



1. Identificación

1.1. De la asignatura

Curso Académico	2025/2026
Titulación	GRADO EN INGENIERÍA INFORMÁTICA, PROGRAMA ACADÉMICO DE SIMULTANEIDAD DE DOBLE TITULACIÓN CON ITINERARIO ESPECIFICO DE GRADO EN MATEMÁTICAS Y GRADO EN INGENIERÍA INFORMÁTICA, PROGRAMA ACADÉMICO DE SIMULTANEIDAD DE DOBLE TITULACIÓN CON ITINERARIO ESPECÍFICO DE GRADO EN MATEMÁTICAS Y GRADO EN INGENIERÍA INFORMÁTICA
Nombre de la asignatura	MODELOS DE COMPUTACIÓN
Código	3862
Curso	CUARTO QUINTO QUINTO
Carácter	OPTATIVA
Número de grupos	3
Créditos ECTS	6.0
Estimación del volumen de trabajo	150.0 150.0 150.0
Organización temporal	2º Cuatrimestre 2º Cuatrimestre 2º Cuatrimestre
Idiomas en que se imparte	Español

1.2. Del profesorado: Equipo docente

VALDES VELA, MERCEDES

Docente: GRUPO 1, PCEO MATE+INFOR_EXTINCION GRUPO PCEO MATE+INFORM,

Coordinación de los grupos: GRUPO 1, PCEO MATE+INFOR_EXTINCION GRUPO PCEO MATE+INFORM,

Coordinador de la asignatura

Categoría

PROFESORES TITULARES DE UNIVERSIDAD

Área

Departamento

INGENIERÍA DE LA INFORMACIÓN Y LAS COMUNICACIONES

Correo electrónico / Página web / Tutoría electrónica

mdvaldes@um.es Tutoría electrónica: No

Teléfono, horario y lugar de atención al alumnado

2. Presentación

Esta asignatura se enmarca dentro de la **Teoría de la Computación**. El objetivo final de esta disciplina es dar respuesta a la siguiente pregunta: ¿Cuáles son las capacidades fundamentales y las limitaciones de los ordenadores? La Teoría de la Computación estudia, por tanto, los fundamentos matemáticos que subyacen en la Informática, tanto en el hardware como en el software. Dentro de esta disciplina se distinguen tres áreas: la Teoría de Autómatas, la Teoría de la Computabilidad y la Teoría de la Complejidad:

2.1. La Teoría de Autómatas, trata con las definiciones y propiedades de los formalismos matemáticos de la computación: los distintos tipos de autómatas y sus capacidades para procesar los distintos tipos de lenguajes formales entre los que se incluyen, obviamente, los lenguajes de programación.

2.2. La Teoría de la Computabilidad trata de establecer las características que distinguen a los llamados **problemas computables** (resolubles mediante un programa de ordenador) de los **no computables** o, equivalentemente, **los lenguajes decidibles** de los **no decidibles**. Así, en la primera mitad del siglo XX, matemáticos como Kurt Gödel, Alan Turing o Alonzo Church, identificaron problemas matemáticos que nunca podrían ser resueltos por ordenadores por muy potentes que éstos llegaran a ser. El ejemplo paradigmático de este tipo de problemas es el de determinar (automáticamente) si una sentencia lógica cualquiera es verdadera o falsa. El largo proceso para llegar a ese resultado llevó consigo la aparición de la **Máquina de Turing**, modelo matemático de computación sobre la cual se construye toda la Teoría de la Computabilidad y que constituyen la base de diseño de los ordenadores de hoy en día.

2.3. Por su parte, la Teoría de la Complejidad, se centra en los problemas que sí son computables, para categorizarlos según su grado de dificultad, atendiendo al tiempo necesario o espacio necesario para computarlos. La Teoría de la Complejidad intenta responder a la pregunta: ¿Dónde radica la diferencia fundamental entre aquellos problemas computacionalmente duros y los que no lo son tanto? Aunque se esté intentando resolver desde hace décadas, a día de hoy, no se tiene respuesta a esta pregunta. Lo que sí se ha logrado, es establecer un esquema para clasificar los problemas según su dificultad computacional y mecanismos para evidenciar que un problema es computacionalmente duro. Un área de aplicación donde la Teoría de la Complejidad juega un papel muy importante es la Criptografía, ya que son los problemas computacionales más difíciles los que inspiran a los criptógrafos en el diseño de los mecanismos de seguridad de la información.

