

## 1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura: <b>Programación en Tiempo Real</b>
Carrera: <b>Ingeniería Mecatrónica</b>
Clave de la asignatura: <b>MTF-0535</b>
Horas teoría-horas práctica-créditos: <b>2-4-8</b>

## 2.- HISTORIA DEL PROGRAMA

Lugar y fecha de elaboración o revisión	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Reynosa, del 6 al 10 de diciembre del 2004.	Representante de las academias de ingeniería Mecatrónica de los Institutos Tecnológicos.	Reunión nacional de evaluación curricular de la carrera de Ingeniería Mecatrónica
Institutos tecnológicos de Querétaro y Toluca e Instituto Tecnológico Superior de Ecatepec, de enero a marzo del 2005	Academias de Ingeniería Mecatrónica y Sistemas Computacionales	Análisis y enriquecimiento de las propuestas de los programas diseñados en la reunión nacional de evaluación
Instituto Tecnológico de del 16 al 20 de Mayo del 2005	Comité de consolidación de la carrera de Ingeniería Mecatrónica	Definición de los programas de estudio de la carrera de Ingeniería Mecatrónica

## 3.- UBICACIÓN DE LA ASIGNATURA

### a). Relación con otras asignaturas del plan de estudio

Anteriores		Posteriores	
Asignaturas	Temas	Asignaturas	Temas
Lógica de programación	- Proporciona las bases para el desarrollo del curso	Tópicos de Programación	- Proporciona las bases para el desarrollo del curso

### b). Aportación de la asignatura al perfil del egresado

Proporcionar las bases teóricas para analizar, desarrollar y programar modelos matemáticos, estadísticos y de simulación utilizados en el desarrollo de programas computacionales.

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

Comprenderá los conceptos básicos de lógica matemática, relaciones, árboles y grafos para aplicarlos a modelos que resuelvan problemas de Mecatrónica.

#### 5.- TEMARIO

Unidad	Temas	Subtemas
1	Grafos	1.1 Introducción a grafos 1.2 Caminos y Ciclos. 1.3 Ciclos Hamiltonianos 1.4 Algoritmos de la ruta más corta. 1.5 Representaciones gráficas 1.6 Isomorfismo. 1.7 Grafos planares. 1.8 Algoritmos para uso grafos 1.9 Ejercicio de la locura instantánea.
2	Árboles	2.1 Definición de un árbol. 2.2 Terminología y caracterización de los árboles. 2.3 Árboles de expansión mínimos. 2.4 Árboles Binarios. 2.5 Recorridos en un árbol 2.6 Árboles de decisión y el tiempo mínimo para el ordenamiento. 2.7 Isomorfismo de árboles.
3	Codificación	3.1 Grupos 3.1.1 Homeomorfismos 3.1.2 Isomorfismos. 3.1.3 Cíclicos. 3.1.4 Cosets 3.1.5 Teorema de Lagrange 3.1.6 Métrica de Hamming 3.1.7 Matrices Generadoras y de paridad. 3.2 Anillos. 3.2.1 Grupos de códigos. 3.2.2 Cosets líderes. 3.2.3 Matrices de Hamming. 3.2.4 Campos finitos. 3.2.5 Anillos de polinomios. 3.2.6 Polinomios irreducibles 3.2.7 Cuadrados latinos 3.2.8 Criptografía.

## 5.- TEMARIO

Unidad	Temas	Subtemas
4	Lenguajes y Autómatas	4.1 Introducción a Lenguajes y Autómatas. 4.2 Circuitos secuenciales y máquinas de estado finito. 4.3 Autómatas de estado finito. 4.4 Lenguajes y gramáticas. 4.5 Autómatas de estado finito no determinista. 4.6 Relación entre lenguajes y autómatas.
5	Máquinas de Turing	5.1 Definiciones Básicas. 5.2 Máquinas de Turing 5.3 Construcción de máquinas de Turing.
6	Redes de Petri	6.1 Modelo de redes. 6.2 Algoritmo de flujo máximo 6.3 Teorema de flujo máximo y corte mínimo. 6.4 Acoplamiento 6.5 Redes de Petri

## 6.- APRENDIZAJES REQUERIDOS

- Conocimiento de álgebra general.
- Lógica de programación.
- Matemáticas discretas.

## 7.- SUGERENCIAS DIDÁCTICAS

- Desarrollar en cada una de las unidades de aprendizaje algoritmos computacionales.
- Propiciar la búsqueda y selección de información sobre: Grafos Redes de Petri, máquinas de Turing.
- Seleccionar temas de la representación de grafos para discusión en clase.
- Realizar ejercicios relacionados con árboles generadores en aplicaciones computacionales
- Relacionar los sistemas numéricos con aplicaciones en los circuitos lógicos.
- Utilizar herramientas de simulación de circuitos.

