.</p> <p>3.3.3 Sentencia <b>for</b>.</p> <p>3.3.4 Ciclos anidados.</p> <p>3.3.5 Ciclos infinitos.</p>
4	Funciones	<p>4.1 Declaración, definición y el prototipo de una función creada por el programador.</p> <p>4.2 Cuerpo de la función.</p> <p>4.3 Llamadas a una función y el paso de argumentos.</p>
5	Arreglos	<p>5.1 Declaración de arreglos y procesamiento de un arreglo</p> <p>5.2 Arreglos de caracteres.</p> <p>5.3 Paso de arreglos a funciones.</p> <p>5.4 Búsqueda en arreglos y ordenamiento de arreglos.</p>
6	Apuntadores	<p>6.1 Declaración e inicialización de variables de apuntadores.</p> <p>6.2 Aritmética de apuntadores.</p> <p>6.3 Arreglo de apuntadores.</p> <p>6.4 Apuntadores a funciones.</p>
7	Estructuras	<p>7.1 Definición, declaración e inicialización de estructuras.</p> <p>7.2 Estructuras y apuntadores.</p> <p>7.3 Arreglo de estructuras.</p> <p>7.4 Paso de estructuras como argumentos a funciones.</p>
8	Archivos	<p>8.1 Archivos de acceso secuencial.</p> <p>8.2 Archivos de acceso directo o aleatorio.</p>

## 6.- APRENDIZAJES REQUERIDOS

- Uso básico de la computadora.

## 7.- SUGERENCIAS DIDÁCTICAS

- Realizar lecturas de diferentes autores para comparar las diferentes reglas de sintaxis.
- Organizar sesiones grupales de discusión de conceptos.
- Fomentar talleres de solución de casos aplicados a la carrera tanto en clase como en laboratorio.
- Investigación documental
- Realización de reportes
- Exposiciones auxiliadas por software de presentación.

## 8.- SUGERENCIAS DE EVALUACIÓN

- Exámenes escritos.
- Prácticas de laboratorio.
- Solución de casos prácticos donde el alumno diseñe y plantee problemas.
- Participación individual y en grupo.
- Entrega de reportes o trabajos de investigación
- Exposiciones auxiliadas por software de presentación

## 9.- UNIDADES DE APRENDIZAJE

### Unidad 1.- Introducción a la programación

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
Conocerá la evolución de los lenguajes de programación e identificará los pasos principales en la solución de problemas y diseño algoritmos.	<ul style="list-style-type: none"><li>• Investigar la evolución de los lenguajes de programación. Exponer los resultados en forma grupal.</li><li>• Analizar el problema para identificar los elementos que se tienen para realizar el proceso para la solución del mismo, usando ejemplos sencillos que faciliten la comprensión.</li><li>• Identificar los conceptos básicos del diseño de algoritmos tales como; la representación de los datos, bucles, arreglos y estructuras, así como la utilidad de un buen diseño de algoritmo. Aplicar una metodología de programación (utilizando pseudocódigo o diagramas de flujo) para la solución de ejercicios propuestos.</li><li>• Investigará los diferentes tipos de variables, operadores así como los tipos</li></ul>	1,2,3, 4,5,6, 7, 8 y 9

	<p>de datos y la construcción de expresiones. Se discutirá la investigación en forma grupal. Se realizarán ejercicios en computadora.</p> <ul style="list-style-type: none"> <li>• Conocer la estructura básica y los elementos que componen un programa.</li> </ul>	
--	--	--

