

## PLAN DOCENTE DE LA ASIGNATURA

**Curso académico: 2024/2025**

Identificación y características de la asignatura			
Código	501321	Créditos ECTS	6
Denominación (español)	Teoría de Lenguajes		
Denominación (inglés)	Theory of Languages		
Titulaciones	Grado en Ingeniería Informática en Ingeniería del Software		
Centro	Escuela Politécnica		
Semestre	6º	Carácter	Obligatorio
Módulo	Tecnología Específica en Ingeniería del Software		
Materia	Informática teórica		
Profesorado			
Nombre	Despacho	Correo-e	Página web
Alberto Gómez Mancha	10	<a href="mailto:agomez@unex.es">agomez@unex.es</a>	epcc.unex.es
Julia González Rodríguez	25	<a href="mailto:juliagon@unex.es">juliagon@unex.es</a>	
Javier Berrocal Olmeda	117	<a href="mailto:jberolm@unex.es">jberolm@unex.es</a>	
Área de conocimiento	Lenguajes y Sistemas Informáticos		
Departamento	Ingeniería de Sistemas Informáticos y Telemáticos		
Profesor coordinador (si hay más de uno)	Julia González Rodríguez		
Competencias			
<b>Competencias básicas (comunes a todas las enseñanzas de Grado)</b>			
<ul style="list-style-type: none"> <li>• <b>CB1:</b> Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio.</li> <li>• <b>CB2:</b> Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.</li> <li>• <b>CB3:</b> Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.</li> <li>• <b>CB4:</b> Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.</li> <li>• <b>CB5:</b> Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.</li> <li>• <b>CG08:</b> Conocimiento de las materias básicas y tecnológicas, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.</li> <li>• <b>CG09:</b> Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.</li> </ul>			

**Según el plan de estudio aprobado, esta asignatura debe cubrir, parcialmente, las siguientes competencias técnica y transversales.**

- **CIS04:** Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.
- **CT03:** Capacidad para resolver problemas.
- **CT16:** Capacidad para adaptarse a nuevas situaciones y cambios.

## Contenidos

### Breve descripción del contenido

Hay que tener en cuenta que los contenidos que aparecen en el plan de estudios asociados a la materia de Informática teórica son los siguientes: Gramáticas y lenguajes formales: jerarquía de Chomsky. Teoría de autómatas. Compiladores e intérpretes: estructura y diseño. Introducción a la teoría de la computación.

Los lenguajes de programación constituyen una herramienta básica en la mayor parte de las actividades que debe realizar un graduado en ing. del software, por ese motivo su capacidad para elegir el lenguaje más apropiado puede resultar crucial dentro del desarrollo de un proyecto de software.

Esta asignatura aportará una visión global de los lenguajes de programación y de los diferentes paradigmas de programación. Además, sentará las bases necesarias para el conocimiento y el desarrollo del proceso de traducción de los lenguajes (gramáticas, lenguajes formales y autómatas).

Además, el estudio de las máquinas de Turing sienta la base teórica que permite establecer los límites de la computabilidad y la complejidad algorítmica.

### Temario de la asignatura

#### **Primera parte: Fundamentos sobre traductores y lenguajes formales**

Denominación del tema 1: Conceptos básicos

Contenidos del tema 1: Alfabetos, lenguajes, gramáticas, compiladores, intérpretes. La jerarquía de gramáticas de Chomsky. Teoría de Autómatas: Autómatas Finitos, Autómatas de pila, Máquinas de Turing. Paradigmas de programación

Denominación del tema 2: Análisis léxico

Contenidos del tema 2: Expresiones regulares, los AF como reconocedores de lenguajes. Descripción de las actividades prácticas del tema 2: Diseño e implementación de analizadores léxicos

Denominación del tema 3: Análisis sintáctico

Contenidos del tema 3: Gramáticas ind. del contexto. Árbol de derivación. Ambigüedad  
Descripción de las actividades prácticas del tema 3: Diseño e implementación de analizadores sintácticos

Denominación del tema 4: Análisis semántico

Contenidos del tema 4: Gramáticas atribuidas  
Descripción de las actividades prácticas del tema 4: Diseño e implementación de analizadores semánticos

#### **Segunda parte: Los límites de la computación**

Denominación del tema 5: Introducción a la teoría de la computabilidad

Contenidos del tema 5: Computabilidad y máquinas de Turing. El problema de la parada. Recursividad.

