

Grado en Computer Science and Technology

Descripción del título			
Denominación	Grado en Computer Science and Technology		
Ciclo	Grado	Número de ECTS del título	240,00
Tipo de enseñanza	Presencial	Número de ECTS mínimos a tiempo completo	40
Rama de enseñanza	Ingeniería y Arquitectura	Número de ECTS mínimos a tiempo parcial	18
Universidades participantes (títulos conjuntos)			
Profesiones para las que capacita una vez obtenido el título (si			
Lenguas utilizadas a lo largo del proceso formativo (si procede)	Inglés		

Centros donde se imparte el título		Número de plazas de nuevo ingreso ofertada en			
ERT	Carácter	primer año	segundo año	tercer año	cuarto año
Beihang Valencia Polytechnic Institute	Propio				

3. Objetivos

Objetivos

El Grado en Computer Science and Technology tiene como finalidad dotar a los estudiantes de una formación técnica y científica sólida, que les permita desenvolverse con éxito en el ámbito de la tecnología informática. Este programa busca desarrollar competencias específicas y transversales que capaciten a los egresados para afrontar los desafíos del sector tecnológico actual y futuro.

El currículo está diseñado para proporcionar una comprensión profunda de los principios fundamentales de la informática, así como de las herramientas y técnicas necesarias para el diseño, desarrollo y gestión de sistemas informáticos. Los estudiantes adquirirán habilidades en áreas críticas como la programación, el análisis de datos, la inteligencia artificial y la computación en la nube, entre otras, lo que les permitirá abordar problemas complejos y proponer soluciones innovadoras.

Además, se fomenta una formación integral que incluye aspectos de humanidades, economía y derecho, lo que contribuye a formar profesionales con una visión global y ética, capaces de trabajar en entornos multiculturales y multidisciplinarios. La interacción con el sector industrial a través de prácticas y proyectos reales es un componente clave, asegurando que los estudiantes puedan aplicar sus conocimientos en situaciones prácticas y reales.

Los egresados estarán preparados para desempeñarse en diversas áreas, desde el desarrollo de software hasta la gestión de proyectos tecnológicos, con una sólida base que les permitirá adaptarse a la rápida evolución del sector. Así, el Grado en Computer Science and Technology no solo busca formar expertos en tecnología, sino también líderes que puedan contribuir de manera significativa a la sociedad y al avance de la innovación tecnológica.

Competencias generales y específicas

C1. (Habilidad o destreza)

Aplicar matemáticas discretas, cálculo, ecuaciones diferenciales, probabilidad y modelización matemática para formular y resolver problemas en contextos de computación y tecnológicos.

C10. (Competencia)

Evaluar la calidad de sistemas software y de información considerando mantenibilidad, seguridad, rendimiento y consistencia de datos.

C11. (Conocimiento o contenido)

Comprender los principios de las redes de computadores y los protocolos TCP/IP para analizar y configurar infraestructuras de comunicación.

C12. (Competencia)

Resolver problemas de redes, evaluando riesgos y vulnerabilidades y proponiendo medidas básicas de seguridad.

C13. (Habilidad o destreza)

Diseñar e implementar sistemas concurrentes y distribuidos seguros y eficientes.

C14. (Competencia)

Evaluar arquitecturas concurrentes y distribuidas considerando sincronización, escalabilidad, rendimiento y fiabilidad.

C15. (Conocimiento o contenido)

Comprender los principios de los compiladores y lenguajes de programación y su relación con la ejecución del software.

C16. (Habilidad o destreza)

Diseñar e implementar compiladores básicos, aplicando técnicas de optimización y analizando aspectos de seguridad.

C17. (Conocimiento o contenido)

Comprender los fundamentos teóricos y técnicos de la inteligencia artificial y sus principales paradigmas.

C18. (Competencia)

Diseñar e implementar soluciones basadas en IA, evaluando su rendimiento, limitaciones y riesgos.

C19. (Habilidad o destreza)

Aplicar técnicas de aprendizaje automático, deep learning y reconocimiento de patrones para resolver problemas reales.

C2. (Competencia)

Evaluar críticamente modelos matemáticos y físicos, valorando sus supuestos, limitaciones y validez en contextos ingenieriles.

C20. (Competencia)

Evaluar modelos de aprendizaje automático y sistemas inteligentes en contextos industriales y críticos.

C21. (Competencia)

Diseñar y analizar sistemas de computación inteligente y sistemas multiagente, justificando decisiones arquitectónicas

C22. (Habilidad o destreza)

Aplicar paradigmas de computación avanzada (como computación en la nube, de alto rendimiento y cuántica) así como conceptos básicos de tecnologías aeroespaciales para analizar y resolver problemas complejos en contextos tecnológicos emergentes.

C23. (Competencia)

Evaluar críticamente las limitaciones, oportunidades, tendencias y retos de las tecnologías de computación avanzada y del ámbito aeroespacial, valorando su impacto tecnológico, científico, ambiental y social, y seleccionando enfoques adecuados en función de criterios de rendimiento, escalabilidad, sostenibilidad y viabilidad técnica.

C24. (Habilidad o destreza)

Desarrollar aplicaciones móviles, visuales y seguras, evaluando arquitecturas y técnicas en función de usabilidad, rendimiento y seguridad.

C25. (Competencia)

Comunicar de forma eficaz información técnica y científica en contextos académicos y profesionales, utilizando correctamente el inglés y el español, y mostrando sensibilidad intercultural en entornos internacionales.

C3. (Habilidad o destreza)

Analizar fenómenos electromagnéticos, circuitos eléctricos y sistemas electrónicos, aplicando métodos experimentales y de laboratorio con rigor técnico.

C4. (Conocimiento o contenido)

Comprender la organización y arquitectura del computador y el funcionamiento de los sistemas operativos para analizar el rendimiento de aplicaciones.

C5. (Competencia)

Evaluar configuraciones de hardware y sistemas operativos justificando decisiones en términos de eficiencia, seguridad y uso de recursos.

C6. (Habilidad o destreza)

Aplicar principios de programación orientada a objetos, estructuras de datos y algoritmos para desarrollar software eficiente y mantenible.

C7. (Competencia)

Analizar y comparar algoritmos y estructuras de datos justificando decisiones en función de su complejidad y rendimiento.

C8. (Habilidad o destreza)

Integrar teoría de autómatas y computabilidad en el diseño de soluciones algorítmicas y sistemas software.

C9. (Habilidad o destreza)

Diseñar, desarrollar e integrar sistemas de información aplicando metodologías de ingeniería del software y bases de datos relacionales.

TFG. (Competencia)

Ejercicio original a realizar individualmente y presentar y defender ante un tribunal universitario, consistente en un proyecto en el ámbito de las tecnologías específicas del ámbito del título de naturaleza profesional en el que se sintetizan e integran las competencias adquiridas en las enseñanzas.

PLANIFICACIÓN DE LA ENSEÑANZA DEL TÍTULO: GRADO EN COMPUTER SCIENCE AND TECHNOLOGY

Planificación enseñanza

Se considera la equivalencia de 1 ECTS = 27,5 horas de dedicación (10 horas de interacción académica dirigida por el docente ponderadas por el porcentaje de presencialidad para modalidades no presenciales), correspondientes a las clases lectivas, teóricas, de seminario o prácticas, así como a la realización de las pruebas de evaluación que procedan; y 17,5 horas de actividad de trabajo autónomo, correspondientes a las horas de estudio autónomo, las dedicadas a la realización de trabajos (teóricos o prácticos) y/o proyectos, y las exigidas para la preparación de las pruebas de evaluación que procedan).

Distribución del plan de estudios en créditos ECTS por tipo de materia

Formación básica	65,00
Obligatorias	151,00
Optativas	0,00
Prácticas externas	0,00
Trabajo de fin de grado/máster	24,00
Total	240,00

Número máximo de créditos reconocibles por títulos propios: 0,00

Número máximo de créditos reconocibles por otros títulos oficiales no universitarios: 0,00

Número máximo de créditos reconocibles por acreditación de experiencia laboral y profesional: 0,00

Objetivos

El Grado en Computer Science and Technology tiene como finalidad dotar a los estudiantes de una formación técnica y científica sólida, que les permita desenvolverse con éxito en el ámbito de la tecnología informática. Este programa busca desarrollar competencias específicas y transversales que capaciten a los egresados para afrontar los desafíos del sector tecnológico actual y futuro.

El currículo está diseñado para proporcionar una comprensión profunda de los principios fundamentales de la informática, así como de las herramientas y técnicas necesarias para el diseño, desarrollo y gestión de sistemas informáticos. Los estudiantes adquirirán habilidades en áreas críticas como la programación, el análisis de datos, la inteligencia artificial y la computación en la nube, entre otras, lo que les permitirá abordar problemas complejos y proponer soluciones innovadoras.

Además, se fomenta una formación integral que incluye aspectos de humanidades, economía y derecho, lo que contribuye a formar profesionales con una visión global y ética, capaces de trabajar en entornos multiculturales y multidisciplinarios. La interacción con el sector industrial a través de prácticas y proyectos reales es un componente clave, asegurando que los estudiantes puedan aplicar sus conocimientos en situaciones prácticas y reales.

Los egresados estarán preparados para desempeñarse en diversas áreas, desde el desarrollo de software hasta la gestión de proyectos tecnológicos, con una sólida base que les permitirá adaptarse a la rápida evolución del sector. Así, el Grado en Computer Science and Technology no solo busca formar expertos en tecnología, sino también líderes que puedan contribuir de manera significativa a la sociedad y al avance de la innovación tecnológica.

Secuenciación de materias

Secuenciación de materias del Ert Beihang Valencia Polytechnic Institute

Curso	Primer Semestre	Segundo Semestre
Curso 1	Object-oriented software development (4 ECTS) Basics and techniques of programming	Computing and algorithmic development (6 ECTS) Basic training
	Fundamentals of mathematics (15 ECTS) Basic training	Introduction to artificial intelligence (4 ECTS) Artificial intelligence
	Language skills for engineering (5.5 ECTS) Complementary training	Mathematical methods and modeling (6 ECTS) Basic training
		Electronics (5 ECTS) Complementary training
		Language skills for engineering (5.5 ECTS) Complementary training
	Fundamentals of physics (9 ECTS) Complementary training	
Curso 2	Mathematical tools (4 ECTS) Complementary training	Language skills for engineering (2.5 ECTS) Complementary training
	Computer networks (4 ECTS) Computing systems and technologies	Concurrent and distributed systems (6 ECTS) Computing systems and technologies
	Architecture and operating systems (11 ECTS) Basic training	Computing and algorithmic development (12 ECTS) Basic training
	Language skills for engineering (2.5 ECTS) Complementary training	Architecture and operating systems (9 ECTS) Basic training
	Mathematical methods and modeling (6 ECTS) Basic training	Introduction to aerospace technologies (3 ECTS) Complementary training
Curso 3	Information systems development and management (12 ECTS) Computing systems and technologies	Application development technologies (8.5 ECTS) Basics and techniques of programming
	Mathematical tools (4.5 ECTS) Complementary training	Computer networks (3 ECTS) Computing systems and technologies
	Object-oriented software development (5.5 ECTS) Basics and techniques of programming	Advanced machine learning systems (4.5 ECTS) Artificial intelligence
	Advanced machine learning systems (2 ECTS) Artificial intelligence	Optimization (4.5 ECTS) Basics and techniques of programming
		Intelligent systems (4.5 ECTS) Computing systems and technologies
		Mathematical tools (2 ECTS) Complementary training
Curso 4	Application development technologies (4.5 ECTS) Basics and techniques of programming	Final degree project (24 ECTS) Final degree project
	Mathematical tools (0 ECTS) Complementary training	Advanced computing technologies (4.5 ECTS) Computing systems and technologies
	Intelligent systems (4 ECTS) Computing systems and technologies	
	Optimization (4 ECTS) Basics and techniques of programming	
	Advanced computing technologies (4.5 ECTS) Computing systems and technologies	
	Advanced machine learning systems (6.5 ECTS) Artificial intelligence	
	Object-oriented software development (1.5 ECTS) Basics and techniques of	

	programming	
Curso 4	Introduction to artificial intelligence (4.5 ECTS) Artificial intelligence	

