



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803200 - Gestión empresarial		<b>Abrev:</b> GE	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Business Management		<b>Carácter:</b> Formación básica	
<b>Materia:</b> Empresa		<b>6 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> No hay			
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Administración Financiera y Contabilidad		<b>Coordinador:</b> Sánchez Martín, Pilar	

**Descripción de contenidos mínimos:**

- Estudiar la naturaleza de la empresa y su relación con el entorno desde el punto de vista organizativo y financiero.
- Conocer la forma en que las empresas toman sus decisiones de inversión y financiación.
- Adquirir unas nociones básicas de contabilidad financiera.

**Programa detallado:****TEMA I LA EMPRESA: ORGANIZACIÓN Y ESTRUCTURA**

1. Entorno de la empresa
2. El empresario
3. La organización y las funciones de la empresa
4. La estructura organizativa

**TEMA II DECISIONES DE INVERSIÓN Y FINANCIACIÓN**

1. Decisiones de inversión
2. La elección de las fuentes de financiación

**TEMA III CONCEPTOS BASICOS DE CONTABILIDAD**

1. La contabilidad como sistema de información de la empresa
2. Emisores de normas contables

**TEMA IV PATRIMONIO Y MARCO CONCEPTUAL DE LA CONTABILIDAD**

1. Concepto de patrimonio.
2. Marco conceptual de la contabilidad
3. Los principios contables

**TEMA V LAS CUENTAS ANUALES: NORMAS DE ELABORACIÓN**

1. Documentos que integran las Cuentas Anuales
2. Formulación de las Cuentas Anuales. Aprobación y publicidad de las mismas
3. Estructura de las Cuentas Anuales

**TEMA VI EL BALANCE**

1. Elementos del Balance
2. Activos
3. Pasivos
4. Patrimonio Neto

**TEMA VII LA CUENTA DE PÉRDIDAS Y GANANCIAS**

1. Concepto de Gasto
2. Concepto de Ingreso
3. Gastos e ingresos imputados al patrimonio neto
4. Determinación del resultado

**TEMA VIII METODOLOGÍA DE LA PARTIDA DOBLE**

1. Concepto de cuenta. Lectura e interpretación de las cuentas
2. Clasificación de las cuentas
3. Teoría del cargo y del abono
4. Criterios de registro y reconocimiento contable de los elementos de las cuentas anuales
5. Métodos de registro: libros Diario y Mayor
6. Ciclo contable

**TEMA IX: PROBLEMÁTICA CONTABLE DE LA VALORACIÓN**

1. Amortizaciones
2. Provisiones y deterioros

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



3. Variación de existencias

**Programa detallado en inglés:**

CHAPTER 1: INTRODUCTION TO ACCOUNTING

- 1.1 TYPES OF COMPANIES
- 1.2 WHAT TYPE OF COMPANY TO CHOOSE?
- 1.3 ACCOUNTING CONCEPT AND AIMS
- 1.4 ACCOUNTING USERS
- 1.5 TYPES OF ACCOUNTING
- 1.6 SOCIAL RESPONSIBILITY AND CORPORATE ETHICS

CHAPTER 2: CONCEPTUAL ACCOUNTING FRAMEWORK

- 2.1 ACCOUNTING FRAMEWORK
- 2.2 THE ACCOUNTING CONCEPTUAL FRAMEWORK PURPOSE AND STATUS
- 2.3 THE ACCOUNTING CONCEPTUAL FRAMEWORK IN EUROPE
- 2.4. SPANISH ACCOUNTING CONCEPTUAL FRAMEWORK

CHAPTER 3: ANNUAL REPORTS

- 3.1 CONCEPT OF ANNUAL REPORT
- 3.2 FINANCIAL STATEMENTS

CHAPTER 4: BOOKKEEPING

- 4.1 ACCOUNTING BOOKS
- 4.2 HOW THE ACCOUNTS WORK
- 4.3 DOUBLE-ENTRY BOOKKEEPING SYSTEM: DEBIT AND CREDIT RULES
- 4.4 BASIC ACCOUNTING LOGIC
- 4.5 ADJUSTMENTS

CHAPTER 5: ACCOUNTING CYCLE

- 5.1 THE ACCOUNTING CYCLE PROCESS
- 5.2 COMPLETING ACCOUNTING CYCLE FOR SERVICE FIRMS
- 5.3 COMPLETING ACCOUNTING CYCLE FOR MERCHANDISING FIRMS

CHAPTER 6: MOST COMMON BUSINESS TRANSACTIONS

- 6.1. ASSETS ACCOUNTING
- 6.2. LIABILITIES ACCOUNTING
- 6.3. EQUITY ACCOUNTING
- 6.4. ACCRUAL AND DEFERRAL ACCOUNTING
- 6.5. ACCOUNTING FOR TAXES

**Competencias de la asignatura:**

**Generales:**

CG6-Conocimiento adecuado del concepto de empresa, marco institucional y jurídico de la empresa. Organización y gestión de empresas.

