

# SÍLABO

## Teoría de Autómatas y Lenguajes Formales

(Borrador consolidado)

Docente: Helder Octavio Fernández Guzmán

Versión: v1.3 (08-ene-2026 a 05-feb-2026)

---

### 1. Datos generales

- **Asignatura:** Teoría de Autómatas y Lenguajes Formales
- **Carrera/Programa:** Ingeniería Informática / Ciencias de la Computación (o equivalente)
- **Nivel:** Pregrado (intermedio)
- **Modalidad:** Mixta (presencial + sesiones sincrónicas en línea)
- **Duración:** Lunes a sábado, del **08 de enero de 2026** al **05 de febrero de 2026** (sin domingos)
- **Carga diaria:** 6 periodos académicos por día (45 minutos cada uno)
- **Total estimado:** 25 días de clase (112.5 horas reloj aprox.)
- **Pre-requisitos recomendados:** Matemática Discreta, Programación (bases), nociones de lógica, conjuntos y demostración básica.
- **Plataforma de apoyo:** Moodle (gestión de materiales y evidencias), herramienta de práctica con retroalimentación (p. ej., Automata Tutor), y/o simuladores (p. ej., JFLAP).

### 2. Descripción del curso

La asignatura introduce los fundamentos de **lenguajes formales** y **modelos de cómputo** desde una perspectiva aplicada a la Ingeniería en Ciencias de la Computación. El curso prioriza el aprendizaje significativo: comprender **para qué sirven** los modelos (autómatas y gramáticas), **cómo se construyen**, y **cómo se justifican** sus propiedades, especialmente en el marco de validación de patrones, especificación de sintaxis y reconocimiento de límites de expresividad.

El enfoque es intensivo y combina sesiones presenciales con sesiones sincrónicas en línea dedicadas a práctica (modelación, verificación con casos de prueba, y argumentación breve). Se privilegia la resolución de problemas y el uso de herramientas que entregan retroalimentación inmediata.

### 3. Propósito formativo

Al finalizar el curso, el estudiante podrá modelar formalmente lenguajes y especificaciones, construir autómatas y gramáticas para lenguajes relevantes en computación, y reconocer límites de expresividad mediante técnicas estándar (incluyendo el Lema de Bombeo para lenguajes regulares y, de forma acotada, para lenguajes libres de contexto).

### 4. Resultados de aprendizaje

Al finalizar la asignatura, el estudiante será capaz de:

1. Usar notación formal (alfabetos, cadenas, lenguajes y operaciones) para describir lenguajes y justificar propiedades simples.
2. Construir y validar autómatas finitos (AFD/AFN y  $\varepsilon$ -AFN), convertir AFN  $\rightarrow$  AFD por subconjuntos y minimizar AFD en casos típicos.
3. Diseñar y manipular expresiones regulares para describir lenguajes regulares y verificar ejemplos.
4. Aplicar el **Pumping Lemma** para mostrar **no regularidad** en casos típicos, cuidando cuantificadores y estructura de contradicción.
5. Diseñar **gramáticas libres de contexto** (CFG), construir derivaciones y/o árboles de derivación, e identificar ambigüedad en ejemplos simples, proponiendo un ajuste mínimo cuando corresponda.
6. Clasificar lenguajes (a nivel introductorio) como **regular** / **CFL** / **ninguno** usando patrones y una justificación breve (frontera micro).

**Profundización (TA2, trabajo adicional del 2do parcial):** investigación guiada con un ejemplo ilustrativo de **AST** y un ejemplo ilustrativo de **Pumping Lemma para CFL**. Este trabajo se entrega por Moodle y no reemplaza la preparación del EE2.

## 5. Contenidos por unidades (selección priorizada)

### Unidad 0: Fundamentos (inicio)

Alfabetos, cadenas y lenguajes; operaciones con lenguajes; criterios de corrección de un modelo y de una justificación mínima.

### Unidad 1: Lenguajes regulares y autómatas finitos (núcleo)

AFD: diseño y verificación; AFN y  $\varepsilon$ -AFN; conversión AFN  $\rightarrow$  AFD (subconjuntos); expresiones regulares (diseño/interpretación); propiedades básicas de cerradura (aplicadas); minimización de AFD (procedimiento básico); introducción al Lema de Bombeo (regular) como frontera.

### Unidad 2: Gramáticas libres de contexto y sintaxis (núcleo)

CFG: diseño para lenguajes típicos; derivaciones y árboles; ambigüedad (identificación y ajuste mínimo); mini-lenguajes (listas, paréntesis, expresiones) y reglas de precedencia/asociatividad a nivel práctico.

### Unidad 3: Fronteras de expresividad (cierre)

Lema de Bombeo para regulares (consolidación) y clasificación regular/CFL/ninguno con justificación breve (frontera micro). (Complejidad P/NP: opcional, solo mención conceptual si el tiempo lo permite).

