



MACROPROCESO: DOCENCIA  
PROCESO: GESTIÓN DE PROGRAMAS ACADÉMICOS  
PROCEDIMIENTO: FORMULACION O ACTUALIZACION DEL PROYECTO ACADÉMICO EDUCATIVO-PAE PARA PROGRAMAS DE  
PREGRADO  
CONTENIDOS PROGRAMATICOS PROGRAMAS DE PREGRADO

Código: D-GPA-P01-F02

Versión: 02

Página 1 de 3

Fecha: diciembre de 2021

**PROGRAMA ACADÉMICO:** Ingeniería de Sistemas y Computación

**SEMESTRE:** Séptimo

**ASIGNATURA:** Lenguajes Formales

**CÓDIGO:** 8108269

**NÚMERO DE CRÉDITOS:** Cuatro

### PRESENTACIÓN

Dentro del estudio de la ingeniería de sistemas, el conocimiento en las formas de interacción entre el hombre y la máquina, es fundamental, esto en función de entender el comportamiento del computador ante las solicitudes que puede realizar un usuario particular, es decir, qué puede escribir, cómo debe hacerlo y qué alcances pueden llegar a tener estas entradas dentro del dispositivo de cómputo. Es así, que esta asignatura busca ser una aproximación hacia el lenguaje que puede usarse para comunicarse con estos dispositivos, y pretende enseñar cuáles son sus formalismos propios para lograr una respuesta satisfactoria ante una entrada de datos en específico. De allí el nombre de Lenguajes Formales.

Como contenido temático, al interior de esta asignatura se estudiarán los modelos de máquinas computacionales, gramáticas formales y sus lenguajes correspondientes, así como la aplicación de estos modelos a los aspectos básicos del diseño de compiladores, gracias a los referentes del análisis léxico y el análisis sintáctico. Esto para que al final, el estudiante esté en capacidad de comprender la forma en que funciona cada autómatas, su correspondencia y aplicabilidad dentro del contexto de la ingeniería de sistemas.

Por otra parte, la asignatura cubre también los tópicos fundamentales referentes a la definición de un lenguaje y evidentemente parámetros que lo hacen formal, además de las estructuras gramaticales que lo gobiernan y sus aplicaciones en entornos computacionales. Todo lo anterior desde el marco de las matemáticas y teniendo en cuenta, su aplicabilidad dentro del entorno de los esquemas de comunicación hombre-máquina y las ciencias computacionales.

### JUSTIFICACIÓN

Cuando se ejecutan comandos en la consola de un sistema operativo, o se está escribiendo un programa en un lenguaje de programación determinado, siempre se usan palabras ordenadas en algún sentido, que buscan transformarse en instrucciones para el procesador, que de alguna forma intentará desarrollar en función de lograr un objetivo particular. Es así que, es importante poder conocer el origen de estas citadas palabras, su lenguaje y el por qué, cuándo y dónde se pueden usar dentro del proceso interactivo con un computador, ya que eso permite entender el tipo de comunicación que se puede establecer con la máquina, sus límites y la forma de poder implementar mecanismos más eficientes de escritura y lectura de programas de cómputo, merced a esto último de la creciente demanda en la disminución del tiempo de desarrollo de aplicaciones, la cual exige mecanismos más eficaces en procesos de esta índole, como la detección y corrección de errores en un código arbitrario, por citar solo un ejemplo.



Por otra parte, la aplicación de teoría de autómatas es aplicable a diferentes campos de la ingeniería de sistemas y computación, los cuales pueden reforzarse con esta asignatura, tal es el caso de la implementación de circuitos electrónicos, redes neuronales, cadenas de ensamble, control de procesos, robótica y protocolos de comunicaciones, como muestras del espectro disciplinar que abarca este espacio de estudio, además, claro sea dicho, de la lingüística computacional de la cual es base.