Descripción de los módulos

Denominación	Créditos ECTS
Basic training	65,00
Complementary training	45,50
Artificial intelligence	21,50
Basics and techniques of programming	41,50
Computing systems and technologies	42,50
Final degree project	24,00

Basic training

Descripción

-

Complementary training

Descripción

-

Artificial intelligence

Descripción

Basics and techniques of programming

Descripción

-

Computing systems and technologies

Descripción

Final degree project

Descripción

Descripción de las materias

Denominación	Créditos ECTS	Carácter	Unidad temporal
Fundamentals of mathematics	15	Formación Básica	Semestres 1
Mathematical methods and modeling	12	Formación Básica	Semestres 2 y 3
Computing and algorithmic development	18	Formación Básica	Semestres 2 y 4
Architecture and operating systems	20	Formación Básica	Semestres 3 y 4
Language skills for engineering	16	Obligatorio	Semestres 1, 2, 3 y 4
Fundamentals of physics	9	Obligatorio	Semestres 1 y 2
Electronics	5	Obligatorio	Semestres 2
Introduction to aerospace technologies	3	Obligatorio	Semestres 4
Mathematical tools	12.5	Obligatorio	Semestres 3, 5, 6 y 7
Introduction to artificial intelligence	8.5	Obligatorio	Semestres 2 y 7
Advanced machine learning systems	13	Obligatorio	Semestres 5, 6 y 7
Object-oriented software development	11	Obligatorio	Semestres 1, 5 y 7
Compilers	9	Obligatorio	Semestres 5
Application development technologies	13	Optativo	Semestres 6 y 7
Optimization	8.5	Obligatorio	Semestres 6 y 7
Concurrent and distributed systems	6	Obligatorio	Semestres 4
Computer networks	7	Obligatorio	Semestres 3 y 6
Information systems development and management	12	Obligatorio	Semestres 5
Intelligent systems	8.5	Obligatorio	Semestres 6 y 7
Advanced computing technologies	9	Obligatorio	Semestres 7 y 8
Final degree project	24	Trabajo Fin Titulación	Semestres 8

Fundamentals of mathematics

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
CT2	(2)	Proponer soluciones creativas e innovadoras a situaciones o problemas complejos, propios del ámbito de conocimiento, para dar respuesta a las diversas necesidades profesionales y sociales.
C1	(3)	Aplicar matemáticas discretas, cálculo, ecuaciones diferenciales, probabilidad y modelización matemática para formular y resolver problemas en contextos de computación y tecnológicos.
C10	(2)	Evaluar la calidad de sistemas software y de información considerando mantenibilidad, seguridad, rendimiento y consistencia de datos.
C2	(2)	Evaluar críticamente modelos matemáticos y físicos, valorando sus supuestos, limitaciones y validez en contextos ingenieriles.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
FM1	Identificar y describir conceptos fundamentales de matemáticas discretas y cálculo.
FM2	Analizar conceptualmente las relaciones entre funciones matemáticas, ecuaciones diferenciales y su aplicación en la resolución de problemas complejos en el ámbito de la informática.
FM3	Formular y resolver problemas matemáticos propios de la ingeniería utilizando herramientas analíticas, algebraicas y discretas, seleccionando los métodos adecuados en cada caso.
FM4	Modelar situaciones reales utilizando ecuaciones diferenciales e implementando soluciones numéricas y analíticas para optimizar procesos en ingeniería informática.
FM5	Evaluar críticamente la adecuación de modelos y soluciones matemáticas, considerando sus limitaciones, supuestos y validez en el contexto del problema ingenieril.
FM6	Comparar críticamente diversas técnicas de modelado matemático y su aplicabilidad en contextos de ingeniería, integrando conocimientos de matemáticas discretas y cálculo avanzado.
RADM_1	Aplicar conceptos de lógica y conjuntos en la resolución de problemas matemáticos en contextos de programación y desarrollo de software con precisión y rigor.
RADM_2	Analizar relaciones y funciones en estructuras discretas utilizando herramientas matemáticas adecuadas en entornos académicos y prácticos.
RADM_3	Evaluar y comparar diferentes algoritmos de búsqueda y ordenación en función de su eficiencia y complejidad computacional en ejercicios prácticos.
RADM_4	Diseñar y representar grafos para modelar problemas reales en informática, considerando criterios de optimización y claridad en la presentación.
RAMI_1	Identificar y clasificar funciones de una variable, incluyendo funciones hiperbólicas y números complejos, en contextos de resolución de problemas matemáticos en ingeniería.
RAMI_3	Analizar y resolver sistemas de ecuaciones lineales utilizando métodos de álgebra lineal, evaluando la viabilidad de las soluciones en contextos de ingeniería.
RAMI_4	Interpretar y formular modelos matemáticos aplicados a situaciones de ingeniería, validando los resultados mediante habilidades computacionales en entornos simulados.

Resumen de contenidos

Proporciona una formación sólida en razonamiento lógico, estructuras matemáticas y herramientas analíticas necesarias para el análisis y diseño de sistemas computacionales. Incluye contenidos de matemáticas discretas, cálculo diferencial e integral, álgebra lineal, probabilidad y modelización matemática, todos ellos imprescindibles para comprender la naturaleza formal y cuantitativa de los sistemas computacionales. La materia favorece el desarrollo de habilidades de

abstracción, rigor y pensamiento crítico, así como la capacidad para formular modelos que representen sistemas complejos y permitan su análisis computacional. Su orientación es tanto teórica como aplicada, potenciando la conexión entre los conceptos matemáticos y su uso en algoritmos, estructuras de datos, optimización, inteligencia artificial o análisis de datos.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,75	30,00	80,00
Práctica Campo	0,00		
Práctica Informática	3,75	30,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	7,50	60,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		262,50	0,00
TOTAL	15,00	382,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Trabajos académicos	10%	30%
Prueba escrita	30%	70%
Prueba práctica de laboratorio/campo/informática/aula	30%	50%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Mathematical methods and modeling

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
CT2	(2)	Proponer soluciones creativas e innovadoras a situaciones o problemas complejos, propios del ámbito de conocimiento, para dar respuesta a las diversas necesidades profesionales y sociales.
C1	(3)	Aplicar matemáticas discretas, cálculo, ecuaciones diferenciales, probabilidad y modelización matemática para formular y resolver problemas en contextos de computación y tecnológicos.
C2	(2)	Evaluar críticamente modelos matemáticos y físicos, valorando sus supuestos, limitaciones y validez en contextos ingenieriles.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
MM1	Identificar y describir los conceptos fundamentales de métodos y teorías matemáticas avanzadas en su aplicación a problemas de ingeniería.
MM2	Explicar los principios y fundamentos probabilísticos, para la modelización matemática de fenómenos inciertos.
MM5	Evaluar la adecuación de distintos modelos matemáticos para representar fenómenos reales, valorando limitaciones, supuestos y validez.
MM6	Seleccionar y justificar las técnicas matemáticas más apropiadas para abordar un problema de ingeniería, integrando criterios de eficiencia, precisión y rigor conceptual.
RAMII_1	Aplicar métodos de resolución de ecuaciones diferenciales de primer y segundo orden en problemas de ingeniería utilizando ejemplos prácticos y con precisión en los cálculos.
RAMII_2	Analizar series de Fourier y su aplicación en la representación de funciones periódicas en contextos de ingeniería, considerando criterios de convergencia y propiedades.
RAMII_3	Resolver sistemas de ecuaciones diferenciales lineales utilizando la Transformada de Laplace, evaluando la efectividad de los métodos en situaciones reales de ingeniería.
RAMII_4	Implementar métodos numéricos para la aproximación de soluciones a ecuaciones diferenciales parciales en entornos simulados, garantizando la validez de los resultados obtenidos.
RAPM_1	Aplicar conceptos de probabilidad para resolver problemas de incertidumbre en contextos reales utilizando herramientas estadísticas adecuadas.
RAPM_2	Modelar fenómenos aleatorios mediante técnicas de simulación en entornos computacionales, evaluando la validez de los modelos generados.
RAPM_3	Analizar datos empíricos utilizando métodos de inferencia estadística, interpretando los resultados con rigor y claridad.
RAPM_4	Diseñar modelos matemáticos que representen situaciones prácticas en ingeniería, considerando criterios de precisión y aplicabilidad.

Resumen de contenidos

Esta materia profundiza en las herramientas matemáticas avanzadas necesarias para formular, analizar y resolver problemas complejos en el análisis y diseño de sistemas computacionales. Complementa la formación matemática básica mediante el estudio de ecuaciones diferenciales ordinarias y parciales, transformadas integrales, métodos numéricos y técnicas de modelización basadas en la probabilidad y la estadística. La materia introduce al estudiante en el uso de las series de Fourier, la teoría fundamental de las ecuaciones diferenciales y sus métodos de resolución, así como en la interpretación y aplicación de sistemas dinámicos lineales. Se abordan

también la transformada de Laplace y la integración de métodos numéricos para aproximar soluciones cuando no existen métodos analíticos directos. Por otro lado, la materia también capacita al estudiante para representar fenómenos inciertos, construir modelos probabilísticos y analizar su comportamiento mediante herramientas estadísticas y analíticas. En conjunto, esta materia proporciona los fundamentos necesarios para abordar problemas reales de la ingeniería mediante modelos matemáticos robustos, preparando al estudiante para asignaturas avanzadas relacionadas con la inteligencia artificial o la optimización de sistemas.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,00	24,00	80,00
Práctica Campo	0,00		
Práctica Informática	3,00	24,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	6,00	48,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		210,00	0,00
TOTAL	12,00	306,00	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Trabajos académicos	10%	30%
Prueba escrita	30%	70%
Prueba práctica de laboratorio/campo/informática/aula	30%	50%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Computing and algorithmic development

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C6	(3)	Aplicar principios de programación orientada a objetos, estructuras de datos y algoritmos para desarrollar software eficiente y mantenible.
C7	(2)	Analizar y comparar algoritmos y estructuras de datos justificando decisiones en función de su complejidad y rendimiento.
C8	(3)	Integrar teoría de autómatas y computabilidad en el diseño de soluciones algorítmicas y sistemas software.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
CAD1	Identificar y describir los principios fundamentales de la teoría de autómatas y la computabilidad, integrando conceptos de programación.
CAD2	Explicar las estructuras de datos y algoritmos relevantes para evaluar su impacto en la eficiencia de aplicaciones informáticas.
CAD4	Modelar y diseñar sistemas informáticos utilizando principios de algorítmica, integrando conocimientos de programación.
CAD5	Evaluar críticamente diferentes algoritmos y estructuras de datos, integrando conocimientos de programación y teoría de autómatas para justificar decisiones en el desarrollo de software.
CAD6	Comparar críticamente la eficiencia de algoritmos y estructuras de datos para tomar decisiones fundamentadas en el desarrollo de aplicaciones.
RAAT_1	Identificar y clasificar diferentes tipos de autómatas y lenguajes formales en contextos teóricos y prácticos de la computación.
RAAT_2	Aplicar técnicas de análisis de lenguajes formales para resolver problemas de computabilidad en entornos simulados.
RAAT_3	Evaluar la complejidad de algoritmos relacionados con autómatas y gramáticas formales utilizando criterios de eficiencia y precisión.
RAAT_4	Diseñar y desarrollar modelos de autómatas que representen problemas computacionales específicos, considerando criterios de claridad y funcionalidad.
RADE_1	Seleccionar estructuras de datos adecuadas para la resolución de problemas de ingeniería considerando criterios de eficiencia y complejidad computacional en contextos prácticos.
RADE_2	Implementar algoritmos básicos de procesamiento de datos en lenguajes de programación, evaluando su rendimiento y adecuación a problemas específicos en entornos simulados.
RADE_3	Analizar la complejidad temporal y espacial de diferentes algoritmos, justificando su elección en función de las características del problema a resolver.

