



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA

GUÍA DOCENTE

CURSO 2025-26

GRADO EN INGENIERÍA INFORMÁTICA

DATOS DE LA ASIGNATURA

Nombre:

MODELOS AVANZADOS DE COMPUTACIÓN

Denominación en Inglés:

Advanced Models of Computation

Código:

606010237

Tipo Docencia:

Presencial

Carácter:

Obligatoria

Horas:

Totales

Presenciales

No Presenciales

Trabajo Estimado

150

60

90

Créditos:

Grupos Grandes

Grupos Reducidos

Aula estándar

Laboratorio

Prácticas de campo

Aula de informática

4

0

2

0

0

Departamentos:

TECNOLOGIAS DE LA INFORMACION

Áreas de Conocimiento:

CIENCIA DE LA COMPUTACION E INTELIG. ARTIFICIAL

Curso:

4º - Cuarto

Cuatrimestre

Primer cuatrimestre

DATOS DEL PROFESORADO (*Profesorado coordinador de la asignatura)

Nombre:	E-mail:	Teléfono:
* Francisco Jose Moreno Velo	francisco.moreno@dti.uhu.es	959 217 659
Antonio Palanco Salguero	antonio.palanco@dedu.uhu.es	959 219 978

Datos adicionales del profesorado (Tutorías, Horarios, Despachos, etc...)

Francisco José Moreno Velo

- **Correo:** francisco.moreno@dti.uhu.es
- **Teléfono:** +34 959 21 76 59
- **Despacho:** Campus El Carmen - Edificio ETSI - Despacho ETP-141
- **Horario:**
<https://www.uhu.es/etsi/informacion-academica/informacion-comun-todos-los-titulos/horarios-2/>
- **Tutorías:**
<https://www.uhu.es/etsi/informacion-academica/informacion-comun-todos-los-titulos/horarios-2/>

Antonio Palanco Salguero

- **Correo:** antonio.palanco@dti.uhu.es
- **Teléfono:** +34 959 21 73 94
- **Despacho:** Campus El Carmen - Edificio ETSI - Despacho ETP-101
- **Horario:**
<https://www.uhu.es/etsi/informacion-academica/informacion-comun-todos-los-titulos/horarios-2/>
- **Tutorías:**
<https://www.uhu.es/etsi/informacion-academica/informacion-comun-todos-los-titulos/horarios-2/>

DATOS ESPECÍFICOS DE LA ASIGNATURA

1. Descripción de Contenidos:

1.1 Breve descripción (en Castellano):

La máquina de Turing. Solución de problemas mediante máquinas de Turing.
Los límites de la máquina de Turing. El teorema de Turing. Problemas no computables.
Complejidad P, NP y PSPACE.
Problemas NP-completos
Modelos de computación alternativos. La máquina RAM.
Funciones recursivas y parcialmente recursivas.
Conjuntos recursivos y recursivamente enumerables.
Equivalencia entre modelos de computación. La tesis de Church-Turing.
Modelos de computación paralela: máquinas PRAM y circuitos booleanos
Complejidad en tiempo paralelo.

1.2 Breve descripción (en Inglés):

The Turing machine. Troubleshooting by Turing machines.
The limits of the Turing machine. Turing's theorem. Non-computable problems.
Complexity P, NP and PSPACE.
NP-complete problems
Alternative computing models. RAM machine.
Recursive and partially recursive functions.
Recursive and recursively enumerable sets.
Equivalence between computer models. The Church-Turing thesis.
Models of parallel computation: PRAM machines and Boolean circuits
Parallel time complexity

2. Situación de la asignatura:

2.1 Contexto dentro de la titulación:

Esta asignatura complementa la formación relativa a la complejidad y el análisis del algoritmos. Los conceptos impartidos en esta asignatura son básicos para entender cuanto tardará un ordenador en resolver un determinado problema. En el contexto de la informática actual, con un creciente volumen de datos (Big Data) originados en redes sociales, sistemas de adquisición de datos o en sistemas de gestión empresarial, se hace si cabe más importante analizar la complejidad de los algoritmos, de forma que el ingeniero cuente con un soporte sólido a la hora de diseñar los sistemas de información actuales. Además, veremos que tipo de problemas son resolubles con un determinado modelo de computador (por ej. una máquina de Turing) en un tiempo razonable (que llamaremos P) y que problemas necesitan mucho más tiempo para su resolución (problemas NP) o incluso aquellos que no son resolubles con un determinado modelo de computador. Así mismo estudiaremos diferentes modelos de computadores distintos a los tradicionales como las máquinas de acceso aleatorio, las funciones recursivas, el cálculo lambda y los sistemas de producción.