**Específicas:**

No tiene

**Básicas y Transversales:**

CB1-Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio

CB2-Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- CB3-Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética
- CB4-Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado
- CB5-Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía
- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Desarrollar las prácticas y ejercicios tomando decisiones sobre su diseño. (CT3)
- Desarrollar soluciones para supuestos prácticos concretos tomando decisiones y analizando las posibilidades. (CG6, CT2)
- Trabajos en equipo sobre organización y gestión de empresas. (CG6, CT1)

**Evaluación detallada:**

Todas las pruebas realizadas en cada asignatura serán comunes a todos los grupos de la misma.  
Es requisito para aprobar la asignatura tanto en la convocatoria ordinaria como en la extraordinaria la asistencia al menos al 80% de las clases (teóricas, prácticas y laboratorio).  
La calificación final tanto en la convocatoria ordinaria como en la extraordinaria tendrá en cuenta:  
• Exámenes sobre la materia: 70%  
• Otras actividades: 30%  
Para poder hacer la media es necesario obtener un mínimo de 4 sobre 10 en el examen final de la convocatoria ordinaria y/o extraordinaria.  
En el apartado “Otras actividades” se valorará la participación activa en el proceso de aprendizaje, la realización de prácticas y ejercicios y la realización de las actividades dirigidas propuestas por los profesores durante el curso. Esta parte de la nota (30%) se consigue durante el curso y no se puede recuperar.

**Actividades docentes:**

Reparto de créditos:	Otras actividades:
Teoría: 3,00	Clases teóricas
Problemas: 3,00	1 hora semanal
Laboratorios: 0,00	
	Clases prácticas
	3 horas semanales
	Seminarios
	2 horas quincenales

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID  
FACULTAD DE INFORMÁTICA

**Bibliografía:**

**BIBLIOGRAFÍA:**

- Plan General de Contabilidad, 2007.
- Supuestos prácticos. Material elaborado por los profesores de la asignatura, 2010.
- Camacho, M<sup>a</sup> del Mar y Rivero, M<sup>a</sup> José: Introducción a la Contabilidad Financiera en el Marco del EEES, Editorial Pearson, Madrid, 2010.
- CTO Hacienda: Manual de Contabilidad Financiera, Madrid, 2009.
- Amador Fernández, S. y Romano Aparicio, J.: Manual del Nuevo Plan General Contable 2007, Editorial CEF, Madrid, 2007.
- Muñoz Merchante, Ángel: Fundamentos de Contabilidad, Editorial Ramón Areces, Madrid, 2008.

**Enlaces de Interés:**

- AECA: [www.aeca.es](http://www.aeca.es)
- CNMV, COMISIÓN NACIONAL DEL MERCADO DE VALORES: [www.cnmv.es](http://www.cnmv.es)
- FASB, FINANCIAL ACCOUNTING STANDARDS BOARD / US GAAP: [www.fasb.org](http://www.fasb.org)
- ICAC, INSTITUTO DE CONTABILIDAD Y AUDITORÍA DE CUENTAS: [www.icac.mineco.es](http://www.icac.mineco.es)
- [www.noticiasjuridicas.com](http://www.noticiasjuridicas.com)
- [www.cef.es](http://www.cef.es)
  
- Camacho-Miñano, M.M., Akpinar, M., Rivero-Menéndez M.J., Urquía-Grande, E. and Eskola, A. (2012). Introduction to Financial Accounting. European Financial Accounting Manual. Ed.: Piramide.
- Camacho, M<sup>a</sup> del Mar y Rivero, M<sup>a</sup> José: Introducción a la Contabilidad Financiera en el Marco del EEES, Editorial Pearson, Madrid, 2010.
- Amador Fernández, S. y Romano Aparicio, J.: Manual del Nuevo Plan General Contable 2007, Editorial CEF, Madrid, 2007.
- Muñoz Merchante, Ángel: Fundamentos de Contabilidad, Editorial Ramón Areces, Madrid, 2008.