Resumen de contenidos

Esta materia proporciona al estudiante una base sólida sobre los principios que sustentan la computación, combinando los aspectos teóricos del cómputo con los fundamentos prácticos de la programación y el diseño algorítmico. Integra el estudio de los modelos formales de cálculo, la computabilidad y las limitaciones inherentes a la resolución de problemas, junto con el aprendizaje de las técnicas esenciales para desarrollar programas correctos, estructurados y eficientes. La materia abarca el análisis y diseño de algoritmos, la selección y uso adecuado de

estructuras de datos, y la comprensión de los modelos de autómatas y las funciones computables que definen el alcance y los límites del procesamiento de la información. Asimismo, introduce al estudiante en el razonamiento computacional y en las metodologías necesarias para resolver problemas mediante la programación, aplicando técnicas de abstracción, modularidad y evaluación de la eficiencia.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	4,50	36,00	80,00
Práctica Campo	0,00		
Práctica Informática	6,00	48,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	7,50	60,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		315,00	0,00
TOTAL	18,00	459,00	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	40%	70%
Prueba práctica de laboratorio/campo/informática/aula	40%	70%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Architecture and operating systems

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C4	(1)	Comprender la organización y arquitectura del computador y el funcionamiento de los sistemas operativos para analizar el rendimiento de aplicaciones.
C5	(2)	Evaluar configuraciones de hardware y sistemas operativos justificando decisiones en términos de eficiencia, seguridad y uso de recursos.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
AOS2	Explicar las funciones de los sistemas operativos y su impacto en la eficiencia de aplicaciones informáticas.
AOS3	Analizar el comportamiento de programas en relación con la arquitectura del computador, identificando cómo afectan las decisiones de hardware al rendimiento y ejecución.
AOS4	Utilizar herramientas, comandos y entornos de sistemas operativos para gestionar procesos, memoria y archivos, así como para diagnosticar problemas básicos de funcionamiento.
AOS5	Evaluar alternativas arquitectónicas o de configuración del sistema operativo justificando su impacto en rendimiento, eficiencia y seguridad.
RACO_1	Identificar los componentes de un sistema de hardware informático y su funcionamiento en contextos de diseño y análisis de sistemas computacionales.
RACO_2	Analizar arquitecturas contemporáneas de sistemas informáticos utilizando principios universales y casos típicos en entornos académicos y prácticos.
RACO_4	Desarrollar soluciones de organización de sistemas informáticos en situaciones simuladas, evaluando su rendimiento y viabilidad técnica con rigor.
RAOS_1	Identificar los conceptos fundamentales de los sistemas operativos y su relación con la gestión de procesos y almacenamiento en entornos simulados.
RAOS_2	Aplicar algoritmos y estructuras de datos para la gestión de entradas/salidas y dispositivos en la implementación de un sistema operativo básico.
RAOS_3	Analizar las técnicas de diseño e implementación de las principales partes de sistemas operativos modernos en contextos de desarrollo de software.

Resumen de contenidos

Esta materia proporciona una visión integral del funcionamiento interno de los computadores, abordando tanto su estructura física como los mecanismos software que gestionan sus recursos. Combina el estudio de la organización y arquitectura del computador con el análisis de los fundamentos y servicios que ofrece un sistema operativo moderno. El estudiante adquiere las competencias necesarias para comprender cómo se ejecutan los programas, cómo se gestionan procesos, memoria y dispositivos, y cómo interactúan el hardware y el software a través de interfaces bien definidas. La materia también capacita al estudiante para analizar el rendimiento de los sistemas, diagnosticar problemas asociados al hardware o al sistema operativo.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	5,00	40,00	80,00
Práctica Campo	0,00		

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Informática	6,00	48,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	9,00	72,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		350,00	0,00
TOTAL	20,00	510,00	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	40%	70%
Prueba práctica de laboratorio/campo/informática/aula	40%	70%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Language skills for engineering

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
CT4	(2)	Comunicarse de manera efectiva, tanto de forma oral como escrita, adaptándose a las características de la situación y de la audiencia.
C25	(2)	Comunicar de forma eficaz información técnica y científica en contextos académicos y profesionales, utilizando correctamente el inglés y el español, y mostrando sensibilidad intercultural en entornos internacionales.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
LS4	Implementar estrategias de aprendizaje autónomo para mejorar la competencia lingüística en inglés y español, integrando recursos de las asignaturas de lectura y escritura.
LS5	Evaluar críticamente la calidad de textos técnicos en inglés, justificando decisiones sobre su adecuación en contextos de ingeniería, a partir de conocimientos adquiridos en las asignaturas.
LS6	Comparar críticamente las metodologías de enseñanza de idiomas aplicadas en las asignaturas, tomando decisiones fundamentadas sobre su efectividad en el aprendizaje de competencias lingüísticas.
RACE_1	Identificar y describir los principales conceptos y estructuras gramaticales del chino y el español en contextos de comunicación cotidiana.
RACE_2	Aplicar técnicas de traducción básica entre chino y español en situaciones comunicativas reales, utilizando recursos digitales y diccionarios especializados.
RACE_3	Analizar textos sencillos en chino y español, evaluando su contenido y estructura para facilitar la comprensión intercultural.
RACE_4	Comunicar de forma efectiva en chino y español, utilizando vocabulario adecuado y considerando aspectos culturales relevantes en presentaciones orales y escritas.
RAER_1	Identificar y clasificar términos técnicos en inglés relacionados con la informática y la tecnología en contextos académicos y profesionales.
RAER_2	Analizar textos en inglés sobre temas de informática, extrayendo ideas principales y argumentos relevantes en un entorno de discusión grupal.
RAER_3	Aplicar estrategias de lectura crítica en inglés para evaluar la validez y la relevancia de información técnica en documentos especializados.
RAER_4	Comunicar de manera efectiva en inglés los resultados de análisis de textos técnicos, utilizando terminología adecuada y estructuras coherentes en presentaciones orales.
RAES_2	Expresar ideas y argumentos sobre temas de tecnología de la información en inglés, utilizando estructuras gramaticales correctas y un léxico apropiado.
RAES_3	Participar en discusiones grupales sobre proyectos tecnológicos, demostrando habilidades de comunicación oral y escucha activa en inglés.
RAES_4	Elaborar presentaciones orales sobre conceptos de informática, utilizando recursos visuales y un lenguaje técnico claro y coherente para audiencias especializadas.
RAEW_1	Redactar textos técnicos en inglés utilizando terminología adecuada y siguiendo las convenciones del estilo académico en contextos de ingeniería y tecnología.

Resumen de contenidos

La materia de Competencias Lingüísticas para la Ingeniería tiene como objetivo desarrollar habilidades comunicativas avanzadas en inglés, español y chino, fundamentales en un entorno profesional globalizado. Su diseño curricular integra la lectura, la escritura y la expresión oral en inglés, junto con el aprendizaje de español y chino, lo que permite al estudiante adquirir vocabulario técnico y comprender textos complejos propios relacionados con las tecnologías computacionales. Además, fomenta el análisis crítico de diferencias culturales y lingüísticas, así como la adaptación de la comunicación a distintos contextos profesionales. Los estudiantes aprenden a evaluar la calidad de textos técnicos y a justificar decisiones comunicativas, fortaleciendo su capacidad para participar en equipos internacionales y presentar proyectos tecnológicos. La materia también impulsa el aprendizaje autónomo, promoviendo la reflexión crítica y la autoevaluación. Esto permite al alumnado tomar decisiones fundamentadas sobre sus estrategias de aprendizaje y mantener una formación lingüística continua en un sector en constante evolución.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	5,50	44,00	80,00
Práctica Campo	0,00		
Práctica Informática	3,50	28,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	7,00	56,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		280,00	0,00
TOTAL	16,00	408,00	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Examen/defensa oral	30%	60%
Trabajos académicos	10%	30%
Prueba escrita	40%	60%
Prueba práctica de laboratorio/campo/informática/aula	10%	30%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Fundamentals of physics

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C2	(2)	Evaluar críticamente modelos matemáticos y físicos, valorando sus supuestos, limitaciones y validez en contextos ingenieriles.
C3	(3)	Analizar fenómenos electromagnéticos, circuitos eléctricos y sistemas electrónicos, aplicando métodos experimentales y de laboratorio con rigor técnico.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
FPH1	Explicar los conceptos básicos de campos y ondas y electromagnetismo y teoría de circuitos eléctricos.
FPH2	Identificar y describir los principios físicos de los semiconductores y familias lógicas, dispositivos electrónicos y fotónicos.
FPH3	Aplicar las leyes de Kirchhoff y otros métodos para resolver redes eléctricas lineales en problemas propios de la ingeniería.
FPH4	Utilizar los aparatos de medida en circuitos eléctricos aplicando la metodología propia del laboratorio de electricidad.
FPH5	Evaluar la validez de modelos y resultados experimentales en electricidad y electrónica, valorando posibles fuentes de error y su impacto en las conclusiones.
FPH6	Seleccionar y justificar métodos, herramientas y procedimientos adecuados para analizar fenómenos electromagnéticos y circuitos en contextos ingenieriles.

Resumen de contenidos

Esta materia proporciona al estudiante los fundamentos físicos esenciales para comprender el comportamiento de los sistemas electromagnéticos y electrónicos que sustentan la tecnología moderna, y le permite adquirir una base conceptual para interpretar y analizar fenómenos físicos directamente vinculados al diseño y funcionamiento de dispositivos y sistemas utilizados en ingeniería informática. La materia aborda el estudio de los campos y ondas, el electromagnetismo, la teoría de circuitos, los principios físicos de los semiconductores, las familias lógicas y los dispositivos electrónicos y fotónicos. Todo ello proporciona al estudiante una visión clara de los elementos que constituyen la base del hardware computacional.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,00	24,00	80,00
Práctica Campo	0,00		
Práctica Informática	3,00	24,00	80,00
Práctica Laboratorio	0,00		

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Teoría Aula	3,00	24,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		157,50	0,00
TOTAL	9,00	229,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	50%	80%
Prueba práctica de laboratorio/campo/informática/aula	20%	50%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Electronics

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C2	(2)	Evaluar críticamente modelos matemáticos y físicos, valorando sus supuestos, limitaciones y validez en contextos ingenieriles.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
ELE1	Identificar y describir los principios fundamentales de diseño de sistemas electrónicos, integrando conocimientos de circuitos en contextos prácticos.
ELE2	Explicar y analizar conceptualmente los parámetros y características de los sistemas de información electrónica, aplicando conocimientos de matemáticas y física en la evaluación de su rendimiento.
ELE3	Diseñar e implementar simulaciones y experimentos para resolver problemas complejos en sistemas electrónicos, integrando herramientas de software en proyectos prácticos.
ELE4	Aplicar software profesional resolver problemas de diseño de sistemas de información electrónica, comprendiendo los principios y limitaciones inherentes a las herramientas de simulación.
ELE5	Evaluar críticamente el rendimiento de sistemas electrónicos, justificando decisiones de diseño basadas en análisis de datos y resultados de simulaciones.
ELE6	Comparar críticamente diferentes enfoques de diseño en sistemas electrónicos, integrando conocimientos adquiridos para tomar decisiones fundamentadas en proyectos de ingeniería.

