

# FACULTAD DE INGENIERÍA

## POSGRADO EN INGENIERÍA Y CIENCIAS DE LA COMPUTACIÓN



Nombre de la materia :

AUTÓMATAS Y LENGUAJES FORMALES

Horas/Clase/Semana: 3

Horas totales/Semestre: 48

Créditos: 6

### OBJETIVO DEL CURSO

Estudiar de manera formal todos los conceptos de Autómatas Finitos, Gramáticas de tipo 0 hasta el tipo 3 y Lenguajes Formales. Al finalizar el curso el estudiante deberá ser capaz de aplicar dicha teoría en áreas específicas de la computación, principalmente en Lenguajes de Programación, Compiladores, Lenguaje Natural, Diseño Lógico y Análisis de Algoritmos. Las ideas que se presentarán tienen su origen en muy

diversos campos de estudio: en Biología se crearon modelos de redes neuronales; en Ingeniería Eléctrica los encontramos en el desarrollo de la teoría para la construcción de circuitos combinatorios y secuenciales; en Lingüística en la investigación de gramáticas de Lenguajes Naturales. Con el paso del tiempo todas estas investigaciones se convirtieron en parte central de las bases de la Ciencia de la Computación.

### CONTENIDO TEMÁTICO

#### 1.-Fundamentos Matemáticos

- 1.1 Conjuntos y Relaciones.
- 1.2 Relaciones de Equivalencia y Particiones.
- 1.3 Funciones: Inyectiva, Sobreyectiva y Biyectiva
- 1.4 Conjuntos Contables, No-contables y Enumerables.
- 1.5 Isomorfismos y Homomorfismos.
- 1.6 Definiciones recursivas.
- 1.7 Lenguaje Z.
- 1.8 Inducción Matemática.

#### 2.-Autómatas y Gramáticas

- 2.1 Cadenas, Alfabetos y Lenguajes.
- 2.2 Autómatas Finitos y Expresiones Regulares.
- 2.3 Árboles de Derivación.
- 2.4 Máquinas de Turing Sencillas y Paralelas.
- 2.2 Gramáticas
- 2.2 AFN, AFD, Pushdown
- 2.3 Minimización de Estados.
- 2.4 L-systems.

#### 3.-Autómatas Celulares

- 3.1 Historia.
- 3.2 Definición Formal.
- 3.3 Simulación.
- 3.4 Aplicaciones.

#### 4.-Autómatas de Redes de Petri

- 4.1 Antecedentes.
- 4.2 Definición Formal.
- 4.3 Tipos de Redes de Petri.
- 4.4 Aplicaciones

#### 4.5 StateCharts

- 4.5.1 Definición
- 4.5.2 Profundidad
- 4.5.1 Conversión de StateCharts a Redes de Petri
- 4.6 Ortogonalidad
- 4.6 Broadcasting
- 4.6 StateCharts en UML

#### 5.-Autómatas Probabilísticos o Estocásticos

- 5.1 Definición Formal.
- 5.2 Matrices de transición.
- 5.3 Vectores de Estado y Lenguaje Reconocido.
- 5.4 Aplicaciones.

#### 6.-Autómatas de Células de McCulloch-Pitts

- 6.1 Descripción.
- 6.2 Construcción de un Autómata Finito equivalente a un Autómata de Células de McCulloch-Pitts.
- 6.2 Aplicaciones.

#### 7.-Autómatas Planificadores

- 7.1 Definición Formal.
- 7.2 Creación
- 7.3 Aplicaciones

#### 8.-Compiladores

- 8.1 Definición
- 8.2 Estructura de un Compilador
- 8.3 Análisis Léxico
- 8.4 Análisis Sintáctico
- 8.5 Análisis Semántico
- 8.6 Generación de Código

8.7 Optimización

8.8 Tratamiento de Errores

## METODOLOGÍA

Durante el curso se destinará una sesión para la revisión de la teoría, previa lectura del tema por parte del alumno y en la siguiente sesión se resolverán ejercicios selectos en clase. Para cada uno de los temas el alumno deberá entregar una serie de ejercicios asignados y un programa

hecho en un lenguaje específico, el cual será definido para cada tema. Durante todo el semestre los alumnos harán investigación sobre un tema específico y al final del curso deberán entregar un reporte en forma de artículo del mismo.

## EVALUACIÓN

Cinco programas 25%

Reporte final de la investigación 25%

Dos exámenes 25%

Tareas y Exposiciones en clase 25%

## BIBLIOGRAFÍA

*Teoría de la Computación. Lenguajes Formales, Autómatas y Complejidad.*

Brookshear Glenn J.,

Addison Wesley Iberoamericana 1993.

*Teoría de Autómatas y Lenguajes Formales*

Kelley, Dean

Prentice Hall, 1995.

*Autómatas y Lenguajes Formales.*

Espitia L., Ibarquengoitia G., Comunicaciones Técnicas Serie Azul # 102. IIMAS.-UNAM, México, 1987.

*Algebra Abstracta*

I.N. Herstein,,

Grupo Editorial Iberoamérica, 1986, Mexico.

*Compiladores, Principios, Técnicas y Herramientas*

Aho, Sethi, Ullman

Addison Wesley, 1990

*Introduction to Automata Theory, Languages, and Computation.*

Hopcroft John E., Ullman Jeffrey., Addison Wesley, 1979.

*The theory and practice of compiler writing*

Tremblay, Sorenson

Mc Graw Hill International Edition

*Compilation of EPA's Sampling and Analysis Methods*

Keith Lawrence H.

Ed. Lewis

*Crafting a compiler with C*

Fisher Charles N., LeBlanc Richard Jr.

Addison Wesley Longman, 2001

*Apuntes de la Materia de Compiladores*

Botello Aragón Miguel Angel

Area de Computación e Informática, Fac. Ing. UASLP.

*Introduction to Computer Theory.*

Cohen Daniel I.A.,

Hohn Willey & Sons., Inc. 1991.

*A Modelling and Analysis Method for Manufacturing*

*Systems Based on Petri-Nets,*

Zimmerman, Kühnel, Hommel,

Proc. Computational, Engineering in Systems

Applications (CESA'98), pp.: 276-281.

*Petri Net Theory and the modeling of systems,*

James L. Peterson, Prentice Hall, 1981.

*Constructing the User Interface with Statecharts,*

Ian Horrocks,

Addison-Wesley.

*The STATEMATE Semantics of Statecharts,*

Harel D., Naamad A, ACM Transactions on Software

Engineering and Methodology, Vol. 5, No. 4, Oct. 1996,

pp.: 293-233.

*Practical Statecharts in C/C++: Quantum Programming for Embedded Systems,*

CMP Books,

Miro Samek, Ph.D.