### COMPETENCIAS

- Identifica la estructura principal de un lenguaje formal, sus componentes y diferencias con otros tipos de lenguajes, enmarcado esto en los eminentes teóricos de Chomsky.
- Identifica las características y aplicaciones de los autómatas finitos deterministas y no deterministas, las diferentes máquinas abstractas aplicables a los lenguajes formales y la utilización de pilas de almacenamiento en lingüística computacional.
- Utiliza expresiones regulares para validar palabras propias de un lenguaje determinado, además de realizar las transformaciones desde estas hacia sus correspondientes autómatas y gramáticas.
- Identifica la arquitectura de un compilador teniendo en cuenta las capas que lo componen, los datos que se transmiten entre ellas y los archivos generados durante un ejercicio de compilación arbitrario.
- Construye códigos de validación, tanto léxicos como sintácticos, dentro de la arquitectura de un compilador, generando validaciones que utilicen todo el contenido del curso.

### RESULTADOS DE APRENDIZAJE

Utilizo los formalismos propios de los lenguajes formales de acuerdo a la jerarquía propuesta por Chomsky, incluyendo los procedimientos de generación de gramáticas regulares y de validación de artefactos lingüísticos (Autómatas Finitos y de Pila) para la implementación de analizadores léxico-sintácticos que coadyuven al mejoramiento del proceso de compilación de una solución informática desarrollada.

Desarrollo aplicaciones basadas en expresiones regulares a partir de la teoría de lenguajes formales y problemas específicos en el ámbito ingenieril (Programación y entradas en un sistema operativo) que puedan reconocer partículas léxicas, que permitan implementar motores de búsqueda en función de estructurar procesos de validación y normalización para los datos de entrada en una solución determinada.

Reconozco la transformación de un código fuente en un archivo ejecutable, a través del front end (analizadores léxicos, sintácticos y semánticos) y el back end de un compilador (Generador de código, intermedio, optimizador de código y generador de código objeto) en función de identificar el funcionamiento de cada una de estas capas dentro del proceso de desarrollo de aplicaciones en un ambiente de producción.

### METODOLOGÍA

Para el desarrollo de este ítem, se tendrán en cuenta las siguientes metodologías de aplicación:

#### Métodos base:

**Aprendizaje basado en problemas:** Con la aplicación de esta metodología se busca que los estudiantes, con la guía del docente, encuentren soluciones a problemáticas relativas a los lenguajes formales, en función del diseño y selección de soluciones informáticas para casos específicos planteados desde la misma clase, soportadas en un proceso de descubrimiento de los saberes que faciliten la estructuración de una respuesta adecuada a la situación planteada.



Para el desarrollo ABP se usará el esquema propuesto por Morales y Landa en 2004:

1. Leer y Analizar el escenario del problema
2. Realizar una lluvia de ideas
3. Hacer una lista con aquello que se conoce
4. Hacer una lista con aquello que no se conoce
5. Hacer una lista con aquello que necesita hacerse para resolver el problema
6. Definir el Problema
7. Obtener Información
8. Presentar resultados

Hay que notar para este particular que este proceso se realiza de forma iterativa. Esa así que desde el punto ocho puede volverse al uno para una segunda aproximación a la solución

Métodos de apoyo:

**Aprendizaje colaborativo:** Dentro de la planeación metodológica, se contempla el trabajo en pequeños grupos para la implementación de ABP. Es así que este componente se desarrollará de forma colaborativa, en aras de propiciar habilidades individuales y grupales, teniendo en cuenta la construcción de escenarios interactivos que generen, tanto el conocimiento, como la responsabilidad de su adquisición en los estudiantes (*Prescott, 1993*)

#### INVESTIGACIÓN

El estudiante podrá vincularse a los grupos de investigación del programa, al interior de la escuela, para iniciarse en procesos de formación como semillero de investigación en aspectos relativos a los lenguajes formales en particular, dentro del marco de la lingüística computacional. Análogamente, es posible desarrollar, proyectos de grado, en donde se centre la investigación en temas relacionados con esta área en específico, con el fin de generar documentos propios de este campo, como artículos o ponencias, los cuales, a su vez puedan ser sustentados en eventos destinados para este fin.

#### MEDIOS AUDIOVISUALES

Para el desarrollo de las clases se tendrán en cuenta los siguientes:

- Video Proyector
- Laboratorio de Computadores (aula de sistemas)
- Aplicaciones para construcción de autómatas, gramáticas y expresiones regulares.

