

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERIA

ESCUELA DE SISTEMAS



NOMBRE DEL CURSO: Laboratorio de Lenguajes Formales y de Programación

CODIGO:	796	CREDITOS:	3
ESCUELA:	Ciencias y Sistemas	AREA A LA QUE PERTENECE:	Ciencias de la computación
PRE REQUISITOS:	770 - Introducción a la Programación 1 795 - Lógica de sistemas 960 - Matemática de cómputo 1	POST REQUISITO:	777 Organización de Lenguajes y Compiladores 1 772 Estructuras de Datos
CATEGORIA:	Obligatorio	SEMESTRE:	1er. 2020
CATEDRÁTICO (A):	Ing. Otto Rodríguez	AUXILIAR:	Elmer Real
EDIFICIO:	T-3	SECCIÓN:	A+
SALON DEL CURSO:	212	SALON DEL LABORATORIO:	India 1
HORAS POR SEMANA DEL CURSO:	2	HORAS POR SEMANA DEL LABORATORIO:	4
DÍAS QUE SE IMPARTE EL CURSO:	Martes	DIAS QUE SE IMPARTE EL LABORATORIO:	Martes
HORARIO DEL CURSO:	07:10 - 08:50 HRS	HORARIO DEL LABORATORIO:	10:50 - 12:30 12:30 - 14:10

DESCRIPCIÓN DEL CURSO:

El laboratorio tiene como propósito introducir al estudiante de ciencias de la computación al estudio, análisis, comprensión e implementación de lenguajes de programación bajo una estructura genérica que contribuya a que el estudiante sea capaz de manejar y diseñar gramáticas para lenguajes regulares, además, de modelos matemáticos que las resuelven; abarcando conocimientos de lenguajes reales conocidos.

Comprende también toda la base para que el estudiante sea capaz de entender sin problemas un curso avanzado de compiladores.

OBJETIVOS:***Objetivo General***

- Introducir al estudiante al conocimiento y desarrollo de las funciones básicas de los compiladores, y los ponga en práctica en la construcción de las primeras fases de éste.

Objetivos Específicos

- Aplicar los conocimientos adquiridos en clase para implementar analizadores léxicos.
- Implementar un analizador sintáctico utilizando las técnicas vistas en clase.

METODOLOGÍA:

- Se imparten clases presenciales con material de apoyo que es proporcionado al estudiante al finalizar la clase.
- En el transcurso del semestre se realizarán tareas y exámenes cortos para evaluar los conocimientos adquiridos.
- Se realizan prácticas y proyectos en donde se ponga en práctica y se puedan evaluar los conceptos adquiridos en el curso, tomando en cuenta que pueden incluirse temas de cursos pre requisito.
- Se impartirán clases prácticas donde se resuelvan problemas relacionados con el tema a desarrollar.

REQUISITOS:

- El laboratorio se debe aprobar con nota mínima de 61 puntos.
- Es obligatorio aprobar el laboratorio para tener derecho a examen final.
- Solo se calificarán exámenes y proyectos de estudiantes asignados en el curso. NO se agregan estudiantes en acta de notas.
- En este curso, no se pasan notas de semestres anteriores, no se guardan notas para semestres posteriores, y no se aceptan estudiantes con problemas de prerrequisitos.
- Copias parciales o totales en las tareas, investigaciones, etc. serán sancionadas con una nota de cero.
- Copias en los proyectos y prácticas serán sancionadas con una nota de cero y reportadas a la Escuela de Sistemas.
- Las tareas, investigaciones, prácticas, proyectos, deben ser entregadas en la fecha indicada y con el formato establecido.

EVALUACION:

Aspecto	Valor
Tareas, cortos	10 pts.
Prácticas	15 pts.
Práctica 1	15 pts.
Proyectos	65 pts.
Proyecto 1	30 pts.
Proyecto 2	35 pts.
Examen final	10 pts.
	Total 100 pts.

CONTENIDO

1. Lenguajes

- 1.1 Lenguajes Naturales
- 1.2 Características de lenguajes naturales
- 1.3 Lenguajes Formales
- 1.4 Características de lenguajes formales.

2. Lenguajes de Programación

- 2.1 Lenguaje de máquina
- 2.2 Lenguaje bajo nivel
- 2.3 Lenguaje alto nivel
- 2.4 Abstracciones de datos y control
- 2.5 Elementos de definición de lenguajes

3. Paradigmas de los lenguajes de programación

- 3.1 Imperativo
- 3.2 Declarativo

4. Jerarquía de Chomsky

- 4.1 Tipos de gramáticas
- 4.2 Restricciones
- 4.3 Ejemplos

5. Lenguajes regulares

- 5.1 Lenguajes regulares
- 5.2 Expresiones regulares
- 5.3 Ejemplos

6. Autómatas finitos

Deterministas

- 6.1 Definición
- 6.2 Tabla de transiciones
- 6.3 Autómatas Finitos Deterministas (AFD)
- 6.4 Implementación de AFD's
- 6.5 Ejemplos

7. Autómatas finitos NO Deterministas (NFA)

- 7.1 Definición
- 7.2 Procesamiento
- 7.3 Notación y grafos
- 7.4 Ejemplos

8. Implementación de un analizador léxico

- 8.1 Tabla de símbolos
- 8.2 Programación de un autómata finito
- 8.3 Manejo de errores

9. Lenguajes libres de contexto

- 9.1 10.1. Lenguajes Independientes del contexto
- 9.2 Gramáticas Tipo 2
- 9.3 Diseño de gramáticas independientes del contexto
- 9.4 Ejemplos y ejercicios de gramáticas libres de contexto
- 9.5 Recursividad
- 9.6 Por la izquierda
- 9.7 Por la derecha
- 9.8 Gramáticas ambiguas
- 9.9 Autómatas de Pila
- 9.10 Parser recursivo descendente
- 9.11 Ejemplos y ejercicios
- 9.12
- 9.14

BIBLIOGRAFÍA:

1. Louden y Lambert. (2010). **Lenguajes de programación**. 3 ed.
2. Hopcroft, John. y Ullman, Jeffrey (2008). **Introducción a la Teoría de Autómatas, Lenguajes y Computación**. 3 ed.
3. Brookshear, J. Glenn. Teoría de la Computación - **Lenguajes formales, autómatas y complejidad**. Addison-Wesley Iberoamericana