Denominación del tema 6: Introducción a la teoría de la complejidad

Contenidos del tema 6: Complejidad y máquinas de Turing. Problemas NP-completos. El problema P-NP.

### Actividades formativas

Horas de trabajo del alumno/a por tema		Horas teóricas	Actividades prácticas				Actividad de seguimiento	No presencial
Tema	Total	GG	PCH	LAB	ORD	SEM	TP	EP
1	17	7		0			0	10
2	34	10		6			1	17
3	37	9		8			1	19
4	20	3		8			1	8
5	10	3		0			0	7
6	10	3		0			0	7
<b>Evaluación</b>	22	2.5		0.5			0	19
<b>TOTAL</b>	150	37.5		22.5			3	87

GG: Grupo Grande (85 estudiantes).

PCH: prácticas clínicas hospitalarias (7 estudiantes)

LAB: prácticas laboratorio o campo (15 estudiantes)

ORD: prácticas sala ordenador o laboratorio de idiomas (20 estudiantes)

SEM: clases problemas o seminarios o casos prácticos (40 estudiantes).

TP: Tutorías Programadas (seguimiento docente, tipo tutorías ECTS).

EP: Estudio personal, trabajos individuales o en grupo, y lectura de bibliografía.

### Metodologías docentes

En el desarrollo de la asignatura se combinarán distintas actividades que permitan la participación activa del estudiante.

A continuación, se nombran algunas de las actividades formativas que se plantearán a lo largo del curso para alcanzar los resultados de aprendizaje de la asignatura.

Aunque cada actividad sólo se detalla dentro de una modalidad (presenciales en grupo grande, presenciales en laboratorio y no presenciales), algunas de ellas se desarrollarán en varias.

Algunas de estas actividades se realizarán de forma individual y otras en grupo.

#### **Presenciales en grupo grande**

- Clase expositiva
- Clase de explicación de ejercicios y problemas
- Método del caso
- Resolución de test
- Resolución de ejercicios y problemas
- Trabajo en grupo para alcanzar acuerdos y desarrollar problemas en común
- Ensayo de examen
- Exámenes de problemas
- Presentación de problemas resueltos

#### **Presenciales en laboratorio**

- Clase de demostración de software
- Trazas de algoritmos
- Laboratorios guiados
- Laboratorios abiertos
- Portafolio de actividades
- Evaluación entre pares, con aplicación de rúbrica
- Autoevaluación, con aplicación de rúbrica
- Prueba y detección de errores en programas
- Proyecto de programación (traductor para lenguajes formales)
- Modificación de programas para incorporar nuevas funcionalidades
- Resolución de cuestionarios

### **No presenciales**

- Estudio individual
- Reuniones de grupo
- Búsqueda de información
- Plantear preguntas de test
- Seguimiento de problemas resueltos
- Estudio inicial de temas
- Implementación de programas
- Acceso a documentación del aula virtual
- Consulta de páginas web
- Visualización de vídeos de funcionamiento de herramientas, de explicación de conceptos y de resolución de problemas
- Comunicación con profesorado y estudiantes mediante foros

### **Resultados de aprendizaje**

Se han establecido los siguientes resultados de aprendizaje concretos de esta asignatura:

- RA1. Diseñar una gramática que permita generar un lenguaje formal dado, identificando el tipo de gramática necesitada según la jerarquía de Chomsky
- RA2. Construir autómatas que permitan resolver problemas en general y específicamente que permitan el reconocimiento de un lenguaje, sabiendo determinar el tipo de autómata que se necesita en cada caso
- RA3. Evaluar y optimizar el diseño de una gramática formal dada
- RA4. Optimizar el diseño de un autómata finito
- RA5. Construir un traductor para un lenguaje, a partir de sus especificaciones, utilizando herramientas que permitan total o parcialmente la generación automática de analizadores léxicos, sintácticos y semánticos.
- RA6. Explicar con claridad las diferencias entre clases P y NP de complejidad algorítmica, así como el concepto y la importancia de los problemas NP-completos
- RA7. Explicar los límites que tiene la computabilidad y enunciar ejemplos de problemas no computables

### **Sistemas de evaluación**

Tal como se contempla en la vigente normativa de evaluación de la Universidad de Extremadura, esta asignatura puede superarse siguiendo las modalidades de evaluación continua (E.C.) y de evaluación global (E.G.).