Resumen de contenidos

Esta materia introduce al estudiante en los principios fundamentales del diseño electrónico, combinando la enseñanza teórica con la práctica experimental. Proporciona una visión general de los componentes básicos de los sistemas electrónicos, así como de su funcionamiento, sus características y sus aplicaciones en el contexto de la ingeniería. También profundiza en la comprensión del comportamiento de los dispositivos electrónicos y en la estructura y funcionamiento de los sistemas de información electrónica. El enfoque es eminentemente práctico, con actividades orientadas al diseño asistido por ordenador, la simulación de circuitos y el desarrollo de prototipos electrónicos simples, tales como sistemas de iluminación activados por sonido, contadores, control de displays digitales o mediciones por ultrasonidos. Estas actividades facilitan el desarrollo progresivo de competencias en diseño, fabricación y depuración de circuitos.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	1,50	12,00	80,00
Práctica Campo	0,00		
Práctica Informática	0,00		
Práctica Laboratorio	2,00	16,00	80,00
Teoría Aula	1,50	12,00	80,00

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Teoría Seminario	0,00		
Trabajo autónomo		87,50	0,00
TOTAL	5,00	127,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Proyecto	50%	75%
Prueba escrita	25%	50%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Introduction to aerospace technologies

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C23	(2)	Evaluar críticamente las limitaciones, oportunidades, tendencias y retos de las tecnologías de computación avanzada y del ámbito aeroespacial, valorando su impacto tecnológico, científico, ambiental y social, y seleccionando enfoques adecuados en función de criterios de rendimiento, escalabilidad, sostenibilidad y viabilidad técnica.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
IAT1	Identificar y describir los conceptos fundamentales, principios físicos y terminología básica de la aeronáutica y la astronáutica.
IAT2	Describir de forma clara los componentes fundamentales de una aeronave o sistema espacial y explicar su función en el conjunto del sistema.
IAT3	Aplicar conceptos fundamentales de aerodinámica o mecánica orbital para analizar, de forma simplificada, situaciones típicas del vuelo atmosférico o espacial.
IAT4	Utilizar herramientas digitales básicas para la visualización, simulación o consulta de datos sobre aeronaves y misiones espaciales, interpretando sus resultados a nivel introductorio.
IAT5	Evaluar la relevancia y el impacto de las tecnologías aeroespaciales en la sociedad y en la ingeniería moderna, considerando aspectos científicos y tecnológicos.
IAT6	Analizar críticamente las limitaciones, retos y oportunidades del desarrollo aeroespacial actual, valorando sus implicaciones éticas, ambientales y tecnológicas.
RAIA_2	Aplicar conceptos de propulsión y dinámica de vuelo en la evaluación de sistemas aeronáuticos utilizando herramientas de modelado y simulación.
RAIA_3	Analizar los componentes principales de un sistema espacial y su funcionamiento en condiciones de microgravedad mediante estudios de caso y simulaciones.

Resumen de contenidos

Esta materia introduce al estudiante en los principios fundamentales, la terminología básica y el desarrollo histórico y tecnológico de la aeronáutica y la astronáutica. Su objetivo es proporcionar una comprensión general de los conceptos esenciales que sustentan el diseño, operación y evolución de aeronaves y vehículos espaciales, así como de las tecnologías que los hacen posibles. El contenido abarca desde los fundamentos físicos del vuelo, los sistemas que integran una aeronave, los principios de propulsión y navegación, hasta los conceptos básicos de ingeniería espacial, órbitas y misiones. Además, se presenta una visión actual del sector aeroespacial, sus avances más relevantes, los desafíos tecnológicos y su creciente impacto en múltiples ámbitos de la ingeniería. La materia está orientada también a despertar el interés del estudiante por el campo aeroespacial y a establecer conexiones entre estos conocimientos y su propia área de estudio, facilitando la integración de conceptos aeronáuticos y astronáuticos en futuros proyectos académicos y profesionales.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	1,00	8,00	80,00

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Campo	0,00		
Práctica Informática	1,00	8,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	1,00	8,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		52,50	0,00
TOTAL	3,00	76,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	40%	70%
Prueba práctica de laboratorio/campo/informática/aula	30%	60%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Mathematical tools

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C1	(3)	Aplicar matemáticas discretas, cálculo, ecuaciones diferenciales, probabilidad y modelización matemática para formular y resolver problemas en contextos de computación y tecnológicos.
C2	(2)	Evaluar críticamente modelos matemáticos y físicos, valorando sus supuestos, limitaciones y validez en contextos ingenieriles.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
MTT1	Identificar y describir los principios fundamentales de análisis matricial y modelado matemático, integrando conceptos de optimización y análisis de datos en contextos de ingeniería informática.
MTT2	Explicar y caracterizar las técnicas de análisis de datos y diseño de experimentos, integrando métodos de optimización y modelado matemático para resolver problemas prácticos en ingeniería.
MTT4	Diseñar y simular experimentos utilizando técnicas de diseño de experimentos y análisis de regresión, integrando conocimientos de optimización y análisis de datos para mejorar procesos.
MTT5	Evaluar críticamente los resultados obtenidos de modelos matemáticos y análisis de datos, justificando decisiones basadas en la integración de métodos de optimización y análisis estadístico.
MTT6	Comparar críticamente diferentes enfoques de modelado matemático y análisis de datos, tomando decisiones fundamentadas sobre la metodología más adecuada para resolver problemas específicos en ingeniería.
RADC_1	Aplicar herramientas estadísticas para monitorizar y mejorar la calidad de productos en ingeniería informática utilizando software estadístico en casos prácticos.
RADC_2	Diseñar experimentos (DOE) para optimizar el rendimiento de procesos en ingeniería informática, considerando variables relevantes y condiciones experimentales adecuadas.
RADC_3	Analizar modelos de regresión para establecer relaciones entre variables en sistemas de ingeniería informática, utilizando datos reales y software especializado.
RAMA_2	Aplicar técnicas de análisis matricial para resolver sistemas de ecuaciones lineales en entornos simulados y reales.
RAMA_3	Evaluar métodos de optimización en función de su eficiencia y aplicabilidad a problemas concretos en ingeniería.
RAMA_4	Diseñar algoritmos de optimización utilizando herramientas computacionales, considerando criterios de precisión y eficiencia en su implementación.
RAMM_1	Establecer modelos matemáticos para describir problemas del mundo real utilizando lenguaje matemático y considerando las limitaciones y supuestos del contexto específico.
RAMM_4	Trabajar en equipo para comunicar y presentar soluciones a problemas de modelado matemático, utilizando terminología técnica adecuada y recursos visuales efectivos.

Resumen de contenidos

Materia orientada al estudio y aplicación de técnicas matemáticas avanzadas necesarias en áreas como la optimización, el modelado matemático y el análisis de datos. Se profundiza en conceptos

de análisis matricial, cálculo avanzado, ecuaciones, estadística computacional y métodos numéricos que permitan resolver problemas reales en ingeniería informática mediante herramientas formales. Se presta especial atención a la formulación de modelos matemáticos que representen fenómenos físicos, industriales o computacionales, y al uso de técnicas de optimización para la toma de decisiones óptimas en contextos complejos. Además, la materia introduce métodos avanzados de análisis de datos, fundamentales en campos como la inteligencia artificial, el machine learning, la simulación o el análisis de grandes volúmenes de información, tratados en otras materias del grado.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,50	28,00	80,00
Práctica Campo	0,00		
Práctica Informática	4,50	36,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	4,50	36,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		218,75	0,00
TOTAL	12,50	318,75	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	40%	70%
Prueba práctica de laboratorio/campo/informática/aula	20%	50%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Introduction to artificial intelligence

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C17	(1)	Comprender los fundamentos teóricos y técnicos de la inteligencia artificial y sus principales paradigmas.
C18	(2)	Diseñar e implementar soluciones basadas en IA, evaluando su rendimiento, limitaciones y riesgos.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
IAI1	Identificar y describir los fundamentos teóricos de la inteligencia artificial, sus modelos de razonamiento y su evolución histórica y tecnológica.
IAI2	Explicar y caracterizar las principales técnicas de IA, incluyendo métodos de búsqueda, representación del conocimiento, inferencia y toma de decisiones.
IAI3	Aplicar técnicas de inteligencia artificial para resolver problemas reales, seleccionando métodos adecuados según la naturaleza del dominio.
IAI5	Analizar críticamente las limitaciones, ventajas y riesgos de las técnicas de IA aplicadas a diferentes escenarios tecnológicos.
RAIN_2	Aplicar métodos básicos de inteligencia artificial para resolver problemas simples utilizando herramientas y lenguajes de programación adecuados.
RAIN_3	Analizar las tendencias actuales en inteligencia artificial y su impacto en diferentes sectores, considerando aspectos éticos y sociales.
RAIN_4	Comunicar de manera efectiva los resultados de investigaciones sobre inteligencia artificial a audiencias técnicas y no técnicas, utilizando terminología precisa.
RAIT_1	Identificar problemas complejos en contextos reales utilizando técnicas de inteligencia artificial y representando sus conocimientos sobre restricciones, objetivos y métricas adecuadas.
RAIT_2	Discriminar entre diferentes técnicas de inteligencia artificial para recomendar el modelo y la técnica de resolución más apropiados en función de las características del problema.
RAIT_3	Aplicar técnicas de inferencia basada en reglas y búsqueda heurística para resolver problemas de optimización en entornos simulados con precisión y rigor profesional.
RAIT_4	Evaluar el rendimiento de las técnicas de inteligencia artificial aplicadas a problemas reales, considerando criterios de efectividad y eficiencia en la solución propuesta.

