

## UNIVERSIDAD DE MENDOZA – FACULTAD DE INGENIERÍA

<b>CARRERA</b> Ingeniería en Informática	<b>ASIGNATURA</b> TEORÍA DE COMPILADORES	<b>CÓDIGO</b> 2035
<b>CURSO</b> 3ro	<b>ÁREA</b> TECNOLOGÍAS BÁSICAS	<b>ULTIMA REVISIÓN</b> Marzo de 2015
<b>MATERIAS CORRELATIVAS:</b> 2022- MATEMÁTICA DISCRETA y DISEÑO LÓGICO		<b>AÑO LECTIVO 2016</b>

Profesor Titular: Ing. Alfredo Iglesias
Profesor Asociado:
Profesores Adjuntos:
Jefes de trabajos prácticos: Ing. Nora Costa

Carga Horaria Semanal:	4
Carga Horaria Total:	60

### **OBJETIVOS:**

Al finalizar el curso, el alumno deberá haber adquirido los conceptos básicos de la teoría de lenguajes formales y autómatas. Conocer los modelos, teorías y algoritmos asociados a la construcción de compiladores, y ser capaz de aplicar estos modelos a la solución de problemas relacionados al diseño y desarrollo de software.

### **PROGRAMA ANALÍTICO:**

#### **Capítulo I: Introducción**

**Tema 1:** Antecedentes de la arquitectura computacional y los principios de los lenguajes de programación.

**Tema 2:** Procesadores de lenguajes. Estructura de un compilador. Evolución de los lenguajes de programación.

**Tema 3:** Construcción de compiladores. Aplicaciones de la teoría de compiladores. Nuevas arquitecturas de computadoras

#### **Capítulo II: Análisis Léxico**

**Tema 1:** Especificaciones de tokens: cadenas, lenguajes, operaciones en los lenguajes, expresiones regulares, definiciones regulares.

**Tema 2:** Reconocimiento de tokens: Diagramas de transición, reconocimiento de palabras reservadas. Autómatas finitos deterministas y no deterministas, tablas de transición, aceptación de cadenas.

**Tema 3:** Obtención de autómatas a partir de expresiones regulares. Conversión de AFN en AFD, Diseño de analizadores léxicos, estructura del analizador.

**Tema 4:** Máquinas de Turing. Definición formal. Diagrama de estados. Lenguajes asociados. Restricciones. Máquinas de Turing no deterministas.

### **Capítulo III: Análisis Sintáctico**

**Tema 1:** Función del analizador sintáctico, representación de gramáticas, manejo de errores. Gramáticas libres de contexto, definiciones formales, derivaciones, árboles sintácticos y derivaciones, eliminación de ambigüedad, eliminación de recursividad.

**Tema 2:** Análisis sintáctico descendente: Análisis de descenso recursivo, primero y siguiente. Análisis sintáctico predictivo no recursivo. Recuperación de errores.

**Tema 3:** Análisis sintáctico ascendente: Reducciones, poda de mangos. Analizadores LR, algoritmos, construcción de tablas. Analizadores LR canónicos y LALR. Gramáticas ambiguas, precedencia y asociatividad.

**Tema 4:** Generadores de analizadores sintácticos Yacc.

### **Capítulo IV: Traducción dirigida por la sintaxis**

**Tema 1:** Traducción dirigida por la sintaxis, definiciones dirigidas por la sintaxis. atributos heredados y sintetizados,

**Tema 2:** Evaluación de atributos. Construcción de árboles sintácticos. Estructura de tipos.

**Tema 3:** Esquemas de traducción orientados por la sintaxis. Grafo dirigido acíclico.

### **Capítulo V: Generación de código intermedio**

**Tema 1:** Código de tres direcciones. Cuádruplos, Triplos.

**Tema 2:** Tipos y declaraciones, equivalencias, distribución de almacenamiento

**Tema 3:** Comprobación de tipos, conversiones, sobrecarga de funciones y operadores.

Formación Práctica	Horas
Resolución de Problemas Rutinarios:	
Laboratorio, Trabajo de Campo:	
Resolución de Problemas Abiertos de ingeniería:	15
Proyecto y Diseño:	

### **PROGRAMA DE TRABAJOS PRÁCTICOS:**

Trabajo Práctico 1: Expresiones Regulares, Autómatas y Maquinas de Turing

Trabajo Práctico 2: Construcción de Analizadores Léxicos – Uso de Lex

Trabajo Práctico 3: Construcción de Analizadores Sintácticos – Uso de Yacc

### **ARTICULACIÓN HORIZONTAL Y VERTICAL DE CONTENIDOS:**

- Los contenidos abordados en esta materia se basan en conceptos de las siguientes cátedras:

<b><i>Asignatura</i></b>	<b><i>Curso</i></b>
Arquitectura de computadoras	2do
Sistemas Operativos	2do
Matemática Discreta y Diseño Lógico	2do

- Comparte e integra elementos horizontalmente con las siguientes cátedras:

<b><i>Asignatura</i></b>	<b><i>Curso</i></b>
Computación II	3ro

- Los contenidos abordados en esta materia aportan conceptos a las siguientes cátedras:

<b><i>Asignatura</i></b>	<b><i>Curso</i></b>
Computación II	

### **CONDICIONES PARA REGULARIZAR LA MATERIA y RÉGIMEN DE EVALUACIÓN:**

Para aprobar la materia el alumno deberá:

Aprobar el 100% de los prácticos.

Cumplir con el 80% de asistencia

Rendir un examen final.

### **BIBLIOGRAFÍA:**

Principal:

Autor	Título	Editorial	Año Ed.	Dispon.
Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jersey D. Ullman	COMPILADORES. PRINCIPIOS, TÉCNICAS Y HERRAMIENTAS	PEARSON ADDISON WESLEY	2008 1998	1 3
Alfonseca Cubero E., Alfonseca Moreno M. , Mariyon Salomón R.	TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES	MCGRAW- HILL	2007	1

De Consulta:

Autor	Título	Editorial	Año Ed.	Dispon.
Louden Kenneth. C.	CONSTRUCCION DE COMPILADORES PRINCIPIOS Y PRÁCTICAS -	THOMSON INTERNACIONAL	2004	1
Dean Kelley	TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES	LONGMAN	1995	
Cases Muñoz Rafel - Marquez Villodre Lluís	LENGUAJES, GRAMÁTICAS Y AUTÓMATAS; CURSO BÁSICO	ALFAOMEGA	2002	1

**ESTRATEGIAS DIDÁCTICAS UTILIZADAS:**

- Clases expositivas
- Trabajos teórico - prácticos grupales e individuales
- Trabajos prácticos individuales

**RECURSOS DIDÁCTICOS UTILIZADOS:**

- Textos
- Pizarrón
- Multimedia
- WEB de la Cátedra
- Guías de trabajos prácticos

**PROGRAMA DE EXAMEN:**

Ídem Analítico.