La elección de la modalidad de evaluación corresponde a cada estudiante. Se podrá elegir el sistema de evaluación de cada convocatoria en una consulta del aula virtual de la asignatura durante las cuatro primeras semanas de impartición de la asignatura. Si no se comunica el tipo de evaluación elegido en el plazo indicado se supondrá que se opta por la modalidad de evaluación continua.

A continuación, se detallan las características de ambos sistemas de evaluación.

### **Instrumentos de evaluación**

En esta asignatura se utilizan distintos instrumentos de evaluación que aseguran la adquisición por parte del estudiante de todas las competencias (conocimientos, habilidades y actitudes) de las cuales derivan los objetivos propuestos. Se han considerado adecuados los siguientes instrumentos de

evaluación:

- Actividades
- Proyectos
- Pruebas escritas

Estos son instrumentos generales que integran otros más directos y simples y que permiten realizar un análisis completo de los niveles de consecución de las distintas competencias.

### ***Actividades***

Son actividades simples realizadas por cada estudiante a lo largo del semestre (individualmente o en grupo, según se determine), recopilando trabajos realizados en las clases presenciales (en grupo grande y en laboratorio) o durante el tiempo dedicado a su trabajo personal.

Estas actividades pueden ser de muchos tipos: resolución de problemas, resolución de test, propuestas de nuevos problemas, corrección de trabajos de estudiantes, búsqueda de información, programas, actividades del aula virtual, trabajos en grupo, actas de reuniones, mapas conceptuales, etc.

### ***Proyectos***

Los proyectos son instrumentos que permiten evaluar muchas de las competencias técnicas y transversales de la futura graduada o graduado en Ingeniería Informática, al tratarse de una actividad muy próxima a la que deberá enfrentarse regularmente durante su vida laboral: la resolución de un problema planteado, utilizando herramientas cuyo manejo conoce.

En el caso de esta asignatura, los proyectos a realizar consistirán en el diseño de lenguajes formales y en la construcción de traductores para dichos lenguajes. Para la realización de estos proyectos los estudiantes utilizarán generadores automáticos de analizadores léxicos y sintácticos.

Se realizarán individualmente o en grupo, según se determine.

### ***Pruebas escritas***

Para asegurar la adquisición de los conocimientos y habilidades mínimos de las competencias técnicas se realizarán pruebas escritas que consistirán en la resolución de problemas, preguntas de tipo test, preguntas cortas, etc.

### **Criterios de evaluación**

Esta asignatura se puede superar según dos modalidades de evaluación distintas: evaluación continua (E.C.) o evaluación global (E.G.)

#### **Evaluación continua:**

- Para superar esta asignatura deben superarse los requisitos mínimos de los bloques de Pruebas escritas, Proyectos y Actividades
- La nota en un bloque superado (cumplidos los requisitos mínimos) se guardará durante todas las convocatorias del curso (siempre que se tenga derecho a examen en la convocatoria en la que se supere el bloque)
- La puntuación de cada bloque se calculará sobre 10
- Obtendrán una calificación de "No presentado" en la convocatoria ordinaria los estudiantes que no tengan aprobado ningún bloque por evaluación continua y no se presenten a ninguna prueba realizada durante el periodo de exámenes de esa convocatoria. En las convocatorias extraordinarias, obtendrán una calificación de "No presentado" los estudiantes que no se presenten a ninguna prueba en dicha convocatoria.

### **Bloque 1: Actividades**

- La nota del bloque de actividades, **Act**, representa el 20% de la nota final de la asignatura.
- Este bloque se llevará a cabo por evaluación continua realizando una serie de actividades que se irán proponiendo a lo largo del curso y no será posible recuperar o modificar la calificación en las convocatorias extraordinarias.
- Según su naturaleza, las actividades se realizarán en las clases de teoría, laboratorio, tutorías programadas o en horario no presencial utilizando el aula virtual.
- No es necesario obtener una nota mínima en este bloque para considerarlo superado.
- La asistencia a las clases de laboratorio y tutorías programadas será considerada, a efectos de evaluación, como una actividad más.
- La nota de este bloque se obtiene a partir de la ponderación de las calificaciones obtenidas en las actividades propuestas.
- La nota obtenida en este bloque no es recuperable en las siguientes convocatorias oficiales del curso.
- Se realizarán individualmente o en grupo, según se determine.