Resumen de contenidos

Materia dedicada a la comprensión de los fundamentos, técnicas y aplicaciones de la inteligencia artificial. Proporciona una visión general de los modelos de razonamiento, representación del conocimiento, resolución de problemas, búsqueda y toma de decisiones, así como de los algoritmos y metodologías clásicas de la IA. La materia también aborda el uso de técnicas de inteligencia artificial en ámbitos prácticos, incluyendo sistemas basados en reglas, planificación, agentes inteligentes, procesamiento de información y automatización de tareas. El estudiante adquiere competencias para comprender, diseñar e implementar soluciones basadas en IA, analizando sus posibilidades y limitaciones en contextos reales del sector tecnológico.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	2,50	20,00	80,00
Práctica Campo	0,00		
Práctica Informática	3,00	24,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	3,00	24,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		148,75	0,00
TOTAL	8,50	216,75	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Proyecto	40%	70%
Prueba escrita	40%	70%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Advanced machine learning systems

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C19	(3)	Aplicar técnicas de aprendizaje automático, deep learning y reconocimiento de patrones para resolver problemas reales.
C20	(2)	Evaluar modelos de aprendizaje automático y sistemas inteligentes en contextos industriales y críticos.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
AMLS1	Describir los principios fundamentales del aprendizaje automático, incluidos los modelos lineales, no paramétricos y los conceptos matemáticos que los sustentan.
AMLS2	Identificar las arquitecturas de deep learning más utilizadas en la industria y explicar su funcionamiento, fortalezas y limitaciones según el tipo de dato y aplicación.
AMLS3	Identificar y describir los fundamentos teóricos de reconocimiento de patrones, incluyendo estimación de densidades, funciones discriminantes, selección y extracción de características, y métodos de aprendizaje supervisado y no supervisado.
AMLS6	Configurar y utilizar plataformas de hardware avanzado (GPU u otras arquitecturas especializadas) para realizar inferencias eficientes de modelos de deep learning en entornos industriales.
AMLS7	Evaluar el rendimiento, la robustez y la seguridad funcional de los modelos de aprendizaje automático deep learning en distintos dominios, especialmente en aplicaciones industriales críticas.
AMLS8	Analizar problemas complejos mediante técnicas de investigación bibliográfica, elaborar informes técnicos, presentar resultados y argumentar decisiones basadas en evidencias y en los principios de la IA moderna.
RADLS_1	Configurar y utilizar sistemas de inteligencia artificial basados en deep learning en entornos industriales, considerando sus capacidades de procesamiento de imágenes y clasificación de objetos.
RADLS_2	Analizar arquitecturas de plataformas de computación avanzadas para la inferencia de modelos de redes neuronales, evaluando sus características y ventajas en aplicaciones industriales.
RADLS_3	Evaluar las limitaciones y requisitos de diferentes dominios industriales para la implementación de redes neuronales profundas en aplicaciones relacionadas con la seguridad funcional.
RADLS_4	Aplicar conocimientos teóricos en ejercicios prácticos, utilizando plataformas de computación avanzadas para resolver problemas reales en el ámbito del deep learning en la industria.
RAML_2	Aplicar modelos de aprendizaje supervisado a conjuntos de datos reales, evaluando su rendimiento mediante métricas adecuadas y considerando criterios de precisión y robustez.
RAML_3	Analizar y comparar diferentes enfoques de machine learning, argumentando sus ventajas y desventajas en función de las características del problema a resolver.
RAML_4	Implementar soluciones de machine learning utilizando herramientas y bibliotecas adecuadas, demostrando habilidades prácticas en entornos de laboratorio y bajo supervisión.
RAPR_1	Identificar patrones en conjuntos de datos utilizando técnicas de reconocimiento de patrones en contextos de aprendizaje automático.
RAPR_2	Aplicar algoritmos de clasificación para resolver problemas de reconocimiento de patrones en entornos simulados con precisión.

Código	Descripción
RAPR_3	Analizar la efectividad de diferentes métodos de reconocimiento de patrones en casos prácticos considerando criterios de rendimiento y precisión.
RAPR_4	Evaluar resultados de modelos de reconocimiento de patrones mediante métricas adecuadas en contextos de validación de datos.

Resumen de contenidos

Esta materia aborda los fundamentos teóricos, los métodos avanzados y las aplicaciones prácticas del aprendizaje automático, el deep learning y el reconocimiento de patrones, pilares esenciales de la inteligencia artificial moderna. Su finalidad es capacitar al estudiante para comprender, diseñar, entrenar y desplegar modelos de aprendizaje basados en datos, así como para analizar patrones complejos presentes en información visual, auditiva, secuencial o tabular. El contenido integra los principios matemáticos del aprendizaje automático, los modelos lineales y no paramétricos, las redes neuronales profundas y sus variantes para distintas modalidades de datos. La materia también cubre aspectos avanzados del despliegue industrial de sistemas de deep learning, incluyendo plataformas de GPU y hardware especializado, así como los requisitos de seguridad funcional en aplicaciones críticas. Asimismo, se estudian los fundamentos de la percepción computacional en reconocimiento de patrones, los modelos fisiológicos e informáticos de procesamiento sensorial, y las técnicas actuales utilizadas en visión por computador, análisis de señales y minería de datos. En resumen, esta materia proporciona al estudiante una formación integral que combina teoría y práctica, preparándolo para participar en proyectos de IA en los ámbitos empresariales e industriales.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	4,50	36,00	80,00
Práctica Campo	0,00		
Práctica Informática	0,00		
Práctica Laboratorio	4,50	36,00	80,00
Teoría Aula	4,00	32,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		227,50	0,00
TOTAL	13,00	331,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
-----------------------	--------	--------

Sistema de evaluación	Mínimo	Máximo
Proyecto	30%	60%
Prueba escrita	30%	60%
Prueba práctica de laboratorio/campo/informática/aula	20%	40%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Object-oriented software development

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C6	(3)	Aplicar principios de programación orientada a objetos, estructuras de datos y algoritmos para desarrollar software eficiente y mantenible.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
OSD1	Identificar y describir los principios fundamentales de la programación orientada a objetos y la sintaxis del lenguaje Java, así como los mecanismos de abstracción, encapsulación, herencia y polimorfismo.
OSD3	Describir los fundamentos teóricos del diseño y análisis de algoritmos, incluyendo recursión, relaciones de recurrencia, estrategias de diseño algorítmico y complejidad computacional.
OSD6	Analizar rigurosamente la eficiencia de algoritmos y programas, seleccionando estructuras de datos y técnicas de implementación adecuadas para mejorar el rendimiento.
OSD7	Diseñar soluciones de software seguras y extensibles utilizando principios de diseño orientado a objetos.
OSD8	Evaluar alternativas de diseño y selección de algoritmos justificando decisiones basadas en eficiencia, extensibilidad, seguridad y mantenibilidad del software.
OSD9	Realizar argumentaciones sobre las soluciones implementadas, demostrando capacidad de comunicación clara y razonada en el contexto del desarrollo de software.
RAFP_1	Identificar estructuras de control y datos en lenguajes de programación en contextos de desarrollo de software básico.
RAFP_2	Aplicar conceptos fundamentales de programación para resolver problemas simples utilizando un lenguaje de programación específico en entornos de desarrollo integrados.
RAFP_3	Analizar y depurar código fuente en proyectos de programación, utilizando herramientas adecuadas para garantizar la calidad del software.
RAFP_4	Diseñar algoritmos básicos para la resolución de problemas computacionales, considerando criterios de eficiencia y claridad en la implementación.
RAJP_2	Aplicar técnicas de depuración y pruebas unitarias en el desarrollo de aplicaciones Java, asegurando la calidad del código y la funcionalidad esperada.

Código	Descripción
RAJP_3	Desarrollar aplicaciones sencillas en Java que integren estructuras de datos básicas y algoritmos, utilizando entornos de desarrollo adecuados y siguiendo buenas prácticas de programación.
RAJP_4	Evaluar y contrastar diferentes enfoques de diseño de software en Java, considerando criterios de eficiencia, mantenibilidad y escalabilidad en proyectos de programación.
RALG_1	Identificar y clasificar diferentes tipos de algoritmos y sus aplicaciones en contextos de resolución de problemas computacionales específicos.

Resumen de contenidos

Esta materia proporciona al estudiante una formación integral en los principios y métodos fundamentales del desarrollo de software, combinando la programación orientada a objetos, el diseño modular de sistemas y el análisis y la construcción de algoritmos eficientes. Se estructura de forma progresiva, comenzando por la comprensión y la práctica del lenguaje Java como medio para consolidar la abstracción orientada a objetos y avanzando hacia técnicas de diseño profesional que incluyen especificación, modelado, extensibilidad, concurrencia y pruebas. El estudiante adquiere competencias para desarrollar programas robustos, escalables y bien estructurados mediante el uso de abstracciones adecuadas, jerarquías de clases, encapsulación, polimorfismo y mecanismos de diseño orientados a la reutilización. La materia integra, además, los fundamentos del diseño y análisis de algoritmos, permitiendo al estudiante comprender y aplicar estrategias como divide y vencerás, programación dinámica, algoritmos voraces, tratamiento de grafos, algoritmos de flujo y técnicas de aproximación. A través del estudio de la complejidad computacional, el análisis de recursiones y el razonamiento algorítmico riguroso, el estudiante desarrolla la capacidad de seleccionar, diseñar e implementar soluciones eficientes a problemas reales.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,50	28,00	80,00
Práctica Campo	0,00		
Práctica Informática	3,50	28,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	4,00	32,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		192,50	0,00
TOTAL	11,00	280,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Proyecto	20%	50%
Prueba escrita	40%	70%
Prueba práctica de laboratorio/campo/informática/aula	25%	50%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Compilers

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C15	(1)	Comprender los principios de los compiladores y lenguajes de programación y su relación con la ejecución del software.
C16	(3)	Diseñar e implementar compiladores básicos, aplicando técnicas de optimización y analizando aspectos de seguridad.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
CMP1	Identificar y describir las funciones, estructuras, etapas y principios de funcionamiento de un compilador, así como el mecanismo de ejecución de los lenguajes de programación de alto nivel.
CMP2	Explicar la aplicación de lenguajes formales, gramáticas, autómatas y teoría de la computación en la construcción de compiladores.
CMP3	Diseñar e implementar los módulos fundamentales de un compilador, incluyendo analizadores léxicos, sintácticos y semánticos, así como generadores de código.
CMP4	Construir y probar un compilador funcional de pequeña escala utilizando herramientas y técnicas avanzadas de desarrollo de software.
CMP5	Aplicar técnicas de optimización del código para mejorar el rendimiento de programas en plataformas hardware específicas.
CMP6	Analizar problemas asociados a la seguridad en el proceso de compilación y evaluar soluciones relacionadas con la protección del flujo de ejecución, la integridad de código y la detección de vulnerabilidades.
CMP7	Identificar y valorar tendencias actuales en tecnología de compiladores, justificando la relevancia de nuevas arquitecturas, lenguajes y herramientas en su evolución.
RACOT_3	Evaluar y optimizar el rendimiento de un compilador para plataformas de hardware específicas considerando criterios de eficiencia y seguridad.
RACOT_4	Aplicar teorías de lenguajes formales y autómatas en la generación automática del front end de compiladores en proyectos colaborativos.

Resumen de contenidos

Esta materia introduce al estudiante en los principios fundamentales, las técnicas de diseño y los métodos de construcción de compiladores, uno de los sistemas de software más relevantes en la ingeniería informática. Proporciona una comprensión profunda del funcionamiento interno de los lenguajes de programación de alto nivel, abarcando todas las fases del proceso de compilación: análisis léxico, análisis sintáctico, análisis semántico, generación de código intermedio, optimización y generación de código final. El estudiante adquiere competencias para diseñar, implementar y probar un compilador funcional, aplicando herramientas automáticas para la construcción de analizadores, así como estrategias de optimización orientadas a mejorar el rendimiento en arquitecturas específicas. Además, la materia aborda aspectos contemporáneos de la tecnología de compiladores, incluyendo la seguridad del programa, optimizaciones avanzadas, la infraestructura de compilación moderna y las tendencias emergentes en lenguajes de programación y entornos de ejecución. Gracias a su enfoque práctico, contribuye al fortalecimiento de las habilidades de diseño e implementación de sistemas complejos, consolidando conocimientos esenciales para áreas como sistemas operativos, computación de alto rendimiento, ingeniería del software o lenguajes de programación.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,00	24,00	80,00
Práctica Campo	0,00		
Práctica Informática	3,00	24,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	3,00	24,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		157,50	0,00
TOTAL	9,00	229,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	30%	60%
Prueba práctica de laboratorio/campo/informática/aula	30%	60%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Application development technologies