## Unidad 2.- Entradas y Salidas

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
<p>Conocerá las funciones que permiten la transferencia de información entre la computadora y los dispositivos de entrada/salida estándar (teclado, monitor).</p>	<ul style="list-style-type: none"> <li>• Investigar las funciones de salida con formato así como sus especificadores de formato para las diferentes variables y las diferentes secuencias de escape. También se deberá analizar el ancho de campo y los especificadores de precisiones en las salidas. Se expondrá en forma grupal. Se realizaran ejemplos de aplicación en computadora.</li> <li>• Investigar las funciones de entrada con formato así como sus especificadores de formato para las diferentes variables. Se expondrá en forma grupal. Se realizaran ejemplos de aplicación en computadora.</li> <li>• Investigar otras funciones de entrada y salida. Se expondrán en forma grupal. Se realizaran ejemplos de aplicación en computadora.</li> </ul>	<p>1,2,3, 4,5,6, 7, 8 y 9</p>

## Unidad 3.- Estructuras de control selectivas y repetitivas.

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
<p>Comprenderá y aplicará las estructuras de control selectivas y repetitivas en el desarrollo de problemas.</p>	<ul style="list-style-type: none"> <li>• Investigar y realizar una síntesis del ciclo secuencial. Usar ejemplos para demostrarlo. Se realizaran prácticas en la computadora de problemas de aplicación.</li> <li>• Conocer las diferentes estructuras de control selectivas. Usar ejemplos para comprender su funcionamiento. Se</li> </ul>	<p>1,2,3, 4,5,6, 7, 8 y 9</p>

	<p>realizaran prácticas en la computadora de problemas de aplicación.</p> <ul style="list-style-type: none"> <li>• Conocer las diferentes estructuras de control repetitivas. Usar ejemplos para comprender su funcionamiento. Realizar prácticas en la computadora de problemas de aplicación.</li> </ul>	
--	--	--

#### Unidad 4.- Funciones

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
<p>Conocerá la sintaxis y su relación dentro del programa principal de las funciones creadas por el programador.</p>	<ul style="list-style-type: none"> <li>• Investigar y realizar una síntesis de la declaración, definición y el prototipo de una función. Se hará exposición en clases. Usar ejemplos para demostrar.</li> <li>• Investigar y realizar una síntesis del cuerpo de la función. Se expondrá en clases. Usar ejemplos para demostrar</li> <li>• Investigar y realizar una síntesis de la llamada a una función y el paso de argumentos. Se hará exposición en clases. Usar ejemplos para demostrar. Realizar ejercicios de aplicación en la computadora. Resolver un problema de aplicación extraclase.</li> </ul>	<p>1,2,3, 4,5,6, 7, 8 y 9</p>

#### Unidad 5.- Arreglos

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
<p>Conocerá la sintaxis y operación de los arreglos unidimensionales y multidimensionales para aplicar estos conocimientos en el desarrollo y diseño de aplicaciones que requieran el uso de ellos.</p>	<ul style="list-style-type: none"> <li>• Investigar y realizar en el aula, mediante una lluvia de ideas la definición, declaración e inicialización de arreglos unidimensionales y multidimensionales. Realizar exposición de la investigación haciendo uso de ejemplos para demostrarlo.</li> <li>• Investigar y realizar una síntesis del uso de arreglos para guardar datos de cadenas como arreglos de caracteres, así como su inicialización, usando la</li> </ul>	<p>1,2,3, 4,5,6, 7, 8 y 9</p>

	<p>función <b>malloc</b> para asignar espacio a la cadena, también las funciones de entradas y salidas de cadenas. Realizar exposición de la investigación haciendo uso de ejemplos para demostrarlo.</p> <ul style="list-style-type: none"> <li>• Investigar y realizar ejercicios sobre paso de arreglos a funciones.</li> <li>• Investigar y realizar ejercicios sobre búsqueda en arreglos y ordenamiento de arreglos</li> </ul>	
--	--	--

### Unidad 6.- Apuntadores

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
Conocerá la sintaxis y operación de los apuntadores como un método poderoso y flexible para el manejo de datos en la programación.	<ul style="list-style-type: none"> <li>• Investigar y realizar una síntesis de la definición, declaración e inicialización de variables de apuntadores. Realizar exposición de la investigación haciendo uso de ejemplos para demostrarlo.</li> <li>• Investigar y realizar ejercicios sobre aritmética de apuntadores</li> <li>• Investigar y realizar ejercicios sobre arreglo de apuntadores.</li> <li>• Investigar y realizar ejercicios sobre apuntadores a funciones.</li> <li>• Se realizaran ejercicios en la computadora sobre problemas de aplicación para poner en práctica toda esta teoría.</li> </ul>	1,2,3, 4,5,6, 7, 8 y 9