Dicho esto y, como se verá, dada la dualidad de problemas (no) computables - lenguaje (no) decidible, en la asignatura de Modelos de Computación dedicamos un pequeño periodo a repasar los conceptos procedentes de la asignatura de Autómatas y Lenguajes Formales los cuales sirven de base para las dos siguientes partes que conforman el grueso de la asignatura: la Computabilidad y la Complejidad.

Debemos recalcar que ésta es una asignatura eminentemente teórica. En ella se tratan "grandes" ideas que pueden verse como filosóficas; pero también se tratan pequeños detalles que llegan a ser engorrosos. Sin embargo, la **teoría es importante** porque:

2.1. proporciona las herramientas conceptuales que los ingenieros informáticos utilizan en la resolución de problemas: los autómatas finitos se usan para el procesamiento de texto, el diseño de hardware y el diseño de compiladores; las gramáticas libres de contexto se utilizan en el diseño de lenguajes de programación y en Inteligencia Artificial;

2.2. proporciona una imagen elegante y sencilla de los ordenadores y los algoritmos: las máquinas de Turing;

2.3. porque abre la mente y proporciona competencias duraderas: mientras que los conocimientos tecnológicos de hoy día quedarán obsoletos en unos pocos años pues la tecnología de los ordenadores cambia rápidamente, la capacidad de razonamiento, de resolución de problemas y de abstracción, son valores duraderos y el estudio de la teoría nos entrena en este sentido.

3. Condiciones de acceso a la asignatura

3.1. Incompatibilidades

No constan

3.2. Requisitos

No constan

3.3. Recomendaciones

Haber adquirido los conocimientos de la asignatura de Fundamentos Lógicos de la Informática y tener conocimientos básicos de teoría de conjuntos y técnicas de demostración.

Haber adquirido los conocimientos de la asignatura de Autómatas y Lenguajes Formales.

Haber adquirido los conocimientos de las asignaturas de Algoritmos y Estructuras de datos I y II.

4. Competencias

4.1. Competencias básicas

No constan

4.2. Competencias de la titulación

- CGII1: Capacidad de análisis y síntesis.
- CGII7: Resolución de problemas.
- CGII14: Razonamiento crítico.

4.3. Competencias transversales y de materia

- C1 - Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática
- C2 - Capacidad para conocer los fundamentos teóricos de los lenguajes de programación y las técnicas de procesamiento léxico, sintáctico y semántico asociadas, y saber aplicarlas para la creación, diseño y procesamiento de lenguajes
- C3 - Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos

5. Contenidos

5.1. Teoría

Bloque 1: Introducción

Tema 1: Introducción

- Presentación
- Complejidad, Computabilidad y Autómatas
- Nociones Matemáticas y Terminología

Bloque 2: Autómatas y Lenguajes

Tema 2: Lenguajes Regulares

- Autómatas Finitos (AFs)
- Lenguajes Regulares

Tema 3: Lenguajes Libres de Contexto

- Autómatas con Pila
- Lenguajes Libres de Contexto

Bloque 3: Computabilidad

Tema 4: La Tesis de Church-Turing

- Máquinas de Turing
- Variantes de la Máquina de Turing
- Concepto de Equivalencia Expresiva
- Hacia la Definición Formal de Algoritmo
- Computabilidad y Complejidad

Tema 5: Reducibilidad

- Lenguajes Indecidibles

- Mapping-Reducibilidad
- El Teorema de Rice

Tema 6: Decibilidad

- Lenguajes Decidibles
- La Máquina de Turing Universal
- Indecibilidad

Bloque 4: Complejidad

Tema 7: Complejidad en Tiempo

- Complejidad en tiempo
- La Clase P
- La Clase NP

Tema 8: NP-Complejidad

- Reducibilidad Polinómica
- Concepto de NP-Complejidad
- Teorema de Cook-Lein
- Algunos Problemas NP-Complejos

5.2. Prácticas

■ Práctica 1: Ejercicios y problemas sobre Autómatas, Lenguajes y Problemas de Decisión

Sobre los diferentes modelos de computación y clases de lenguajes formales se harán preguntas de V/F con justificación, preguntas tipo test en las que escoger una de entre tres posibles respuestas; rellenar huecos en un texto; rellenar tablas (justificando la respuesta); encontrar el intruso entre varios lenguajes; conectar hipótesis con conclusiones; clasificar lenguajes y modelos de computación; enumerar, ordenar, identificar clases de lenguajes y modelos de computación, etc

Relacionado con:

- Tema 1: Introducción
- Bloque 2: Autómatas y Lenguajes
- Tema 2: Lenguajes Regulares
- Tema 3: Lenguajes Libres de Contexto