## 8.- SUGERENCIAS DE EVALUACIÓN

- Dar seguimiento al desempeño en el desarrollo del programa (dominio de los conceptos, capacidad de la aplicación de los conocimientos en problemas reales, transferencia del conocimiento).
- Participación en actividades individuales y de equipo.
- Cumplimiento de los objetivos y desempeño en los ejercicios.
- Observar el dominio en el uso de técnicas de modelado de grafos.

## 9.- UNIDADES DE APRENDIZAJE

### Unidad 1: Grafos

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
El estudiante reafirmará las bases matemáticas necesarias para la creación de grafos.	<ul style="list-style-type: none"><li>• Diseñar los diferentes grafos.</li><li>• Programar el recorrido de un grafo.</li></ul>	1 2 3 5

### Unidad 2: Árboles

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
Representará árboles.	<ul style="list-style-type: none"><li>• Diseñar un árbol binario</li><li>• Realizar el recorrido de un árbol utilizando un lenguaje de programación.</li></ul>	1 2 3 5

### Unidad 3: Codificación

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
Representara grupos	<ul style="list-style-type: none"><li>• Establecer la diferencia entre los grupos</li><li>• Investigar la aplicación de anillos</li><li>• Investigar métodos de criptografía</li></ul>	2 3 4

#### Unidad 4: Lenguajes y Autómatas

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
Representará lenguajes a través de autómatas, expresiones regulares y su aplicación.	<ul style="list-style-type: none"><li>• Utilizar un lenguaje de programación de alto nivel para representar expresiones regulares.</li><li>• Realizar prácticas de laboratorio para la programación de PLC's, como casos de aplicación de autómatas.</li><li>• Investigar otras aplicaciones sobre la teoría de lenguajes regulares.</li></ul>	6 7 8 9 10

#### Unidad 5: Máquinas de Turing

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
Comprenderá la representación de lenguajes y funciones en una máquina de Turing.	Realizar ejercicios que permitan la representación de operaciones matemáticas básicas como suma, resta, multiplicación, potencia, entre otros. Utilizar la teoría para la representación de lenguajes. Simular a través de un lenguaje de alto nivel, la representación de una máquina de Turing.	6 7 8 9 10

#### Unidad 6: Redes de Petri

<b>Objetivo Educativo</b>	<b>Actividades de Aprendizaje</b>	<b>Fuentes de Información</b>
Comprenderá la representación y función de una red de Petri.	<ul style="list-style-type: none"><li>• Realizar ejercicios para el diseño de una red de Petri.</li><li>• Hacer uso de un simulador para generar una red de Petri.</li></ul>	6 7 8 9 10

## 10.- FUENTES DE INFORMACIÓN

1. Johnsonbaugh, Richard, *Discrete Mathematics*, Ed. Prentice Hall; Última Edición
2. Patrick Suples, *Lógica Matemática*, Ed. CECSA
3. Liu, C.L., *Elements Of Discrete Mathematics*
4. Hasser Lassalle & Sullivan, *Análisis Matemático*, Ed. Trillas Vol. I
5. Johnson & Baugh, *Matemáticas Discretas*, Ed. Interamericana
6. Hopcroft, John, Ullman, Jeffrey, *Introduction to Automatas Theory, Languages and Computation*, Ed. Addison-Wesley.
7. Kelley, Dean, *Teoría de Autómatas y Lenguajes Formales*, Ed. Prentice Hall.
8. Rayward-Smith, V.S, *A First Course in a Formal Language Theory*, Ed. Mc Graw Hill
9. Brookshear, *Teoría de la Computación, Lenguajes Formales, Autómatas y Complejidad*, Ed. Addison Wesley.
10. Isasi, Martínez y Borrajo, *Lenguajes, Gramáticas y Autómatas*, Ed: Addison Wesley

## 11. PRÁCTICAS PROPUESTAS

- Grafos.
  - Demostración de grafos que contenga o involucren los circuitos de Hamilton.
  - Representación de grafos utilizando diferentes tipos de matriz (adyacencia, incidencia)
  - Desarrollar el algoritmo del camino más corto.
- Árboles.
  - Desarrollar un algoritmo que pueda simular un código utilizando los principios de Huffman.
- Codificación
  - Demostración del teorema de Lagrange.
- Lenguajes y Autómatas
  - Realizar prácticas en laboratorio para la programación de PLC's, como casos de aplicación de autómatas o en su defecto el uso de simuladores de software.
- Máquinas de Turing
  - Simular a través de un lenguaje de alto nivel, la representación de una máquina de Turing
- Redes de Petri
  - Realizar una práctica en laboratorio para el diseño de una red de Petri.