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C24	(3)	Desarrollar aplicaciones móviles, visuales y seguras, evaluando arquitecturas y técnicas en función de usabilidad, rendimiento y seguridad.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
APD1	Identificar y describir los principios fundamentales de la computación visual y el desarrollo de aplicaciones móviles, integrando conceptos de programación y técnicas de seguridad en el software.
APD2	Analizar conceptualmente las metodologías de desarrollo de aplicaciones móviles y las técnicas de computación visual, evaluando su aplicación en contextos de seguridad y rendimiento.
APD5	Evaluar críticamente la efectividad de diferentes arquitecturas de software en el desarrollo de aplicaciones móviles, justificando decisiones basadas en principios de seguridad y rendimiento.
APD6	Comparar críticamente las técnicas de desarrollo de software seguro y las metodologías de computación visual, integrando conocimientos para tomar decisiones fundamentadas en proyectos de desarrollo.
RAMAP_1	Desarrollar aplicaciones móviles en Android que implementen una interfaz gráfica de usuario adaptada a las características del dispositivo objetivo y cumpliendo con criterios de usabilidad.
RAMAP_2	Integrar servicios web RESTful en aplicaciones Android, asegurando la correcta comunicación y manejo de datos entre la aplicación y el servidor en condiciones de red variables.
RAMAP_3	Implementar almacenamiento local de datos en aplicaciones Android utilizando bases de datos adecuadas, garantizando la persistencia y recuperación eficiente de la información.
RAMAP_4	Aplicar los principios SOLID y una arquitectura limpia en el desarrollo de aplicaciones Android, asegurando la mantenibilidad, escalabilidad y facilidad de pruebas del software.
RASS_1	Identificar vulnerabilidades comunes en el software utilizando herramientas de análisis en entornos de desarrollo seguro y siguiendo estándares de calidad establecidos.
RASS_2	Aplicar buenas prácticas de desarrollo seguro en la creación de código, considerando directrices de programación y normativas de seguridad en proyectos de software.
RASS_3	Evaluar la efectividad de técnicas de depuración y validación en el proceso de desarrollo de software, asegurando la mitigación de riesgos asociados a la seguridad.
RASS_4	Diseñar soluciones de software que integren principios de ciberseguridad, garantizando el cumplimiento de regulaciones y estándares de calidad en el desarrollo.
RAVM_1	Identificar técnicas de procesamiento de imágenes digitales en contextos de visualización multimedia utilizando herramientas de software específicas.
RAVM_2	Aplicar algoritmos de compresión de datos en la optimización de archivos multimedia considerando criterios de calidad y eficiencia.
RAVM_3	Analizar la interacción entre diferentes formatos de medios visuales en entornos de desarrollo de aplicaciones multimedia con rigor técnico.

Código	Descripción
RAVM_4	Diseñar interfaces gráficas de usuario para aplicaciones de visualización multimedia utilizando principios de usabilidad y accesibilidad.

Resumen de contenidos

Esta materia aborda un conjunto integrado de tecnologías avanzadas y metodologías aplicadas al desarrollo de soluciones informáticas innovadoras en ámbitos multidisciplinares. Su objetivo es proporcionar al estudiante una visión amplia y actualizada de sectores tecnológicos emergentes, así como capacitarlo para diseñar y construir aplicaciones de nueva generación que operen en entornos complejos y distribuidos. La materia incluye competencias en diseño visual, estudiando técnicas para el tratamiento, renderizado y manipulación de imágenes y contenidos multimedia computacional; competencias en arquitectura y desarrollo móvil, estudiando la gestión de recursos limitados, la interacción persona-máquina y la programación en estas plataformas; y competencias en técnicas de desarrollo seguro, estudiando metodologías, patrones y herramientas para prevenir vulnerabilidades y asegurar la integridad, disponibilidad y confidencialidad del software durante todo su ciclo de vida. Se trata de una materia especialmente conectada con el mercado actual de aplicaciones tecnológicas en sectores como el transporte inteligente, los sistemas críticos, la comunicación visual, la movilidad digital o los servicios distribuidos de alta demanda.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	4,00	32,00	80,00
Práctica Campo	0,00		
Práctica Informática	4,50	36,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	4,50	36,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		227,50	0,00
TOTAL	13,00	331,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Examen/defensa oral	10%	20%
Trabajos académicos	40%	60%
Proyecto	30%	50%

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	30%	60%
Prueba práctica de laboratorio/campo/informática/aula	30%	60%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Optimization

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C1	(3)	Aplicar matemáticas discretas, cálculo, ecuaciones diferenciales, probabilidad y modelización matemática para formular y resolver problemas en contextos de computación y tecnológicos.
C2	(2)	Evaluar críticamente modelos matemáticos y físicos, valorando sus supuestos, limitaciones y validez en contextos ingenieriles.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
OPT1	Identificar y describir las técnicas de optimización y sus aplicaciones en problemas de ingeniería y toma de decisiones, integrando conceptos de programación lineal y métodos computacionales.
OPT2	Analizar conceptualmente los modelos matemáticos de optimización, caracterizando sus componentes y evaluando su relevancia en contextos industriales y de ingeniería.
OPT4	Diseñar y simular algoritmos de optimización, integrando técnicas de programación lineal y no lineal para abordar problemas prácticos en ingeniería.
OPT6	Comparar críticamente los resultados obtenidos de diferentes enfoques de optimización, tomando decisiones fundamentadas sobre la mejor estrategia a seguir en situaciones específicas.
RAOC_1	Aplicar técnicas de optimización en problemas de ingeniería informática utilizando herramientas computacionales adecuadas y considerando restricciones de tiempo y recursos.
RAOC_2	Analizar algoritmos de optimización en contextos de programación y desarrollo de software, evaluando su eficiencia y aplicabilidad en diferentes escenarios.
RAOC_3	Diseñar soluciones de optimización para sistemas informáticos, integrando criterios de rendimiento y escalabilidad en entornos simulados.
RAOC_4	Evaluar el impacto de diferentes métodos de optimización en el rendimiento de aplicaciones informáticas, utilizando métricas cuantitativas y cualitativas.
RAOP_1	Formular modelos matemáticos de optimización para resolver problemas de toma de decisiones en contextos industriales utilizando técnicas de programación lineal e entera.
RAOP_2	Aplicar algoritmos de optimización adecuados para obtener soluciones a problemas complejos en entornos empresariales, utilizando software especializado.

Código	Descripción
RAOP_3	Analizar e interpretar los resultados obtenidos de modelos de optimización en situaciones reales, considerando criterios de viabilidad y eficiencia.
RAOP_4	Seleccionar técnicas de investigación operativa apropiadas para abordar problemas de optimización en diferentes contextos de negocio y producción.

Resumen de contenidos

Materia orientada al estudio de los fundamentos teóricos y prácticos de optimización enfocados en la resolución de problemas computacionales e ingenieriles. Proporciona al estudiante las herramientas necesarias para analizar, formular y resolver problemas de toma de decisiones en entornos científicos, industriales y tecnológicos. Se centra en el uso de modelos matemáticos y computacionales para representar sistemas complejos, así como en la selección y aplicación de métodos de optimización adecuados para obtener soluciones eficientes y justificadas. La materia combina técnicas clásicas de programación lineal y entera con métodos de optimización basados en la formulación de modelos, el diseño y el análisis de algoritmos, y el uso de herramientas de software especializadas. Su enfoque es tanto conceptual como aplicado, orientando al estudiante a comprender el papel estratégico de la optimización en la resolución de problemas reales, la automatización de procesos, la ingeniería de sistemas y la toma de decisiones basada en datos.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,00	24,00	80,00
Práctica Campo	0,00		
Práctica Informática	0,00		
Práctica Laboratorio	3,00	24,00	80,00
Teoría Aula	2,50	20,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		148,75	0,00
TOTAL	8,50	216,75	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Trabajos académicos	20%	50%
Proyecto	0%	0%
Prueba escrita	30%	70%

Sistema de evaluación	Mínimo	Máximo
Prueba práctica de laboratorio/campo/informática/aula	30%	70%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Concurrent and distributed systems

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C13	(3)	Diseñar e implementar sistemas concurrentes y distribuidos seguros y eficientes.
C14	(2)	Evaluar arquitecturas concurrentes y distribuidas considerando sincronización, escalabilidad, rendimiento y fiabilidad.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
CDS1	Identificar y describir los principios, ventajas y riesgos de la programación concurrente, incluyendo problemas clásicos como condiciones de carrera, interbloqueos e inanición.
CDS2	Explicar los fundamentos de los sistemas distribuidos, sus desafíos y las técnicas empleadas para abordar sincronización, tolerancia a fallos, consistencia y escalabilidad.
CDS4	Aplicar técnicas de programación en sistemas distribuidos para resolver problemas prácticos, evaluando su impacto en el rendimiento y la fiabilidad del sistema.
CDS5	Evaluar críticamente el rendimiento y la eficiencia de diferentes arquitecturas concurrentes o distribuidas, tomando decisiones fundamentadas sobre su implementación en función de los requisitos de los proyectos.
CDS6	Comparar críticamente las ventajas y desventajas de diversas técnicas de gestión de concurrencia y sincronización en sistemas distribuidos, justificando la elección de métodos adecuados para situaciones específicas.
RACDS2	Analizar y diseñar sistemas que aseguren una ejecución concurrente segura y eficiente, aplicando técnicas de control de concurrencia en entornos de programación.
RACDS3	Implementar soluciones a problemas prácticos de sistemas distribuidos utilizando Java RMI y JMS, demostrando un manejo adecuado de las herramientas de programación.

Resumen de contenidos

Materia centrada en el estudio de los principios fundamentales y las técnicas necesarias para comprender, diseñar e implementar sistemas concurrentes y distribuidos en el ámbito de los sistemas computacionales. Se centra en el estudio de los retos inherentes a la ejecución simultánea de procesos y a la coordinación de sistemas distribuidos que operan en entornos heterogéneos y potencialmente inseguros. A lo largo de la materia, el estudiante adquiere una comprensión profunda de los mecanismos de concurrencia, incluyendo hilos, sincronización, exclusión mutua y la gestión de condiciones de carrera, interbloqueos e inanición. Asimismo, se

exploran los fundamentos de los sistemas distribuidos, sus modelos de comunicación, la tolerancia a fallos, la consistencia, la sincronización distribuida y la escalabilidad. Esta materia constituye un componente esencial para comprender el funcionamiento de sistemas modernos, desde aplicaciones multicore hasta servicios distribuidos, entornos cloud y arquitecturas basadas en microservicios, proporcionando una base sólida para estudios avanzados en sistemas operativos, redes, computación paralela y sistemas ciberfísicos.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	1,50	12,00	80,00
Práctica Campo	0,00		
Práctica Informática	0,00		
Práctica Laboratorio	3,00	24,00	80,00
Teoría Aula	1,50	12,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		105,00	0,00
TOTAL	6,00	153,00	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Proyecto	20%	60%
Prueba escrita	20%	60%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Computer networks

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C11	(1)	Comprender los principios de las redes de computadores y los protocolos TCP/IP para analizar y configurar infraestructuras de comunicación.
C12	(2)	Resolver problemas de redes, evaluando riesgos y vulnerabilidades y proponiendo medidas básicas de seguridad.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
CN1	Identificar y describir los principios fundamentales de las redes de computadores, la arquitectura TCP/IP y la función de cada una de sus capas.
CN2	Explicar los principales protocolos de red, incluyendo HTTP, DNS, TCP, UDP, IP y los mecanismos básicos de enrutamiento y direccionamiento.
CN3	Analizar el funcionamiento de redes y protocolos y mediante herramientas de diagnóstico, simulación o captura de tráfico.
CN4	Configurar aspectos básicos de una red, incluyendo direccionamiento IP, parámetros de transporte, servicios de aplicación y componentes de enrutamiento.
CN5	Resolver problemas habituales en redes de computadores mediante procedimientos de verificación, análisis de errores y aplicación de buenas prácticas.
CN6	Evaluar diferentes soluciones de diseño de redes considerando criterios de escalabilidad, eficiencia, seguridad y adecuación al contexto técnico.
CN7	Evaluar los riesgos y vulnerabilidades básicos en redes de computadores, proponiendo medidas adecuadas para mejorar su fiabilidad y seguridad en diferentes escenarios de uso.
RACN_3	Configurar y poner en marcha redes básicas utilizando herramientas de simulación, considerando criterios de eficiencia y seguridad.