### **Bloque 2: Proyecto**

- La nota del bloque de proyecto, **Pro**, representa el 40% de la nota final de la asignatura.
- Es obligatorio superar el proyecto con una nota mínima de 5 sobre 10.
- Es responsabilidad de cada estudiante la custodia y protección de su proyecto.
- Son requisitos indispensables para superar este bloque: entregar los programas y la documentación solicitados, cumpliendo los requisitos mínimos que se establezcan, superar la defensa del proyecto y utilizar adecuadamente las herramientas de desarrollo.
- La nota de este bloque es recuperable en las convocatorias extraordinarias de la asignatura, para lo que será necesario presentar el proyecto solicitado y superar la defensa de dicho proyecto.
- Se realizarán individualmente o en grupo, según se determine.

### **Bloque 3: Pruebas escritas**

- La nota de este bloque, **PruEs**, representa el 40% de la nota final de la asignatura.
- Para superar este bloque es necesario obtener una nota mínima de 5 sobre 10.
- Parte de este bloque podrá superarse por parciales en la convocatoria ordinaria.
- Cada prueba, parcial o final, podrá estar compuesta de preguntas de test o de respuestas cortas y resolución de problemas, con requisitos adicionales sobre la nota mínima que debe obtenerse en cada prueba para poder calcular la nota media.

### **Evaluación global:**

- Para superar esta asignatura con evaluación global deben superarse los requisitos mínimos de las dos partes de las que consta el examen de la convocatoria: Pruebas y Proyecto
- La puntuación de cada bloque se calculará sobre 10
- La nota de una parte superada no se guardará para ninguna convocatoria posterior del curso.

### **Bloque 1: Proyecto**

- La nota del bloque de proyecto, **Pro**, representa el 40% de la nota final de la asignatura.
- Es obligatorio superar cada proyecto con una nota mínima de 5 sobre 10.
- Son requisitos indispensables para superar este parte: entregar los programas y la documentación solicitados cumpliendo los requisitos mínimos que se establezcan, superar la defensa del proyecto y utilizar adecuadamente las herramientas de desarrollo.

### **Bloque 2: Pruebas escritas**

- La nota de este bloque, **PruEs**, representa el 60% de la nota final de la asignatura.
- Para superar este bloque es necesario obtener una nota mínima de 5 sobre 10.
- Esta prueba podrá estar compuesta de preguntas de test o de respuestas cortas y resolución de problemas, con requisitos adicionales sobre la nota mínima que debe obtenerse en cada prueba para poder calcular la nota media.

Los proyectos, las defensas de los proyectos y las pruebas escritas finales serán las mismas, independientemente del modelo de evaluación que se haya seleccionado.

### Cálculo de la nota final de la asignatura

- La copia o el plagio en cualquier actividad o prueba supone una nota final de SUSPENSO (0) en la convocatoria y una nota de 0 en todas las calificaciones obtenidas hasta el momento para todas las personas implicadas, además de las actuaciones legales pertinentes.
- En el caso de la evaluación continua, si se cumplen todos los requisitos mínimos de los tres bloques, la nota final se calcula como la siguiente media ponderada:
  - $\text{Nota Final} = \text{PruEs} * 0,4 + \text{Pro} * 0,4 + \text{Act} * 0,2$
- En el caso de la evaluación global, si se cumplen todos los requisitos mínimos de las dos partes, la nota final se calcula como la siguiente media ponderada:
  - $\text{Nota Final} = \text{PruEs} * 0,6 + \text{Pro} * 0,4$
- Si se sigue la modalidad de evaluación continua y se cumplen los requisitos mínimos de los bloques, la nota final se calculará con la fórmula de nota final con la que se obtenga un resultado más alto.
- Si no se cumplen todos los requisitos mínimos, la nota final será 2 o 3, dependiendo de los casos.