2.2 Recomendaciones

Serán recomendables conocimientos básicos de programación.

3. Objetivos (expresados como resultado del aprendizaje)

- Conocer el modelo de la Máquina de Turing, su alcance y limitaciones.
- Conocer otros modelos de computación (máquinas RAM, lenguajes algorítmicos sencillos, modelos funcionales) y las relaciones existentes (tesis de Church-Turing).
- Conocer los conceptos de funciones recursivas y parcialmente recursivas.
- Conocer los conceptos de problemas decidibles y semidecidibles.
- Conocer las clases de complejidad computacional más importantes y las relaciones entre ellas.
- Comprender la NP-completitud. Ser capaz de comprobar si un problema es NP-completo.
- Conocer las clases de complejidad para aproximar problemas. Saber clasificar problemas concretos en dichas clases.
- Conocer la jerarquía polinómica. Saber ubicar problemas dentro de dicha jerarquía. Conocer problemas PESPACIO completos.
- Conocer y relacionar los modelos de computación paralela: máquinas PRAM y circuitos booleanos.
- Determinar problemas P-completos. Relacionar la complejidad en tiempo paralelo con la complejidad en espacio secuencial.

4. Competencias a adquirir por los estudiantes

4.1 Competencias específicas:

CE1-C: 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.

4.2 Competencias básicas, generales o transversales:

CB5: Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

CG0: Capacidad de análisis y síntesis: Encontrar, analizar, criticar (razonamiento crítico), relacionar, estructurar y sintetizar información proveniente de diversas fuentes, así como integrar ideas y conocimientos.

CG03: Capacidad para la resolución de problemas.

CG04: Capacidad para tomar decisiones basadas en criterios objetivos (datos experimentales, científicos o de simulación disponibles) así como capacidad de argumentar y justificar lógicamente

dichas decisiones, sabiendo aceptar otros puntos de vista.

CG07: Motivación por la calidad y la mejora continua, actuando con rigor, responsabilidad y ética profesional.

CG08: Capacidad para adaptarse a las tecnologías y a los futuros entornos actualizando las competencias profesionales.

CG09: Capacidad para innovar y generar nuevas ideas.

CT2: Desarrollo de una actitud crítica en relación con la capacidad de análisis y síntesis.

CT4: Capacidad de utilizar las Competencias Informáticas e Informacionales (CI2) en la práctica profesional.

CT3: Desarrollo de una actitud de indagación que permita la revisión y avance permanente del conocimiento.

5. Actividades Formativas y Metodologías Docentes

5.1 Actividades formativas:

- Sesiones de Teoría sobre los contenidos del Programa
- Sesiones Prácticas en Laboratorios Especializados o en Aulas de Informática
- Actividades Académicamente Dirigidas por el Profesorado: seminarios, conferencias, desarrollo de trabajos, debates, tutorías colectivas, actividades de evaluación y autoevaluación...
- Trabajo Individual/Autónomo del Estudiante

5.2 Metodologías Docentes:

- Desarrollo de Prácticas en Laboratorios Especializados o Aulas de Informática en grupos reducidos
- Clase Magistral Participativa
- Resolución de Problemas y Ejercicios Prácticos
- Tutorías Individuales o Colectivas. Interacción directa profesorado-estudiantes
- Planteamiento, Realización, Tutorización y Presentación de Trabajos.
- Evaluaciones y Exámenes

5.3 Desarrollo y Justificación:

- Sesiones de teoría sobre los contenidos del programa

Las clases teóricas tendrán una duración de 3 horas. En ellas se expondrá y explicará, con ayuda del cañón de proyecciones y/o la pizarra los contenidos asociados a cada tema. Habrá bibliografía específica de cada tema disponibles en la web de la asignatura con antelación suficiente.