Ficha docente guardada por última vez el 03/06/2021 23:01:00 por el usuario: Vicedecanato de Estudios y Calidad

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803201 - Fundamentos de electricidad y electrónica <b>Asignatura en Inglés:</b> Introduction to the concepts of electricity and electronics		<b>Abrev:</b> FEE <b>Carácter:</b> Formación básica	<b>6 ECTS</b>
<b>Materia:</b> Física		<b>6 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> No hay			
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Interdepartamental EMFTEL/FM		<b>Coordinador:</b> Sánchez Balmaseda, Margarita	

**Descripción de contenidos mínimos:**

- Conceptos básicos de magnitudes eléctricas.
- Conducción eléctrica.
- Capacidad.
- Tipos de señales en un circuito: ondas.
- Elementos de un circuito y características tensión-corriente.
- Métodos básicos de análisis de circuitos.
- Carga y descarga de un condensador.
- Introducción a los semiconductores: semiconductores intrínsecos y extrínsecos.
- Conductividad eléctrica en semiconductores.
- Aplicaciones.
- Unión p-n.
- Característica de un diodo.
- Modelo de gran señal.
- Circuitos con diodos.
- Dispositivos optoelectrónicos.
- Transistor MOSFET.
- Circuitos con transistores.

**Programa detallado:**

PARTE 1. Conceptos básicos de magnitudes eléctricas y circuitos eléctricos.

Tema I. Campo eléctrico. Corriente eléctrica

1. Ley de Coulomb y campo eléctrico: Carga eléctrica. Ley de Coulomb. Campo eléctrico. 2. Energía potencial y potencial eléctrico: Energía potencial eléctrica. Potencial eléctrico y diferencia de potencial. 3. Condensadores: Los materiales conductores. Capacidad y condensadores. 4. Corriente eléctrica: Corriente eléctrica. Conductividad eléctrica y Ley de Ohm. Ley de Joule. Potencia eléctrica.

Tema II. Campo magnético. Ondas electromagnéticas

1. Campo magnético: Fuentes de campo magnético. Inducción magnética. 2 Ondas electromagnéticas: Ondas: conceptos básicos. Ondas electromagnéticas. Fotones

Tema III. Circuitos eléctricos

1. Análisis de circuitos: Definiciones básicas: nodo, lazo, malla y rama. Leyes de Kirchoff. Tipos de elementos en un circuito. Fuentes DC y fuentes AC. Característica tensión corriente. Asociación de elementos. Métodos básicos de análisis de circuitos. Teorema de Thévenin. 2. Circuitos con señales variables en el tiempo: Carga y descarga de un condensador.

PARTE 2. Semiconductores y dispositivos electrónicos.

TEMA IV. Dispositivos de unión de dos terminales

1. Introducción a los semiconductores: Conductores, aislantes y semiconductores. Semiconductores extrínsecos. Conductividad eléctrica en semiconductores. Generación y recombinación de portadores Aplicaciones. 2. Unión p-n: La unión p-n en equilibrio. La unión p-n polarizada. Curva característica del diodo. 3. El diodo de unión como elemento de un circuito. Punto de trabajo y recta de carga. Análisis a gran señal. Circuitos prácticos con diodos. 4. Dispositivos optoelectrónicos: La célula solar. El diodo emisor de luz.

TEMA V. Transistores

1. Transistor MOS de efecto de campo (MOSFET): Estructura. Funcionamiento básico. Tensión umbral. 2. Ecuaciones características y regiones de operación. 3. Circuitos prácticos con transistores MOS. Introducción a la lógica CMOS.

**Programa detallado en inglés:**

PART 1. Basic concepts of electric parameters and electrical circuits.

Topic I. Electric field. Electric current

1. Coulomb's law and electric field: Electric charge. Coulomb's law. Electric field. 2. Potential energy and electric potential: Electric potential energy. Electric potential and potential difference. 3. Conductors and capacitors: Conductive materials. Capacity and capacitors. 4. Electrical current: Electrical current. Conductivity and Ohm's Law. Joule law. Electrical Power.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Topic II. Magnetic field. Electromagnetic waves.

1. Magnetic field: Magnetic field sources. Magnetic induction. 2. Electromagnetic waves: Waves: the basic concepts. Electromagnetic waves. Photons.

Topic III. Electrical circuits

1. Circuit analysis: Basic definitions: nodes, loops, meshes and branches. Kirchhoff's laws. Circuit components. DC and AC sources. Current-voltage characteristic. Association of components. Basic methods of circuit analysis. Thevenin's theorem. 2. Circuits with time variant signals: Charge and discharge of a capacitor.