En la siguiente tabla se puede consultar cómo calcular la nota final, según los resultados obtenidos en los dos bloques obligatorios de proyecto y pruebas escritas:

<b>Pruebas escritas</b>	NP	NP	NP	NCR	NCR	Nota	Nota
<b>Proyecto</b>	NP	NCR	Nota	NP	NCR/Nota	NCR/NP	Nota
<b>Nota final (*)</b>	NP	2	3	2	3	3	Media ponderada

NP: no se ha presentado a ese bloque

NCR: no se cumplen los requisitos mínimos de ese bloque (Nota < 5)

Nota: nota obtenida en el bloque, superados los requisitos mínimos

(\*) Obtendrán una calificación de "No presentado" en la convocatoria ordinaria quien no tenga superado ningún bloque por evaluación continua y no se presenten a ninguna prueba realizada durante el periodo de exámenes de esa convocatoria.

En las convocatorias extraordinarias obtendrán una calificación de "No presentado" quien no se presenten a ninguna prueba en la convocatoria.

### Sistema de revisión y comentario de exámenes

El día de cada examen o prueba final de evaluación será anunciada la fecha de publicación de las notas, así como la fecha de revisión del examen o prueba.

Se podrán comentar y revisar los resultados de las pruebas, en las fechas previstas, de acuerdo con la normativa vigente.

### Bibliografía (básica y complementaria)

En el aula de la asignatura en el Campus Virtual de la UEx se encontrarán más recursos y referencias actualizados.

### **Bibliografía básica**

- Hopcroft J.E., Motwani R. y Ullman J. D. "Introducción a la Teoría de Autómatas, Lenguajes y Computación". Pearson-Addison Wesley, 2008
- Mitchell J.C. "Concept in Programming Languages". Cambridge University Press, 2002
- Jurado E., "Teoría de Autómatas y Lenguajes Formales" Manuales UEX nº 55, Servicio de Publicaciones (edición on-line), 2008
- Alfonseca E., Alfonseca M. y Morrión R. "Teoría de Autómatas y Lenguajes Formales". McGraw-Hill, 2007
- Isasi P., Martínez P. y Borrajo D. "Lenguajes, Gramáticas y Autómatas. Un enfoque práctico". Addison Wesley, 2001
- Alfonseca M., Sancho y Orga. "Teoría de Lenguajes, Gramáticas y Autómatas". Ed. Universidad, 1993
- Martin, J. "Lenguajes Formales y Teoría de la Computación". McGraw-Hill, 2003
- Brookshear. "Teoría de la computación. Lenguajes Formales, Autómatas y Complejidad". Addison-Wesley, 1993
- Kelley, D. "Teoría de Autómatas y Lenguajes formales". Prentice-Hall, 1995
- Tucker, A. "Programming Languages". McGraw-Hill, 1985

### **Bibliografía para el trabajo de laboratorio**

- Alfonseca, de la Cruz, Ortega y Pulido "Compiladores e Intérpretes: Teoría y Práctica" Prentice-Hall, 2006
- Appel, A. "Modern Compiler Implementation in Java/C". Cambridge University Press, 1998
- Aho, A.V.; Sethi, R. y Ullman, J.D. "Compilers: Principles, Techniques and Tools". Addison Wesley, 1986 (existe traducción en castellano)

### **Otros recursos y materiales docentes complementarios**

#### *Medios materiales utilizados*

- Pizarra
- Cañón de vídeo
- Ordenador

#### *Materiales y recursos utilizados*

Los materiales y recursos utilizados están en versión electrónica en el aula virtual de la asignatura:

- Apuntes y presentaciones utilizadas para cada tema del programa
- Guiones y material de ejemplo para las sesiones de laboratorio
- Agenda del curso

Son recursos propios del aula virtual los siguientes:

- Sistemas de participación
  - Foros de comunicación
  - Tablón de anuncios y novedades
- Información adicional
  - Recopilación de código fuente de programas
  - Conjunto de referencias web relacionadas con la asignatura
  - Vídeos explicativos
- Autoevaluación
  - Test de autoevaluación de contenidos
  - Problemas de autoevaluación

- Baterías de preguntas de test
- Tareas virtuales para la entrega de problemas
- Talleres de coevaluación

#### Tutorías de libre acceso

Las tutorías del profesorado de la asignatura están publicadas en sus respectivos despachos, en el aula virtual de la asignatura y en la web del centro.

Se pueden concertar reuniones fuera de los horarios fijados.

También se resolverán dudas a través de los espacios de comunicación del aula virtual.