- Sesiones de Prácticas en Laboratorios Especializados o Aulas de Informática en grupos reducidos

Las sesiones de prácticas se desarrollarán en aulas provistas de ordenadores y tendrán una duración de 1,5 horas. En estas prácticas se impartirán los contenidos de Programación Funcional utilizando para ello el lenguaje de programación Haskell, así como una introducción al lenguaje Scala.

- Actividades acedémicamente dirigidas por el profesorado

A lo largo del curso se planteará uno trabajo práctico referido al desarrollo de un proyecto utilizando el lenguaje de programación Haskell o Scala. Este trabajo se considera una actividad académica dirigida y su explicación se realizará en el horario de las sesiones de prácticas. El seguimiento de este trabajo se realizará en tutorías individualizadas.

- Trabajo individual/autónomo del estudiante

El trabajo planteado se deberá realizar de forma individual.

6. Temario Desarrollado

Temario de teoría

Tema 1. Introducción a los Modelos de Computación

- 1.1 Definiciones
- 1.2 Galería de personajes
- 1.3 Preliminares matemáticos
- 1.4 Algunos modelos de computación

Tema 2. Circuitos lógicos

- 2.1 Funciones sobre conjuntos finitos
- 2.2 Programación lineal
- 2.3 Circuitos lógicos
- 2.4 Formas normales
- 2.5 Circuitos espacializados
- 2.6 Límites en tamaño y profundidad

Tema 3. Autómatas finitos

- 3.1 Funciones sobre conjuntos infinitos numerables
- 3.2 Autómatas finitos deterministas
- 3.3 Autómatas finitos no deterministas
- 3.4 El lema de bombeo para autómatas finitos

Tema 4. Autómatas de pila

- 4.1 Autómatas de pila
- 4.2 Autómatas deterministas y no deterministas
- 4.3 Lenguajes formales y gramáticas
- 4.4 La Forma Normal de Chomsky
- 4.5 Gramáticas Libres de Contexto y Autómatas de Pila
- 4.6 El lema de bombeo para autómatas de pila

Tema 5. Máquinas de Turing

- 5.1 Límites de los autómatas
- 5.2 Definición de Máquina de Turing
- 5.3 Ejemplos
- 5.4 Máquina de Turing con N cintas
- 5.5 Máquina de Turing no determinista
- 5.6 Máquina de Turing Universal
- 5.7 Representación de autómatas con Máquinas de Turing
- 5.8 Lenguajes aceptados por Máquinas de Turing

Tema 6. Decidibilidad

- 6.1 Lenguajes reconocibles y decidibles
- 6.2 Problemas decidibles sobre lenguajes regulares
- 6.3 Problemas decidibles sobre gramáticas libres de contexto
- 6.4 El problema de la parada
- 6.5 Un lenguaje no reconocible
- 6.6 Problemas no decidibles

Tema 7. Funciones recursivas

- 7.1 Funciones recursivas primitivas
- 7.2 Limitaciones de las funciones primitivas recursivas
- 7.3 Funciones recursivas parciales
- 7.4 Codificación de funciones
- 7.5 Funciones universales
- 7.6 Decidibilidad e indecidibilidad
- 7.7 Equivalencia con las Máquinas de Turing

Tema 8. Complejidad temporal

- 8.1 Medidas de complejidad
- 8.2 La clase P
- 8.3 La clase NP
- 8.4 NP-Compleitud
- 8.5 Algunos problemas NP-completos

Tema 9. Computación cuántica

- 9.1 El formalismo matemático de la Mecánica Cuántica
- 9.2 Descripción cuántica de los fenómenos físicos
- 9.3 Teoría cuántica de la información
- 9.4 Computación cuántica

- 9.5 Algoritmo de Deutsch-Jozsa
- 9.6 Algoritmo de búsqueda de Grover
- 9.7 Algoritmo de factorización de Shor

Temario práctico

1. Introducción a Haskell
2. Tipos y funciones básicas
3. Definición de tipos
4. Programación de funciones
5. Entrada/Salida
6. Testado de programas
7. Mónadas
8. Manejo de errores
9. Programación paralela y concurrente

