



UNIVERSIDAD FERMIN TORO
VICE RECTORADO ACADEMICO
FACULTAD DE INGENIERIA
ESCUELA DE COMPUTACION

PROGRAMA INSTRUCCIONAL

AUTOMATA DE LENGUAJES FORMALES

CODIGO		SEMESTRE	U.C.	DENSIDAD HORARIA				PRE - REQUISITO
ASIGNADO	EQUIVALENTE			H.T.	H.P/HL	H.A.	THS/SEM	
ALF-733		VII	3	2	2		4/64	LOC-643

Especialista en contenido:	ING. EDECIO FREITEZ	AUTORIZADO POR: VICE-RECTORADO ACADÉMICO (FIRMA Y SELLO)
Fecha de elaboración:	JUNIO, 2001	
Elaborado por	ING. EDECIO FREITEZ	

FUNDAMENTACION

Las bases teóricas en las que se fundamenta la computación son las mismas que han permitido el desarrollo de las matemáticas, son de gran importancia para comprender aspectos teóricos y prácticos de uso muy amplio. La teoría de autómatas y lenguajes formales como la complejidad son fundamentales para todo estudiante de las ciencias de la computación tanto por su carácter formativo como por su importancia en la solución de problemas prácticos.

El desarrollo del programa ha sido elaborado con el propósito de transmitir al estudiante de las ciencias de la computación los conceptos fundamentales de la teoría de autómatas y lenguajes formales, presentándole una visión global y profundizando principalmente en aspectos relacionados al diseño de autómatas, máquinas de Turing, la obtención de analizadores léxicos y sintácticos a partir de expresiones regulares y gramáticas independientes del contexto, así como la jerarquización de problemas computables de acuerdo a su complejidad.

La estructura del programa consta de cinco unidades:

- I Unidad: introducción y conceptos básicos.
- II Unidad: autómatas de estados finitos.
- III Unidad: lenguajes formales.
- IV Unidad: máquinas de Turing.
- V Unidad: complejidad.

La asignatura se organizará en torno a clases teóricas, prácticas. Asignación de casos prácticos. En las clases teóricas se expondrán los aspectos concernientes a las unidades que conforman este programa. En las clases de problemas o prácticas se realizarán ejercicios relacionados con los modelos, técnicas y algoritmos presentados en las clases teóricas. Las asignaciones prácticas, son trabajos cortos o proyectos de fin de semestre donde el estudiante pondrá de manifiesto los conocimientos adquiridos en algunas de las etapas teórico – práctica.

OBJETIVO GENERAL DE LA ASIGNATURA

El estudiante conocerá las bases teóricas de la computación y su relación con los aspectos prácticos de la teoría de lenguajes, autómatas y la complejidad de problemas de diversa índole.

UNIDAD I		OBJETIVO TERMINAL			
INTRODUCCION Y CONCEPTOS BASICOS		APLICAR LOS CONCEPTOS BASICOS DE LA TEORIA DE AUTOMATAS Y EXPRESIONES REGULARES COMO MEDIO QUE PERMITA LA CONSTRUCCIÓN DE LENGUAJES REGULARES.			
DURACION					
2 SEMANAS					
EVALUACION					
5 %					
OBJETIVOS ESPECIFICOS		CONTENIDO		ESTRATEGIAS DE INSTRUCCION	
1. Definir claramente los conceptos de símbolo, palabra, alfabeto, universo sobre un alfabeto. Palabra vacía. Longitud de una palabra. Explicación de la relación existente entre ellos.		• Introducción. • Símbolo. • Palabra. • Alfabeto. • Universo sobre alfabeto. • Longitud de una palabra. • Operación con palabras. • Lenguaje. Tipos de lenguaje. • Operaciones con lenguaje. • Lenguaje regular (LR). • Expresión regular. • Teorema de Kleenne.		• Ciencias exactas.	
2. Describir las operaciones que se pueden realizar con las palabras de un lenguaje ilustrando con ejemplos su uso y aplicación.					
3. Explicar los lenguajes regulares resaltando las propiedades y teoremas que los sustentan.					
ESTRATEGIAS DE EVALUACIÓN:					
• Formativa. • Prueba diagnostica.					

UNIDAD II		OBJETIVO TERMINAL	
AUTOMATAS DE ESTADOS FINITOS		ANALIZAR LAS MAQUINAS TEORICAS QUE PERMITEN EL RECONOCIMIENTO DE LENGUAJES REGULARES PARA APLICARLAS EN SOLUCIÓN DE PROBLEMAS COMPUTABLES.	
DURACION			
4 SEMANAS			
EVALUACION			
20 %			
OBJETIVOS ESPECIFICOS	CONTENIDO	ESTRATEGIAS DE INSTRUCCION	
1. Describir los autómatas finitos señalando los componentes que la conforman utilizándola para la representación y reconocimiento de patrones. 2. Utilizar los autómatas de estados finitos como reconocedores de elementos léxicos. 3. Aplicar los autómatas de estados finitos en la solución de problemas computables.	• Maquinas de estados finitos (MEF). • Maquinas secuenciales (MS). • Autómatas finitos determinísticos (AFD). • Autómatas finitos no determinísticos con E transiciones. • Transformación de AFND con E. Transiciones. • Conceptos asociados a (AFD) y (AFND). • Minimización de (AFD). • Automátas de células de Mc Culloch – Pitts. • Autómatas probabilísticas. • Análisis Lexico.	• Ciencias exactas.	
ESTRATEGIAS DE EVALUACIÓN:			
• Formativa. • Taller. • Prueba larga. • Método de proyecto.			