## 6. Cronograma (v1.3)

### Reglas organizativas

- **Miércoles y sábados:** actividades prácticas sincrónicas en línea (preferentemente con Automata Tutor u otra herramienta equivalente) y evidencia registrada en Moodle.
- **Días exclusivos de evaluación escrita:** 1er Parcial, 2do Parcial, Examen Final y 2da Instancia.
- **Feriado nacional:** viernes 23 de enero de 2026 (sin clases).

Fecha	Modalidad	Contenido/Actividad
08-ene-2026 (jue)	Presencial/mixto	Unidad 0: Introducción, notación, operaciones con lenguajes, criterios de corrección/justificación.
09-ene-2026 (vie)	Presencial/mixto	Regulares: AFD (diseño por restricciones, trazas, verificación).
10-ene-2026 (sáb)	Online sincrónico	<b>AP1 P-R1:</b> Práctica AFD (patrones básicos) + evidencia en Moodle.
12-ene-2026 (lun)	Presencial/mixto	Regulares: AFN y $\varepsilon$ -AFN (intuición, diseño).
13-ene-2026 (mar)	Presencial/mixto	Regulares: AFN $\rightarrow$ AFD (subconjuntos) paso a paso.
14-ene-2026 (mié)	Online sincrónico	<b>AP1 P-R2:</b> AFN $\rightarrow$ AFD (subconjuntos) + evidencia.
15-ene-2026 (jue)	Presencial/mixto	Regulares: Expresiones regulares (diseño/interpretación, utilidad en patrones).
16-ene-2026 (vie)	Presencial/mixto	Regulares: cerradura esencial (aplicada) + ejercicios cortos.
17-ene-2026 (sáb)	Online sincrónico	<b>AP1 P-R4:</b> Regex + validación con casos + evidencia.
19-ene-2026 (lun)	Presencial/mixto	Regulares: minimización básica de AFD (procedimiento).
20-ene-2026 (mar)	Presencial/mixto	Frontera (regular): Lema de Bombeo (estructura y patrones típicos).
21-ene-2026 (mié)	Online sincrónico	<b>AP1 P-R3</b> + mini-simulacro EE1 + repaso integrado y criterios.
22-ene-2026 (jue)	Evaluación escrita	<b>1er Parcial P1 (EE1).</b>
23-ene-2026 (vie)	—	<b>FERIADO nacional (sin clases).</b>
24-ene-2026 (sáb)	—	<b>Sin actividad (no se contabiliza avance).</b>
26-ene-2026 (lun)	Presencial/mixto	CFG: diseño para lenguajes típicos + pertenencia (genera/no genera).
27-ene-2026 (mar)	Presencial/mixto	CFG: derivaciones (izq./der.) y ejercicios.
28-ene-2026 (mié)	Online sincrónico	<b>AP2 P-C1:</b> derivación + árbol + evidencia (taller sincrónico).
29-ene-2026 (jue)	Presencial/mixto	CFG: árboles de derivación + ambigüedad (detección con evidencia).
30-ene-2026 (vie)	Presencial/mixto	CFG: ajuste mínimo de ambigüedad (precedencia/asociatividad) + mini-lenguajes.
31-ene-2026 (sáb)	Online sincrónico	<b>AP2 P-C2:</b> reto integrador CFG (diseño+derivación+árbol+ambigüedad) + evidencia.
02-feb-2026 (lun)	Presencial/mixto	Frontera (micro): clasificación regular/CFL/ninguno + simulacro EE2 (escrito).

Continúa en la siguiente página.

Fecha	Modalidad	Contenido/Actividad
03-feb-2026 (mar)	Evaluación escrita	<b>2do Parcial P2 (EE2).</b>
04-feb-2026 (mié)	Evaluación escrita	<b>Examen Final (EF).</b>
05-feb-2026 (jue)	Evaluación escrita	<b>2da Instancia (SI).</b>
05-feb-2026 (jue)	Asincrónico (Moodle)	<b>Entrega TA2:</b> AST + Pumping CFL (ejemplo guiado) hasta 23:59.

## 7. Metodología de enseñanza–aprendizaje

### Estrategia general

Se aplica un enfoque activo con ciclos cortos: **concepto** → **ejemplo** → **práctica guiada** → **práctica autónoma con retroalimentación**. Se prioriza la comprensión de utilidad y la modelación correcta antes que la memorización de definiciones.

### Sesiones presenciales (lunes, martes, jueves y viernes)

- Exposición breve de conceptos esenciales y demostraciones guiadas.
- Resolución de ejercicios representativos (con discusión de errores frecuentes).
- Práctica dirigida tipo evaluación y verificación con casos de prueba.

### Sesiones prácticas sincrónicas en línea (miércoles y sábados)

- Taller/laboratorio centrado en modelación (autómatas/gramáticas) y validación.
- Mini-retos cronometrados y retroalimentación inmediata.
- Registro de evidencias (AP1/AP2) en la plataforma institucional.