Resumen de contenidos

Esta materia introduce al estudiante en los fundamentos de las redes de computadores y en los principios que rigen la comunicación entre sistemas digitales en entornos locales y globales. Proporciona una visión general y estructurada de la arquitectura TCP/IP (el modelo dominante en Internet) mediante un enfoque de arriba hacia abajo que facilita la comprensión progresiva de los distintos niveles y sus funciones. El contenido abarca conceptos esenciales como los protocolos de aplicación, transporte y enrutamiento, el direccionamiento IP, la arquitectura cliente-servidor, el control de la congestión, los medios de transmisión y el funcionamiento de la capa de enlace. Aunque mantiene un carácter introductorio y descriptivo, la materia incluye actividades prácticas orientadas al análisis, configuración y resolución de problemas de red, lo que permite al estudiante adquirir competencias básicas para desenvolverse en entornos de comunicación digital.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	2,00	16,00	80,00
Práctica Campo	0,00		

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Informática	0,00		
Práctica Laboratorio	2,50	20,00	80,00
Teoría Aula	2,50	20,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		122,50	0,00
TOTAL	7,00	178,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	30%	60%
Prueba práctica de laboratorio/campo/informática/aula	30%	60%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Information systems development and management

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C10	(2)	Evaluar la calidad de sistemas software y de información considerando mantenibilidad, seguridad, rendimiento y consistencia de datos.
C9	(3)	Diseñar, desarrollar e integrar sistemas de información aplicando metodologías de ingeniería del software y bases de datos relacionales.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
ISDM1	Identificar y describir los fundamentos de los sistemas de bases de datos, sus modelos de datos, la arquitectura de un DBMS y los mecanismos de integridad, seguridad, transacciones y concurrencia.
ISDM2	Explicar y caracterizar las metodologías ágiles y los enfoques de desarrollo de software, así como su aplicación en la gestión de sistemas de información y bases de datos.
ISDM4	Implementar soluciones de software utilizando herramientas modernas y enfoques de desarrollo, integrando prácticas de gestión de bases de datos y control de calidad en el proceso.
ISDM5	Evaluar críticamente la calidad de un sistema de información, integrando conocimientos de ingeniería de software y bases de datos para justificar decisiones de mejora en su desarrollo.
ISDM6	Comparar críticamente diferentes enfoques de desarrollo de software y gestión de bases de datos, tomando decisiones fundamentadas sobre su aplicación en proyectos específicos.
RADIS_2	Diseñar modelos de datos conceptuales utilizando diagramas UML-ER en situaciones prácticas que reflejen requisitos reales de información.
RADIS_3	Programar consultas SQL complejas para la definición, manipulación y recuperación de datos en un sistema de gestión de bases de datos (DBMS) con precisión técnica.
RADIS_4	Implementar mecanismos de gestión de transacciones y control de concurrencia en bases de datos, considerando criterios de integridad y seguridad de los datos.
RASE_1	Aplicar principios de desarrollo de software orientado a objetos en la creación de aplicaciones utilizando entornos de desarrollo como Visual Studio y frameworks como .NET Entity Framework.
RASE_2	Diseñar y modelar arquitecturas de software en un contexto de trabajo en grupo, considerando aspectos de calidad y escalabilidad del sistema desarrollado.
RASE_3	Implementar y probar componentes de software en un proyecto práctico, utilizando metodologías ágiles y herramientas de integración continua como Azure DevOps.

Resumen de contenidos

Esta materia ofrece una formación integral en los principios, métodos y tecnologías necesarios para el desarrollo de sistemas software y la gestión de la información, combinando la ingeniería del software orientada a objetos con el diseño, implementación y administración de bases de datos relacionales. Su objetivo es capacitar al estudiante para abordar el ciclo completo de creación de un sistema de información moderno, desde el análisis y diseño hasta el desarrollo, prueba y operación del sistema. La materia introduce las metodologías actuales de desarrollo (incluyendo enfoques ágiles, DevOps y desarrollo dirigido por modelos) junto con conceptos

fundamentales como arquitectura multicapa, diseño orientado a objetos, pruebas de software y gestión de proyectos. Se fomenta el uso de herramientas profesionales y entornos de desarrollo contemporáneos que permiten aplicar estas metodologías en contextos reales. Por otro lado, también se proporciona una base sólida en sistemas de información y bases de datos, cubriendo el modelado conceptual mediante UML y E/R, las transformaciones al modelo relacional, la normalización, el diseño lógico y físico, así como el uso avanzado del lenguaje SQL para la definición, manipulación y consulta de datos.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,00	24,00	80,00
Práctica Campo	0,00		
Práctica Informática	6,00	48,00	80,00
Práctica Laboratorio	3,00	24,00	80,00
Teoría Aula	0,00		
Teoría Seminario	0,00		
Trabajo autónomo		210,00	0,00
TOTAL	12,00	306,00	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Trabajos académicos	20%	50%
Proyecto	20%	50%
Prueba escrita	20%	50%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Intelligent systems

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C21	(2)	Diseñar y analizar sistemas de computación inteligente y sistemas multiagente, justificando decisiones arquitectónicas

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
ISYS1	Identificar y describir los conceptos fundamentales de la computación inteligente y los sistemas multi-agente, integrando teorías, modelos y arquitecturas relevantes de ambas áreas.
ISYS2	Analizar conceptualmente las aplicaciones de la inteligencia artificial, el aprendizaje automático y la percepción computacional en la resolución de problemas complejos, integrando conocimientos de agentes inteligentes.
ISYS4	Implementar soluciones basadas en computación inteligente, integrando métodos de aprendizaje automático y arquitecturas de agentes para abordar desafíos reales en contextos específicos.
ISYS5	Evaluar críticamente las capacidades, limitaciones y riesgos de los agentes inteligentes en la resolución de problemas, justificando decisiones basadas en análisis de resultados y contextos aplicados.
ISYS6	Comparar críticamente las ventajas y desventajas de diversas arquitecturas de agentes, integrando conocimientos de la computación inteligente.
RAIAG_2	Aplicar teorías y modelos de agentes inteligentes para diseñar soluciones a problemas complejos en entornos distribuidos, considerando criterios de eficiencia y efectividad.
RAIAG_3	Analizar arquitecturas de sistemas multi-agente y sus aplicaciones en la automatización empresarial, evaluando su impacto en la toma de decisiones.

Resumen de contenidos

Esta materia proporciona una introducción sólida y actualizada a los fundamentos de la computación inteligente y a los sistemas basados en agentes, tecnologías clave en el desarrollo de soluciones autónomas en entornos complejos y distribuidos. El estudiante adquiere una perspectiva global del campo, que incluye su evolución histórica, tendencias emergentes, modelos teóricos, arquitecturas fundamentales y aplicaciones reales en múltiples dominios. Por un lado, se abordan los conceptos esenciales de la computación inteligente, incluyendo los modelos computacionales inspirados en procesos cognitivos, las bases de la percepción computacional, y las conexiones con áreas como el aprendizaje automático, el reconocimiento de patrones y la minería de datos. Se analizan el estado actual del campo y los retos científicos y tecnológicos asociados a la construcción de sistemas inteligentes. Por otro lado, la materia se centra en la teoría de agentes y sistemas multi-agente, introduciendo el concepto de agente autónomo, así como sus capacidades cognitivas, su arquitectura interna y sus mecanismos de interacción, cooperación y negociación. Se estudian, además, tecnologías de acuerdo, organizaciones virtuales y negociación automática, así como la nueva generación de agentes autónomos basados en Agentic AI, que integran modelos de lenguaje, aprendizaje, automatización y toma de decisiones autónoma.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	2,50	20,00	80,00
Práctica Campo	0,00		
Práctica Informática	1,50	12,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	4,50	36,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		148,75	0,00
TOTAL	8,50	216,75	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Proyecto	10%	40%
Prueba escrita	30%	60%
Prueba práctica de laboratorio/campo/informática/aula	30%	60%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Advanced computing technologies

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
C22	(3)	Aplicar paradigmas de computación avanzada (como computación en la nube, de alto rendimiento y cuántica) así como conceptos básicos de tecnologías aeroespaciales para analizar y resolver problemas complejos en contextos tecnológicos emergentes.
C23	(2)	Evaluar críticamente las limitaciones, oportunidades, tendencias y retos de las tecnologías de computación avanzada y del ámbito aeroespacial, valorando su impacto tecnológico, científico, ambiental y social, y seleccionando enfoques adecuados en función de criterios de rendimiento, escalabilidad, sostenibilidad y viabilidad técnica.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Código	Descripción
ADCT1	Analizar conceptualmente los fundamentos de la computación cuántica, los elementos funcionales de los ordenadores cuánticos y el proceso de compilación de algoritmos cuánticos.
ADCT2	Identificar y describir las tecnologías y arquitecturas de computación paralela y distribuida, sus modelos de programación y sus aplicaciones en HPC y cloud computing.
ADCT4	Desarrollar programas paralelos para arquitecturas de memoria compartida, distribuida y GPUs, aplicando técnicas de optimización en distintos niveles.
ADCT6	Evaluar la adecuación de diferentes modelos de computación (cuántico, paralelo, distribuido, GPU, cloud) para resolver problemas complejos, considerando restricciones de recursos, escalabilidad y eficiencia.
ADCT7	Analizar las limitaciones actuales de la computación cuántica y de alto rendimiento, proponiendo estrategias o enfoques alternativos fundamentados en criterios técnicos y científicos.
CHPC_1	Aplicar lenguajes de programación paralela como OpenMP y MPI en entornos de procesamiento de datos masivos utilizando criterios de eficiencia y optimización.
CHPC_2	Configurar y utilizar entornos de computación en la nube para implementar soluciones de procesamiento distribuido, considerando aspectos de escalabilidad y rendimiento.
CHPC_3	Desarrollar programas utilizando el modelo MapReduce para el procesamiento de grandes volúmenes de datos en entornos de computación distribuida con precisión técnica.
CHPC_4	Evaluar el proceso de optimización de software en entornos híbridos, analizando el uso de recursos y la mejora del rendimiento en aplicaciones paralelas.
RAQC_2	Aplicar el marco de programación QisKit para diseñar y ejecutar circuitos cuánticos en simuladores y dispositivos cuánticos reales.
RAQC_3	Analizar algoritmos cuánticos conocidos y su relevancia en la solución de problemas complejos en comparación con algoritmos clásicos.