#### EVALUACIÓN

##### EVALUACIÓN COLECTIVA

Dentro de los criterios a evaluar se tendrán en cuenta los siguientes:

- Desarrollo de Prácticas de Laboratorio
- Control de lecturas
- Participación en las sesiones
- Desarrollo de talleres en grupo
- Exposiciones de los temas relativos a la asignatura
- Actividades relacionadas con el proyecto a realizar



Además de las demás que se consideren necesarias para la implementación de los temas de clase, teniendo en cuenta la siguiente porcentual para cada parcial del curso:

Tabla: Porcentajes de calificación Colectiva: primer y segundo 50%

Ítem	Valor Porcentual
Talleres	30%
Exposiciones	10%
Proyecto final de la clase	20%
Total	60%

Fuente: El Autor

Es de notar que los talleres implican actividades que pueden realizarse dentro o fuera del salón de clases, y en general cualquier trabajo que sea aplicable en el contexto de la asignatura.

### EVALUACIÓN INDIVIDUAL

Para este particular se pueden observar los porcentajes en la siguiente tabla:

Tabla: Porcentajes de calificación Individual: primer y segundo 50%

Ítem	Valor Porcentual
Evaluaciones	40%
Total	40%

Fuente: El Autor

### CONTENIDOS TEMÁTICOS MINIMOS

#### Unidad 1: Introducción a los lenguajes formales, autómatas y gramáticas

- Introducción e historia de las gramáticas y los lenguajes formales
- Maquinas abstractas y lenguajes formales
- Tipos de lenguajes, alfabetos y palabras
- Teorema de la jerarquía de Chomsky
- Definición, demostraciones, tipos y componentes de una gramática
- Introducción a los autómatas finitos
- Arquitectura de un autómata finito (deterministas, no deterministas, con transiciones  $\lambda$ )
- Equivalencias entre autómatas finitos.
- Autómatas finitos y gramáticas regulares

#### Unidad 2: Expresiones Regulares, autómatas de pila y gramáticas libres de contexto

- Definición, lenguaje y propiedades de una expresión regular
- Ecuaciones de expresiones regulares
- Relación autómata finito – expresión regular
- Aplicaciones de expresiones regulares
- Arquitectura y definición de un autómata con pila
- Configuración, cálculo y lenguaje aceptado en un AP



### Unidad 3: lenguajes regulares y máquinas de Turing.

- Minimización de autómatas finitos.
- Arquitectura de una máquina de Turing básica
- Configuración y cálculo en una máquina de Turing
- Lenguajes y máquinas de Turing
- Tipos de máquinas de Turing

### Unidad 4: introducción y fundamentación en compiladores

- Introducción a los Compiladores
- Analizador Léxico
- Analizador Sintáctico
- Analizador Semántico

### Unidad 4: introducción y fundamentación en compiladores (Continuación)

- Generador de código intermedio
- Optimización de código
- Generador de código objeto

## LECTURAS MÍNIMAS

### [Teoría de Autómatas y Lenguajes Formales URL](#)

Autor: Francisco Vico. Departamento de Lenguajes y Ciencias de la Computación. Área de conocimiento: Ciencias de la Computación e Inteligencia Artificial. ETSI Informática. Universidad de Málaga. 2015. 14 páginas

Licencia: Creative Commons. CC: BY - NC - SA

### [Apuntes de Teoría de Autómatas y Lenguajes Formales URL](#)

Autor: García Fernández, Luis Amable; Martínez Vidal, María Gloria. Universitat Jaume I. Departament d'Enginyeria i Ciència dels Computadors. 2005. 37 páginas.

Licencia: Creative Commons. CC: BY - ND - NC

### [Teoría de la Computación \(Lenguajes Formales, Computabilidad y Complejidad\) Apuntes y Ejercicios](#)

Autor: Gonzalo Navarro, Departamento de Ciencias de la Computación. Universidad de Chile. 2016. 178 páginas

Licencia: Creative Commons: CC BY - NC - ND 2.5

## BIBLIOGRAFÍA

**Libros físicos:**