7. Bibliografía

7.1 Bibliografía básica:

- M. Fernández. Models of Computation: An Introduction to Computability Theory. Undergraduate Topics in Computer Science. Springer (2009). ISBN 978-1-84882-433-1.
- J. E. Savage. Models Of Computation: Exploring the Power of Computing. (1998) (<http://cs.brown.edu/~jes/book/home.html>).
- M.D. Davis, R. Sigal, E.J. Weyujer. Computability, Complexity, and Languages (2nd. Ed.): Fundamentals of theoretical Computer Science. Academic Press (1994).
- M. Sipser. Introduction to the Theory of Computation (2nd Edition). Thompson (2005).
- M.A. Nielsen, I.L. Chuang, Quantum Computation and Quantum Information (10th Anniversary Edition), Cambridge University Press (2010)
- B. O'Sullivan, D. Stewart, J. Goerzen. Real World Haskell. O'Really Media (2009). (<http://book.realworldhaskell.org/read/index.html>)

7.2 Bibliografía complementaria:

- S. Arora, B. Barak. Computational Complexity: A Modern Approach. Cambridge University Press (2009)

- G. Ausiello, P. Creszendi et al. Complexity and Approximation. Springer-Verlag, Berlin (1999)
- R. Greenlaw, H.J. Hoover, W.L. Ruzzo. Limits to Parallel Computation: P-Completeness Theory (1995) Oxford University Press.
- J.E. Hopcroft, R. Motwani, J.D. Ullman. Introducción a la Teoría de Autómatas, Lenguajes y Programación, 2ª Ed. Addison Wesley (2002)

8. Sistemas y criterios de evaluación

8.1 Sistemas de evaluación:

- Examen de Teoría/Problemas
- Examen de Prácticas
- Defensa de Trabajos e Informes Escritos

8.2 Criterios de evaluación relativos a cada convocatoria:

8.2.1 Convocatoria I:

La asistencia tanto a las clases teóricas como prácticas no será obligatoria. La evaluación de la asignatura consta de las siguientes partes:

- Examen de teoría/problemas: 50%
- Examen de prácticas: 30%
- Defensa de trabajos e Informes Escritos: 20%

La evaluación de la asignatura consta de una parte teórica y una parte práctica.

La parte teórica (50% de la nota final) se evalúa por medio de las convocatorias oficiales de exámenes.

La parte práctica (50% de la nota final) se evalúa por medio de un examen de prácticas (30% de la nota final) y de un trabajo individual (20% de la nota final).

La calificación global final será la media de la calificación teórica final y la calificación práctica final, siendo necesaria una calificación mínima de 4.5 puntos (sobre 10.0) en el examen teórico y de 4.0 puntos (sobre 10.0) en la parte práctica.

Mediante la parte teórica se evaluarán las competencias CE1-C, CG0, G03, G09 y T02, mientras que por medio de la parte práctica se evaluarán las competencias CB5, G03, G04, G07 y G08.

8.2.2 Convocatoria II:

La asistencia tanto a las clases teóricas como prácticas no será obligatoria. La evaluación de la asignatura consta de las siguientes partes:

- Examen de teoría/problemas: 50%
- Examen de prácticas: 30%
- Defensa de trabajos e Informes Escritos: 20%

La evaluación de la asignatura consta de una parte teórica y una parte práctica.

La parte teórica (50% de la nota final) se evalúa por medio de las convocatorias oficiales de exámenes.

La parte práctica (50% de la nota final) se evalúa por medio de un examen de prácticas (30% de la nota final) y de un trabajo individual (20% de la nota final).

La calificación global final será la media de la calificación teórica final y la calificación práctica final, siendo necesaria una calificación mínima de 4.5 puntos (sobre 10.0) en el examen teórico y de 4.0 puntos (sobre 10.0) en la parte práctica. Mediante la parte teórica se evaluarán las competencias CE1-C, CG0, G03, G09 y T02, mientras que por medio de la parte práctica se evaluarán las competencias CB5, G03, G04, G07 y G08.

8.2.3 Convocatoria III:

La asistencia tanto a las clases teóricas como prácticas no será obligatoria. La evaluación de la asignatura consta de las siguientes partes:

- Examen de teoría/problemas: 50%
- Examen de prácticas: 30%
- Defensa de trabajos e Informes Escritos: 20%

La evaluación de la asignatura consta de una parte teórica y una parte práctica.

La parte teórica (50% de la nota final) se evalúa por medio de las convocatorias oficiales de exámenes.