Resumen de contenidos

Esta materia ofrece una visión integrada de las tecnologías más avanzadas y emergentes en el ámbito de la computación, que abarcan tanto la computación cuántica como la computación de altas prestaciones (HPC) y la computación en la nube. El estudiante adquiere los fundamentos teóricos y prácticos necesarios para comprender y utilizar plataformas de cómputo que representan la frontera tecnológica actual, desde los procesadores cuánticos NISQ hasta arquitecturas masivamente paralelas y sistemas distribuidos en la nube. Por un lado, se introducen los principios fundamentales de la computación cuántica (qubits, puertas cuánticas,

mediciones, circuitos cuánticos, algoritmos elementales y nociones de corrección de errores), se analiza el estado actual de la tecnología cuántica, las limitaciones de los dispositivos NISQ y el proceso de compilación necesario para ejecutar algoritmos cuánticos en hardware real. Por otro lado, la materia también aborda la computación de alto rendimiento y la computación en la nube, incluyendo la programación paralela en arquitecturas de memoria compartida y distribuida, el uso de aceleradores como GPUs y el procesamiento masivo mediante modelos como MapReduce. El estudiante aprende a optimizar código para distintos niveles de paralelismo, a desplegar sistemas distribuidos en plataformas cloud y a comprender los retos de escalabilidad, rendimiento y eficiencia en entornos híbridos. La materia proporciona una formación avanzada indispensable para comprender los sistemas computacionales del futuro, preparar al alumnado para su uso profesional y académico e introducirlo en áreas punteras de investigación y desarrollo tecnológico.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	3,00	24,00	80,00
Práctica Campo	0,00		
Práctica Informática	3,00	24,00	80,00
Práctica Laboratorio	0,00		
Teoría Aula	3,00	24,00	80,00
Teoría Seminario	0,00		
Trabajo autónomo		157,50	0,00
TOTAL	9,00	229,50	

Metodologías de la enseñanza

Metodologías de la enseñanza
Lección magistral
Tutoría
Prácticas

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Prueba escrita	30%	60%
Prueba práctica de laboratorio/campo/informática/aula	30%	60%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Final degree project

Resultados fundamentales cubiertos por la materia

Código	Tipo	Descripción
CT1	(2)	Actuar con ética y responsabilidad profesional ante los desafíos sociales, ambientales y económicos, teniendo como referentes los principios y valores democráticos y los Objetivos de Desarrollo Sostenible.
CT2	(2)	Proponer soluciones creativas e innovadoras a situaciones o problemas complejos, propios del ámbito de conocimiento, para dar respuesta a las diversas necesidades profesionales y sociales.
CT4	(2)	Comunicarse de manera efectiva, tanto de forma oral como escrita, adaptándose a las características de la situación y de la audiencia.
CT5	(2)	Actuar con autonomía en el aprendizaje, tomando decisiones fundamentadas en diferentes contextos, emitiendo juicios en base a la experimentación y el análisis y transfiriendo el conocimiento a nuevas situaciones.
TFG	(2)	Ejercicio original a realizar individualmente y presentar y defender ante un tribunal universitario, consistente en un proyecto en el ámbito de las tecnologías específicas del ámbito del título de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas.

Tipos

- 1 - Conocimiento o contenido
- 2 - Competencia
- 3 - Habilidad o destreza

Resultados del aprendizaje

Resumen de contenidos

Se trata de un trabajo relacionado con una o varias de las materias impartidas relacionadas directamente con el ámbito del título, de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas.

Este trabajo debe de servir para que el estudiante termine de desarrollar la capacidad de completar una experiencia de diseño de proyecto completa, abordándomelas todas las fases del mismo:

Planteamiento del problema y oportunidad

Toma en consideración de los condicionantes (normas técnicas y regulación, necesidades, requisitos y especificaciones)

Establecimiento de objetivos

Generación de soluciones creativas

Evaluación de múltiples soluciones y toma de decisiones

Evaluación del cumplimiento de objetivos

Evaluación del impacto global y alcance (contribuciones y recomendaciones prácticas)

El trabajo debe plasmarse en una memoria escrita en la que se recojan de forma clara y completa cada una de estas fases del proyecto, demostrando una capacidad de comunicación escrita adecuada en términos de claridad, formalidad, y originalidad.

El trabajo se expondrá y defenderá públicamente ante un tribunal compuesto por profesores de la titulación. Deberá demostrar capacidad de expresarse oralmente con claridad, siéndonosla capaz de sintetizar de manera adecuada el trabajo realizado, comunicándomelos de manera efectiva su contribución al mismo, su relevancia, originalidad, e impacto. Deberá ser capaz de responder de manera adecuada a las preguntas que el tribunal le formule.

El trabajo final de grado fomenta habilidades transversales como la gestión del tiempo, la planificación de tareas, la resolución de problemas, la ética profesional y la capacidad para trabajar de manera autónoma o en colaboración con otros. Es un paso fundamental para consolidar la transición del estudiante hacia el ámbito profesional o la especialización académica mediante estudios de posgrado.

Actividades formativas

Actividad formativa	ECTS	Horas dedicación	% presencialidad
Práctica Aula	24,00	192,00	
Práctica Campo	0,00		
Práctica Informática	0,00		
Práctica Laboratorio	0,00		
Teoría Aula	0,00		
Teoría Seminario	0,00		
Trabajo autónomo		600,00	0,00
TOTAL	24,00	792,00	

Metodologías de la enseñanza

Metodologías de la enseñanza
Aprendizaje orientado a proyectos
Prácticas
Otras metodologías

Observaciones a los sistemas de evaluación

Sistema de evaluación	Mínimo	Máximo
Examen/defensa oral	20%	40%
Proyecto	50%	80%

Reconocimientos

Número máximo de créditos reconocibles por actividades estudiantiles: 0,00

Número máximo de créditos reconocibles por experiencia laboral: 0,00

Número máximo de créditos reconocibles por prácticas en empresa curriculares: 0,00

Resultados fundamentales por materia

[illegible]

[illegible][illegible]

Plan de estudios

Módulos	Materias	Asignaturas
Basic training(65 ECTS)	Fundamentals of mathematics (15 ECTS), Formación Básica	Discrete mathematics (6 ECTS), Curso 1, Formación Básica, Semestre A
		Mathematics I (9 ECTS), Curso 1, Formación Básica, Semestre A
	Mathematical methods and modeling (12 ECTS), Formación Básica	Mathematics II (6 ECTS), Curso 1, Formación Básica, Semestre B
		Probability and mathematical modeling (BEI) (6 ECTS), Curso 2, Formación Básica, Semestre A
	Computing and algorithmic development (18 ECTS), Formación Básica	Automata theory and computability (6 ECTS), Curso 1, Formación Básica, Semestre B
		Object-oriented design and construction (BEI) (6 ECTS), Curso 2, Formación Básica, Semestre B
		Data structures and algorithms (6 ECTS), Curso 2, Formación Básica, Semestre B
	Architecture and operating systems (20 ECTS), Formación Básica	Operating systems (BEI) (9 ECTS), Curso 2, Formación Básica, Semestre B
Computer organization (BEI) (11 ECTS), Curso 2, Formación Básica, Semestre A		
Complementary training(45.5 ECTS)	Language skills for engineering (16 ECTS), Obligatorio	English reading I (BEI) (1.5 ECTS), Curso 1, Obligatorio, Semestre A
		English reading II (BEI) (1.5 ECTS), Curso 1, Obligatorio, Semestre B
		English reading III (BEI) (1.5 ECTS), Curso 2, Obligatorio, Semestre A
		English reading IV (BEI) (1.5 ECTS), Curso 2, Obligatorio, Semestre B
		English writing I (BEI) (1 ECTS), Curso 1, Obligatorio, Semestre A
		English writing II (BEI) (1 ECTS), Curso 1, Obligatorio, Semestre B
		English writing III (BEI) (1 ECTS), Curso 2, Obligatorio, Semestre A
		English writing IV (BEI) (1 ECTS), Curso 2, Obligatorio, Semestre B
		English speaking I (BEI) (1 ECTS), Curso 1, Obligatorio, Semestre A
		English speaking II (BEI) (1 ECTS), Curso 1, Obligatorio, Semestre B
		Chinese / Spanish I (BEI) (2 ECTS), Curso 1, Obligatorio, Semestre A
		Chinese / Spanish II (BEI) (2 ECTS), Curso 1, Obligatorio, Semestre B
	Fundamentals of physics (9 ECTS), Obligatorio	Physics (9 ECTS), Curso 1, Obligatorio, Anual
	Electronics (5 ECTS), Obligatorio	Basic training in electronic design (BEI) (5 ECTS), Curso 1, Obligatorio, Semestre B
	Introduction to aerospace technologies (3 ECTS), Obligatorio	Introduction to aeronautics and astronautics (BEI) (3 ECTS), Curso 2, Obligatorio, Semestre B

	Mathematical tools (12.5 ECTS), Obligatorio	Matrix analysis and optimization (BEI) (4 ECTS), Curso 2, Obligatorio, Semestre A
		Mathematical modeling I (BEI) (2 ECTS), Curso 3, Obligatorio, Semestre A
		Mathematical modeling II (BEI) (2 ECTS), Curso 4, Obligatorio, Semestre A
		Advanced data analysis in computer engineering (4.5 ECTS), Curso 3, Obligatorio, Semestre B
Artificial intelligence(21.5 ECTS)	Introduction to artificial intelligence (8.5 ECTS), Obligatorio	Introduction to artificial intelligence (BEI) (4 ECTS), Curso 1, Obligatorio, Semestre B
		Artificial intelligence techniques and applications (4.5 ECTS), Curso 4, Obligatorio, Semestre A
	Advanced machine learning systems (13 ECTS), Obligatorio	Deep learning systems for industry (4.5 ECTS), Curso 3, Obligatorio, Semestre B
		Machine learning (4.5 ECTS), Curso 4, Obligatorio, Semestre A
		Pattern recognition I (BEI) (2 ECTS), Curso 3, Obligatorio, Semestre A
		Pattern recognition II (BEI) (2 ECTS), Curso 4, Obligatorio, Semestre A
Basics and techniques of programming(41.5 ECTS)	Object-oriented software development (11 ECTS), Obligatorio	Fundamentals of programming (BEI) (4 ECTS), Curso 1, Obligatorio, Semestre A
		Algorithm design and analysis (BEI) (4 ECTS), Curso 3, Obligatorio, Semestre A
		Java programming I (BEI) (1.5 ECTS), Curso 3, Obligatorio, Semestre A
		Java programming II (BEI) (1.5 ECTS), Curso 4, Obligatorio, Semestre A
	Compilers (9 ECTS), Obligatorio	Compiler technology (BEI) (9 ECTS), Curso 3, Obligatorio, Semestre A
	Application development technologies (13 ECTS), Optativo	Visual media computing (BEI) (4 ECTS), Curso 3, Obligatorio, Semestre B
		Mobile application development (4.5 ECTS), Curso 4, Obligatorio, Semestre A
		Secure software development and deployment (4.5 ECTS), Curso 3, Obligatorio, Semestre B
	Optimization (8.5 ECTS), Obligatorio	Optimization techniques (4.5 ECTS), Curso 3, Obligatorio, Semestre B
		Optimization in computer engineering (BEI) (4 ECTS), Curso 4, Obligatorio, Semestre A
Computing systems and technologies(42.5 ECTS)	Concurrent and distributed systems (6 ECTS), Obligatorio	Concurrency and distributed systems (6 ECTS), Curso 2, Obligatorio, Semestre B
	Computer networks (7 ECTS), Obligatorio	Computer networks A (4 ECTS), Curso 2, Obligatorio, Semestre A
		Computer networks B (3 ECTS), Curso 3, Obligatorio, Semestre B
	Information systems development and management (12 ECTS), Obligatorio	Databases and information systems (6 ECTS), Curso 3, Obligatorio, Semestre A
		Software engineering (6 ECTS), Curso 3, Obligatorio, Semestre A
	Intelligent systems (8.5 ECTS), Obligatorio	Introduction to intelligent computing (BEI) (4 ECTS), Curso 4, Obligatorio, Semestre A

	Advanced computing technologies (9 ECTS), Obligatorio	Intelligent agents (4.5 ECTS), Curso 3, Obligatorio, Semestre B
		Quantum computing (4.5 ECTS), Curso 4, Obligatorio, Semestre B
		Cloud computing and high-performance computing (4.5 ECTS), Curso 4, Obligatorio, Semestre A
Final degree project(24 ECTS)	Final degree project (24 ECTS), Trabajo Fin Titulación	Bachelor's thesis (24 ECTS), Curso 4, Trabajo Fin Titulación, Semestre B