## 8. Sistema de evaluación y reglas de aprobación

### Componentes

Los parciales combinan evaluación práctica y evaluación escrita:

$$P1 = 0.50 \cdot AP1 + 0.50 \cdot EE1 \quad P2 = 0.45 \cdot AP2 + 0.45 \cdot EE2 + 0.10 \cdot TA2$$

$$PP = \frac{P1 + P2}{2}$$

### Reglas de aprobación

El estudiante aprueba la asignatura si cumple al menos una de las siguientes condiciones:

- **Aprobación por parciales:** si  $PP \geq 51$ .
- **Aprobación por examen final:** si  $EF \geq 51$ .
- **Segunda instancia:** habilitada solo si  $PP \geq 26$ , y aprobación si  $SI \geq 51$ .

### Carácter obligatorio de prácticas (formulación institucional)

Las evidencias prácticas (AP1/AP2) se registran a partir de los laboratorios sincrónicos (miércoles y sábados) y constituyen requisito para el cómputo de la componente práctica de cada parcial.

## Rúbrica mínima sugerida para prácticas (AP1/AP2)

Cada práctica puede evaluarse con los siguientes criterios (25 puntos c/u):

1. Corrección del modelo.
2. Verificación con casos (aceptadas/rechazadas o derivación/árbol correcto).
3. Claridad y presentación.
4. Justificación mínima (3–6 líneas).

## 9. Temario evaluable (EE1, EE2, Final, 2da instancia)

### EE1 (1er Parcial — escrito formal)

#### Entra:

- AFD: diseño y verificación.
- AFN/ $\varepsilon$ -AFN y conversión  $\text{AFN} \rightarrow \text{AFD}$  (subconjuntos).
- Expresiones regulares (diseño o interpretación).
- Cerradura esencial (en ejercicios cortos).
- No regularidad (introducción): Lema de Bombeo (regular) en un caso guiado o argumento breve equivalente.

#### No entra:

- CFG/AST (derivaciones, ambigüedad) como contenido principal de examen.
- Lema de Bombeo para CFL.
- Computabilidad/MT como eje de evaluación.

### EE2 (2do Parcial — escrito formal)

#### Entra:

- CFG: diseño para lenguajes típicos.
- Derivaciones (izq./der.) y/o árboles de derivación.
- Ambigüedad: detección con evidencia y ajuste mínimo (precedencia/asociatividad) en un caso simple.
- Frontera micro: clasificación regular/CFL/ninguno con justificación breve.

#### No entra:

- AST como contenido central de examen (se trabaja en TA2).
- Pumping CFL como prueba completa en examen escrito (se trabaja en TA2 como ejemplo ilustrativo).
- Parsers completos; CNF/CYK u otras transformaciones avanzadas.
- Computabilidad/MT como contenido central evaluable.

### Examen Final (EF) y Segunda Instancia (SI)

Evaluaciones integradoras centradas en lo esencial del curso:

- Un problema de regulares (AFD/ $\text{AFN} \rightarrow \text{AFD}$ /regex).
- Un problema de CFG (diseño + derivación/árbol o ambigüedad).
- Un problema de frontera (pumping regular o clasificación regular/CFL/ninguno con argumento breve).

## 10. Bibliografía y recursos (con URL)

### Bibliografía básica (libros)

- M. Sipser, *Introduction to the Theory of Computation*. Página del autor: <https://math.mit.edu/~sipser/book.html>

- J. E. Hopcroft, R. Motwani, J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*. (Pearson): <https://www.pearson.com/en-us/subject-catalog/p/introduction-to-automata-theory-languages-and-computation/P200000003517/9780321455369>
- D. Kozen, *Automata and Computability*. (SpringerLink): <https://link.springer.com/book/10.1007/978-1-4612-1844-9>
- P. Linz, *An Introduction to Formal Languages and Automata*. (J&B Learning): <https://www.jblearning.com/catalog/productdetails/9781284231601>

## Recursos abiertos y apoyo

- MIT OpenCourseWare 6.045J: <https://ocw.mit.edu/courses/6-045j-automata-computability-and-formal-languages/>
- Stanford CS154 (handouts): <https://theory.stanford.edu/~rajeev/CS154/Handouts.html>

## Herramientas

- Automata Tutor: <https://www.cs.cit.tum.de/tcs/tools/>
- Automata Tutor v3 (arXiv): <https://arxiv.org/abs/2005.01419>
- JFLAP: <https://www.jflap.org/>
- Moodle VPL: [https://moodle.org/plugins/mod\\_vpl](https://moodle.org/plugins/mod_vpl)
- LTI en Moodle: [https://docs.moodle.org/en/LTI\\_and\\_Moodle](https://docs.moodle.org/en/LTI_and_Moodle)

## 11. Políticas del curso (síntesis)

- **Integridad académica:** se aplican las disposiciones institucionales vigentes ante prácticas indebidas.
- **Puntualidad y entregas:** las actividades con tiempo definido deben respetar fechas y horarios establecidos.
- **Comunicación:** avisos oficiales y evidencias se gestionan vía plataforma institucional (p. ej., Moodle).