■ Práctica 2: Ejercicios y problemas sobre Computabilidad

Sobre las cuestiones de computabilidad, decibilidad, recursiva-enumerabilidad y reducibilidad se harán preguntas de V/F con justificación, preguntas tipo test en las que escoger una de entre tres posibles respuestas; rellenar huecos en un texto; rellenar tablas (justificando la respuesta); encontrar el intruso entre varios lenguajes; calcular alguna característica de alguna reducción, etc; conectar hipótesis con conclusiones; enumerar, ordenar o identificar clases de lenguajes y modelos de computación, etc Se aplicarán estrategias para deducir la clase de computabilidad de un problema a partir de otros cuya clase de computabilidad es conocida

Relacionado con:

- Bloque 3: Computabilidad
- Tema 4: La Tesis de Church-Turing
- Tema 5: Reducibilidad
- Tema 6: Decibilidad

■ Práctica 3: Ejercicios y Problemas sobre Complejidad

Sobre las cuestiones de complejidad en tiempo, NP-completitud y reducciones polinómicas se harán preguntas de V/F con justificación, preguntas tipo test en las que escoger una de entre tres posibles respuestas ; rellenar huecos en un texto; rellenar tablas (justificando la respuesta); encontrar el intruso entre varios lenguajes; calcular alguna característica de alguna reducción polinómica, orden de complejidad, etc; conectar hipótesis con conclusiones; enumerar, ordenar o identificar clases de lenguajes y modelos de computación, etc Se aplicarán estrategias para deducir la clase de complejidad de un problema a partir de otros cuya clase de complejidad es conocida

Relacionado con:

- Bloque 4: Complejidad
- Tema 7: Complejidad en Tiempo
- Tema 8: NP-Complejidad

6. Actividades Formativas

Actividad Formativa	Metodología	Horas	Presencialidad
A1: Actividades con grupo grande de alumnos entre las que se encuentran la presentación en el aula de los conceptos propios de la materia mediante metodología expositiva con lecciones magistrales participativas y medios audiovisuales. También se contemplan en este grupo las actividades de evaluación teórico prácticas.		36.0	40.0
A2: Actividades con grupo mediano en el aula de resolución de problemas, seminarios, charlas, ejercicios basados en el aprendizaje orientado a proyectos, estudios de casos, exposición y discusión de trabajos relativas al seguimiento individual y/o grupal de adquisición de las competencias.		24.0	12.5
A5: Estudio y trabajo autónomo orientado a la asimilación de contenidos, realización de problemas, ejercicios o redacción de informes técnicos o memorias descriptivas, desarrollo de proyectos o prácticas individuales o en		90.0	0.0
	Totales	150,00	

7. Horario de la asignatura

8. Sistemas de Evaluación

Identificador	Denominación del instrumento de evaluación	Criterios de Valoración	Ponderación
IE1	Examen teórico-práctico. En este instrumento incluimos desde el tradicional examen escrito o tipo test hasta los exámenes basados en resolución de problemas, pasando por los de tipo mixto que incluyen cuestiones cortas o de desarrollo teórico junto con pequeños problemas. También se incluye aquí la consideración de la participación activa del alumno en clase, la entrega de ejercicios o realización de pequeños trabajos escritos y presentaciones.	<p>La asignatura sigue el modelo de evaluación continua formado por dos tareas y un examen final. Pero si algún estudiante no alcanza los requisitos para seguir este modelo entonces será evaluado mediante evaluación global. En el apartado de Observaciones se pueden ver cómo se usan los instrumentos de evaluación según si se sigue un tipo de evaluación u otro, así como los requisitos para acceder a cada una de esas modalidades.</p> <ul style="list-style-type: none">■ Evaluación continua: el examen final tiene ejercicios y problemas cortos que ponen en relación los contenidos de los distintos bloques de Autómatas, Computabilidad y Complejidad.■ Modelo de evaluación global: el examen global contiene ejercicios y problemas cortos así como ejercicios de desarrollo y demostración, que ponen en relación los contenidos de los distintos bloques de Autómatas, Computabilidad y Complejidad.■ Criterios de valoración:<ul style="list-style-type: none">■ Correcto uso de los conceptos, resultados, teoremas.■ Correcto manejo de la terminología, tecnicismos y nomenclatura de la asignatura.■ No puntuarán aquellos ejercicios en los que se pida justificación y ésta no se proporcione.■ No se considerarán justificaciones válidas aquellas que sólo repitan de una manera o de otra la propia afirmación que se esté cuestionando.■ No se considerarán justificaciones válidas aquellas que sean ilógicas, bizarras y enredadas.	50.0