UNIDAD III		OBJETIVO TERMINAL	
LENGUAJES FORMALES		ESTUDIAR LAS MAQUINAS TEORICAS QUE RECONOCEN LENGUAJES LIBRES DEL CONTEXTO Y LOS PASOS BASICOS DEL PROCESO DE ANALISIS SINTACTICO PARA LA GENERACIÓN DE CADENAS A PARTIR DE UNA GIC.	
DURACION			
4 SEMANAS			
EVALUACION			
25 %			
OBJETIVOS ESPECIFICOS	CONTENIDO	ESTRATEGIAS DE INSTRUCCION	
1. Describir las gramáticas regulares destacando sus componentes aplicándola en la obtención de patrones. 2. Examinar las propiedades de los lenguajes independientes del contexto resaltando su uso por medio de la elaboración de ejercicios prácticos. 3. Construir arboles de derivación como una herramienta que permite la especificación de patrones para una GIC. 4. Utilizar los autómatas de pila para el reconocimiento de lenguajes independientes del contexto.	• Gramáticas regulares: definición, representación. • Gramáticas regulares y lenguajes regulares. • Gramáticas independientes del contexto (GIC): Definición, representación. • Arboles de derivación o de análisis y ambigüedad. • Forma normal de Chomsky (FNC) • Forma normal de Greibach (FNG) • Autómatas de pila: definición. Movimiento. • Descripción instantánea (DIS) • Autómatas de pila y GIC.	• Ciencias exactas.	
ESTRATEGIAS DE EVALUACIÓN:			
• Formativa. • Prueba larga. • Taller.			

UNIDAD IV		OBJETIVO TERMINAL	
MAQUINAS DE TURNING		ESTUDIAR EL PODER COMPUTACIONAL DE LAS MAQUINAS DE TURNING COMO MECANISMO EFICIENTE DE RECONOCIMIENTO DE LENGUAJES PARA LA SOLUCION DE PROBLEMAS DE COMPUTACION.	
DURACION			
3 SEMANAS			
EVALUACION			
25 %			
OBJETIVOS ESPECIFICOS	CONTENIDO	ESTRATEGIAS DE INSTRUCCION	
1. Definir las maquinas de Turning identificando sus componentes, propiedades y movimientos que rigen su funcionamiento. 2. Establecer la capacidad de aceptación de lenguajes de una maquina de turning mediante ejemplos que representan la situación que se quiere conocer. 3. Utilizar técnicas que permitan las construcción de una nueva M de T de otras ya existentes. 4. Mostrar las M de T. Universales que permitan la representación de cualquier otra maquina distinguiendo entre lenguajes aceptables y lenguajes decidibles.	• Introducción. • Definición. • Representación. • Características. • Movimientos • Lenguajes aceptados por una M de Turning. • Construcción y modificación de una m de Turning. • Tesis de Church. • Indencibilidad. • Lenguajes recursivo y recursivamente enumerables. • Maquinas de Turning universales. • Problemas de parada. • El teorema de Rice.	• Ciencias exactas.	
ESTRATEGIAS DE EVALUACIÓN:			
• Formativa. • Prueba larga. • Taller.			

UNIDAD V		OBJETIVO TERMINAL	
COMPLEJIDAD		ESTUDIAR LOS PROCESOS COMPUTACIONALES PARA ESTABLECER UNA ESCALA DE CLASIFICACION DE LOS PROBLEMAS DE ACUERDO A SU COMPLEJIDAD.	
DURACION			
3 SEMANAS			
EVALUACION			
25 %			
OBJETIVOS ESPECIFICOS	CONTENIDO	ESTRATEGIAS DE INSTRUCCION	
1. Definir la complejidad distinguiendo entre los diversos patrones que la generan. 2. Aplicar la complejidad a los problemas diferenciándola entre los diversos tipos que la componen. 3. Establecer una escala para clasificar los problemas de acuerdo a su complejidad para establecer el nivel de rendimiento de los mismos.	• Complejidad de los cálculos. • Complejidad de los algoritmos. • Complejidad de los problemas. • Complejidad temporal del reconocimiento de lenguajes. • Complejidad temporal de maquinas no deterministas.	• Ciencias exactas.	
ESTRATEGIAS DE EVALUACIÓN:			
• Formativa. • Prueba larga. • Taller. • Revisión de proyectos.			

BIBLIOGRAFIA

- Bernard K. y Robert C. Busby. **Estructura de Matemáticas Discretas para la Computación**. Última Edición. Prentice Hall. May 1986.
- D. Keley. **Teoría de Autómatas y Lenguajes Formales**. 1ra Edición. Editorial Prentice Hall. 1995.
- J. E. Hopcroft y J. F. Ullman. **Introducción de Autómata. Lenguajes and Computación**. 1ra Edición. Editorial Addison Wealey. 1979.
- J. G. Bookshear. **Teoría de la Computación. Lenguajes Formales, Autómatas y Complejidad**. 1ra Edición. Editorial Addison Wealey iberoamericana. 1993.
- Niclaus Wirth. **Algoritmos + Estructura de Datos = Programas**. 1ra Edición. Ediciones del Castillo. 1980.
- P.I. Sasi, P. Martínez y D. Borrado. **Lenguajes, Gramática y Autómatas (Un Enfoque Practico)**. 1ra Edición. Editorial Addison Wealey Madrid. 1997.