La parte práctica (50% de la nota final) se evalúa por medio de un examen de prácticas (30% de la nota final) y de un trabajo individual (20% de la nota final).

La calificación global final será la media de la calificación teórica final y la calificación práctica final, siendo necesaria una calificación mínima de 4.5 puntos (sobre 10.0) en el examen teórico y de 4.0 puntos (sobre 10.0) en la parte práctica.

Mediante la parte teórica se evaluarán las competencias CE1-C, CG0, G03, G09 y T02, mientras que por medio de la parte práctica se evaluarán las competencias CB5, G03, G04, G07 y G08.

8.2.4 Convocatoria extraordinaria:

La asistencia tanto a las clases teóricas como prácticas no será obligatoria. La evaluación de la asignatura consta de las siguientes partes:

- Examen de teoría/problemas: 50%
- Examen de prácticas: 30%
- Defensa de trabajos e Informes Escritos: 20%

La evaluación de la asignatura consta de una parte teórica y una parte práctica.

La parte teórica (50% de la nota final) se evalúa por medio de las convocatorias oficiales de exámenes.

La parte práctica (50% de la nota final) se evalúa por medio de un examen de prácticas (30% de la nota final) y de un trabajo individual (20% de la nota final).

La calificación global final será la media de la calificación teórica final y la calificación práctica final, siendo necesaria una calificación mínima de 4.5 puntos (sobre 10.0) en el examen teórico y de 4.0 puntos (sobre 10.0) en la parte práctica.

Mediante la parte teórica se evaluarán las competencias CE1-C, CG0, G03, G09 y T02, mientras que por medio de la parte práctica se evaluarán las competencias CB5, G03, G04, G07 y G08.

8.3 Evaluación única final:

8.3.1 Convocatoria I:

La evaluación única final consistirá, para todas las convocatorias, en un solo acto académico que estará formado por las siguientes pruebas:

- Prueba 1: Examen escrito sobre los contenidos explicados en las sesiones de teoría y problemas. Tendrá un carácter presencial e individual, con una duración máxima de hasta 3 horas.
- Prueba 2: Examen práctico en el que se planteará un problema a resolver mediante programación funcional. El examen se desarrollará en un aula de ordenadores y su duración máxima será de 1 hora.

Para aprobar la asignatura se tienen que superar con más de un 5.0 independientemente ambas pruebas.

8.3.2 Convocatoria II:

La evaluación única final consistirá, para todas las convocatorias, en un solo acto académico que estará formado por las siguientes pruebas:

- Prueba 1: Examen escrito sobre los contenidos explicados en las sesiones de teoría y problemas. Tendrá un carácter presencial e individual, con una duración máxima de hasta 3 horas.
- Prueba 2: Examen práctico en el que se planteará un problema a resolver mediante programación funcional. El examen se desarrollará en un aula de ordenadores y su duración máxima será de 1 hora.

Para aprobar la asignatura se tienen que superar con más de un 5.0 independientemente ambas pruebas.

8.3.3 Convocatoria III:

La evaluación única final consistirá, para todas las convocatorias, en un solo acto académico que estará formado por las siguientes pruebas:

- Prueba 1: Examen escrito sobre los contenidos explicados en las sesiones de teoría y problemas. Tendrá un carácter presencial e individual, con una duración máxima de hasta 3 horas.
- Prueba 2: Examen práctico en el que se planteará un problema a resolver mediante programación funcional. El examen

se desarrollará en un aula de ordenadores y su duración máxima será de 1 hora.

Para aprobar la asignatura se tienen que superar con mas de un 5.0 independientemente ambas pruebas.

8.3.4 Convocatoria Extraordinaria:

La evaluación única final consistirá, para todas las convocatorias, en un solo acto académico que estará formado por las siguientes pruebas:

- Prueba 1: Examen escrito sobre los contenidos explicados en las sesiones de teoría y problemas. Tendrá un carácter presencial e individual, con una duración máxima de hasta 3 horas.
- Prueba 2: Examen práctico en el que se planteará un problema a resolver mediante programación funcional. El examen se desarrollará en un aula de ordenadores y su duración máxima será de 1 hora.

Para aprobar la asignatura se tienen que superar con mas de un 5.0 independientemente ambas pruebas.

Esta guía no incluye organización docente semanal orientativa