IE2	<p>Informe técnico. En este instrumento incluimos los resultados de actividades prácticas, o de laboratorio, junto con sus memorias descriptivas. Los resúmenes del estado del arte o memorias de investigación sobre temas concretos. Y la posibilidad de realizar entrevistas personales o presentaciones de los trabajos realizados también entran en esta categoría.</p>	<p>La asignatura sigue el modelo de evaluación continua formado por dos tareas y un examen final. Pero si algún estudiante no alcanza los requisitos para seguir este modelo entonces será evaluado mediante evaluación global. En el apartado de Observaciones se pueden ver cómo se usan los instrumentos de evaluación según si se sigue un tipo de evaluación u otro, así como los requisitos para acceder a cada una de esas modalidades.</p> <ul style="list-style-type: none"> ■ Modelo de evaluación continua: ■ Primera tarea: Entrega, por grupos, de un documento con la resolución de ejercicios sobre el bloque de Autómatas y Lenguajes y el bloque de Computabilidad. ■ Segunda Tarea: Entrega, por grupos, de una presentación asíncrona (vídeo) explicando la demostración de dos o tres problemas de clasificación de lenguajes según su complejidad. ■ Criterios de valoración: ■ Correcto uso de los conceptos, resultados, teoremas. ■ Correcto manejo de la terminología, tecnicismos y nomenclatura de la asignatura. ■ No puntuarán aquellos ejercicios en los que se pida justificación y ésta no se proporcione. ■ No se considerarán justificaciones válidas aquellas que sólo repitan de una manera o de otra la propia afirmación que se esté cuestionando. ■ No se considerarán justificaciones válidas aquellas que sean ilógicas, bizarras y enredadas. ■ Se penalizará que no se respeten todas y cada una de las instrucciones. ■ La profesora podrá ver necesario convocar a algunos grupos de prácticas a una defensa mediante una entrevista personal. ■ Si un estudiante es requerido a entrevista, no acudir implica un 0 en cada una de las tareas. 	50.0
-----	--	---	------

- La nota inicialmente asignada puede bajarse por un mala defensa, como no contestar con soltura a las preguntas que se plantean
- En grupos de dos o tres estudiantes **la nota puede ser diferente para cada uno**, como consecuencia de una diferencia apreciable en la defensa que cada uno hace de la práctica, o de un reparto descompensado de las tareas realizadas.

9. Fechas de exámenes

<https://www.um.es/web/estudios/grados/informatica/2025-26#exámenes>

10. Resultados del Aprendizaje

10.1. Establecer la relación entre la resolución de un problema y el cálculo algorítmico de una función y, específicamente, entre la resolución de un problema de decisión, el cálculo de un predicado y el problema de la pertenencia de una cadena a un lenguaje formal.

10.2. Explicar en qué consiste un modelo de computación y argumentar la necesidad de establecer modelos de computación.

10.3. Exponer las razones por las cuales las máquinas de Turing se consideran un modelo de computación de propósito general.

10.4. Razonar sobre el alcance de la tesis de Church-Turing, en cuanto al poder computacional de las máquinas de Turing, la equivalencia entre modelos de computación y las limitaciones teóricas de la computación.

10.5. Indicar las características principales de las gramáticas y autómatas para lenguajes regulares y libres del contexto y evaluar las ventajas y limitaciones de estos formalismos como modelo de algoritmos de procesamiento de lenguajes. En particular, establecer la relación entre la resolución de un problema de pertenencia a este tipo de lenguajes y el diseño de algoritmos para las distintas fases de análisis y traducción de lenguajes de programación.

10.6. Distinguir entre problemas computables y no computables y, específicamente, entre problemas de decisión decidibles (recursivos), indecidibles (no recursivos), semidecidibles (recursivamente enumerables) y no semidecidibles (no-RE).

10.7. Describir algunos problemas indecidibles sobre gramáticas libres del contexto, sobre parada de programas y sobre propiedades semánticas de los programas y analizar las implicaciones prácticas de estos resultados.

10.8. Aplicar técnicas que permiten deducir que un problema nuevo no tiene solución algorítmica, directamente o a través de su relación con un problema indecidible conocido.

10.9. Diferenciar entre problemas tratables o intratables. Distinguir entre problemas de la clase P, NP y problemas NP-completos.

10.10. Explicar el alcance de la cuestión no resuelta ¿P es distinto de NP?, en cuanto a lo que supondría si se demostrara o refutara dicha la conjetura.

10.11. Aplicar técnicas que permiten deducir si un problema es de una clase de complejidad u otra y, de ese modo, analizar si es un problema de complejidad polinomial y, si no lo es, evaluar hasta qué punto se puede considerar intratable.