### Unidad 7.- Estructuras

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
Conocerá la sintaxis de las estructuras y su aplicación en la solución de problemas.	<ul style="list-style-type: none"> <li>• Investigar y realizar una síntesis de la definición, declaración e inicialización de estructuras. Realizar exposición de la investigación haciendo uso de ejemplos para demostrarlo.</li> <li>• Investigar y realizar ejercicios sobre estructuras y apuntadores.</li> <li>• Investigar y realizar ejercicios sobre arreglo de estructuras.</li> <li>• Investigar y realizar ejercicios sobre el paso de estructuras como argumentos a</li> </ul>	1,2,3, 4,5,6, 7, 8 y 9

	<p>funciones.</p> <ul style="list-style-type: none"> <li>• Se realizaran ejercicios en la computadora sobre problemas de aplicación para poner en práctica toda esta teoría.</li> </ul>	
--	---	--

## Unidad 8.- Archivos

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
<p>Conocerá la relación de los flujos con archivos de disco y los tipos de archivo así como sus funciones para abrirlos, escribirlos, leerlos y cerrarlos.</p>	<ul style="list-style-type: none"> <li>• Investigar y realizar una síntesis de los flujos y tipos de archivos así como de las diferentes funciones para escribir y leer datos de archivos de acceso secuencial. Realizar exposición de la investigación haciendo uso de ejemplos para demostrarlo.</li> <li>• Investigar y realizar una síntesis de los flujos y tipos de archivos así como de las diferentes funciones para escribir y leer datos de archivos de acceso aleatorios o directos. Realizar exposición de la investigación haciendo uso de ejemplos para demostrarlo.</li> <li>• Se realizaran ejercicios en la computadora sobre problemas de aplicación para poner en práctica toda esta teoría.</li> </ul>	<p>1,2,3, 4,5,6, 7, 8 y 9</p>

## 10. FUENTES DE INFORMACIÓN

1. Joyanes Aguilar, Luis. Programación en C Algoritmos, estructuras de datos y objetos. Editorial Mc Graw Hill, 2001.
2. Cevallos, Javier. C/C++ Curso de Programación. Editorial Alfaomega Ra-Ma, 2002. 2° Edición.
3. Schildt, Hebert. C Manual de Referencia. Editorial Mc Graw Hill, 2002. 4ª Edición
4. Hernán Ruiz, Marcelo. Programación C. EditorialMP Ediciones S. A., 2003.
5. Perry, Greg. C con Ejemplos. Editorial Prentice Hall. 2000.
6. Zhang, Tony. Aprendiendo C en 24 Horas. Editorial Prentice Hall, 2002.
7. S. Gottfried, Byron. Programación en C. Editorial Mc Graw Hill, 1997. 2ª Edición.
8. H. M. Deitel / P. J. Deitel. Como Programar en "C / C++". Editorial Prentice Hall Hispanoamericana, 2000. 4ª edición.

9. P. Aitken / B. Jones. Aprendiendo "C" en 21 días. Editorial Prentice Hall Hispanoamericana, 1991.
10. A. Kelley / I. Pohl. C by Dissection- The Essentials of C Programming. The Benjamin / Cummings Publishing Company Inc., 1992. 2ª edition
11. Hebert Schildt. Turbo C / C++ - Manual de referencia. Editorial Mc Graw Hill, 1992.
12. B. W. Kernighan / D. M. Ritchie. El Lenguaje de Programación. Editorial Prentice Hall Hispanoamericana, 1991. 2ª edición.
13. Delores M. Etter. Engineering Problem Solving with C. Prentice Hall, 2005.