10.12. Razonar sobre limitaciones teóricas y prácticas de la computación y las posibilidades de reducir estas restricciones en un futuro.

10.13. Exponer algunas estrategias generales que se siguen para intentar abordar problemas intratables.

11. Bibliografía

Bibliografía básica

- [Introduction to the theory of computation / Michael Sipser.-- 3rd Ed.-- Delhi : Cengage Learning India, 2013.](#)
- [Apuntes de Modelos de Computación](#)

Bibliografía complementaria

- [El quinteto de Cambridge : una obra de especulación científica / John L. Casti Editorial: Madrid : Taurus, 1998. Colección: Pensamiento](#)
- [Introducción a la teoría de autómatas, lenguajes y computación /John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman.-- Madrid : Pearson Addison Wesley, 2008.](#)
- [La nueva mente del emperador / Roger Penrose.-- Madrid : Mondadori, 1991.](#)

12. Observaciones

La asignatura sigue el modelo de **evaluación continua**. Pero si algún estudiante no alcanza los requisitos para seguir este modelo entonces será evaluado mediante **evaluación global**

La asignatura se aprueba con una **Calificación final (CF)** igual o superior a 5.

Modelo de evaluación continua:

- CF se compone de dos partes:
- nota de teoría (NT): obtenida mediante examen final.
- nota de prácticas (NP): obtenida a partir de las notas de las dos tareas entregadas a lo largo del curso.
- $CF = (NT * 50 + NP * 50) / 100$.
- La nota NP obtenida en junio tras haber entregado las dos tareas se mantiene en julio y en febrero siguientes.

Modelo de evaluación global:

- El 100% de la calificación se obtiene mediante un examen global.

Cómo se opta a evaluación continua/ global:

- Para optar a evaluación continua basta con entregar en tiempo y forma las dos tareas, las cuales serán propuestas a lo largo del curso (al final del bloque 3 y al final del bloque 4).
- Mas allá de las fechas de entrega durante el curso no hay más fechas de entrega.

- Si no se entrega en fecha alguna de las dos tareas planteadas a lo largo del curso, se entiende que el estudiante opta por evaluación global.

Observaciones sobre el REVA

El artículo 86 del Reglamento de Evaluación de Estudiantes (REVA) prevé: "Salvo en el caso de actividades definidas como obligatorias en la guía docente, si el o la estudiante no puede seguir el proceso de evaluación continua por circunstancias sobrevenidas debidamente justificadas, tendrá derecho a realizar una prueba global".

Observaciones sobre estudiantes con discapacidad o necesidades educativas especiales

Aquellos estudiantes con discapacidad o necesidades educativas especiales pueden dirigirse al Servicio de Atención a la Diversidad y Voluntariado (ADYV; <https://www.umes/adyv/inicio>) para recibir orientación sobre un mejor aprovechamiento de su proceso formativo, y en su caso, la adopción de medidas de equiparación y de mejora para la inclusión, en virtud de la Resolución Rectoral R358/2016. El tratamiento de la información sobre este alumnado, en cumplimiento con la LOPD, es de estricta confidencialidad.

Observaciones sobre ODS

Esta asignatura no está relacionada con ninguno de los objetivos de desarrollo sostenible.

NECESIDADES EDUCATIVAS ESPECIALES

Aquellos estudiantes con discapacidad o necesidades educativas especiales podrán dirigirse al Servicio de Atención a la Diversidad y Voluntariado (ADYV - <https://www.um.es/adyv>) para recibir orientación sobre un mejor aprovechamiento de su proceso formativo y, en su caso, la adopción de medidas de equiparación y de mejora para la inclusión, en virtud de la Resolución Rectoral R-358/2016. El tratamiento de la información sobre este alumnado, en cumplimiento con la LOPD, es de estricta confidencialidad.

REGLAMENTO DE EVALUACIÓN DE ESTUDIANTES

El artículo 8.6 del Reglamento de Evaluación de Estudiantes (REVA) prevé que "salvo en el caso de actividades definidas como obligatorias en la guía docente, si el o la estudiante no puede seguir el proceso de evaluación continua por circunstancias sobrevenidas debidamente justificadas, tendrá derecho a realizar una prueba global".

Se recuerda asimismo que el artículo 22.1 del Reglamento de Evaluación de Estudiantes (REVA) estipula que "el o la estudiante que se valga de conductas fraudulentas, incluida la indebida atribución de identidad o autoría, o esté en posesión de medios o instrumentos que faciliten dichas conductas, obtendrá la calificación de cero en el procedimiento de evaluación y, en su caso, podrá ser objeto de sanción, previa apertura de expediente disciplinario".