PART 2. Semiconductors and electronic devices.

Topic IV. Two terminal devices

1. Introduction to semiconductors: Conductors, insulators and semiconductors. Extrinsic semiconductors. Electrical conductivity in semiconductors. Generation and recombination of charge carriers. Applications. 2. P-n junction: P-n junction in equilibrium. Polarized p-n junction. Characteristic I-V curve of the diode. 3. Junction diodes as circuit components: Bias point and load line. Large-signal analysis. Practical circuits with diodes. 4. Optoelectronic devices: Solar cells. Light emitting diodes.

Topic V. Transistors

1. MOS Field Effect Transistor (MOSFET): Structure and basic operation. Threshold voltage. 2. Characteristic equations and operating modes. 3. Circuits with MOS transistors. Introduction to CMOS logic.

**Competencias de la asignatura:**

**Generales:**

CG5-Comprensión y dominio de los conceptos básicos de campos y ondas y electromagnetismo, teoría de circuitos eléctricos, circuitos electrónicos, principio físico de los semiconductores y familias lógicas, dispositivos electrónicos y fotónicos, y su aplicación para la resolución de problemas propios de la ingeniería.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

**Resultados de aprendizaje:**

Análisis crítico de soluciones. (CT2)

Argumentar las elecciones de aproximaciones físicas relevantes. (CT2)

Comprender y resolver problemas de electromagnetismo básico. (CG5)

Evaluar la eficiencia de los métodos de cálculo para elegir el más adecuado. (CT2)

Resolver circuitos eléctricos lineales y electrónicos basados en diodos y transistores. (CG5)

**Evaluación detallada:**

• Examen final (80%-90%).

• Otras actividades. Realización de controles presenciales y/o online, realización de cuestiones y/o ejercicios en clase y/o propuestos para entregar (10%-20%).

Tanto la calificación final de la convocatoria ordinaria como la de la extraordinaria serán la mayor de las dos siguientes opciones:

$CFINAL = 0.8 CEX + 0.2 CACT$  o  $CFINAL = 0.9 CEX + 0.1 CACT$

CEX = Calificación del examen, sobre una escala de 10

CACT = Calificación de otras actividades, sobre una escala de 10.

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,50

Problemas: 1,50

Laboratorios: 0,00

Otras actividades:

Actividades presenciales:

Clases teóricas con ejemplos de aplicación (30%)

Clases de resolución de problemas (10%)

Actividades dirigidas:

Tutorías y trabajos dirigidos (10%)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**

**FACULTAD DE INFORMATICA**

Trabajo personal:

Estudio, preparación de exámenes, realización de ejercicios. Realización de exámenes. (50%)

**Bibliografía:**

1. P. A. Tipler. Física Volumen 2. Editorial Reverté.

2. T. Ruiz, O. Arbelaitz, I. Etxeberria. Análisis Básico de Circuitos Eléctricos y Electrónicos. Pearson Prentice Hall.

Ficha docente guardada por última vez el 20/06/2022 18:30:00 por el usuario: Vicedecanato de Estudios y Calidad

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805471 - Cálculo		<b>Abrev:</b> CAL	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Calculus		<b>Carácter:</b> Formación básica	
<b>Materia:</b> Matemáticas		24 ECTS	
<b>Otras asignaturas en la misma materia:</b>			
Algebra Lineal		6 ECTS	
Matemática Discreta y Lógica Matemática I		6 ECTS	
Matemática Discreta y Lógica Matemática II		6 ECTS	
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Interdepartamental AMMA/AIGeTo		<b>Coordinador:</b> Dominguez Bonilla, Oscar	

**Descripción de contenidos mínimos:**

- Los números reales.
- Sucesiones y series numéricas.
- Continuidad, derivación e integración de funciones de una variable real.
- Sucesiones y series de funciones.
- Funciones de dos o más variables.
- Cálculo vectorial.
- Interpolación, derivación e integración numérica.

**Programa detallado:**

- El cuerpo ordenado de los números reales.
- Sucesiones de números reales.
- Series de números reales.
- Límites y continuidad de funciones reales de variable real.
- Derivadas de funciones reales de variable real. La regla de la cadena.
- Teoremas del valor medio. La regla de L'Hopital
- Aplicaciones de la derivada. Optimización. Representación de gráficas.
- Aproximación por funciones polinómicas.
- Integrales de funciones reales de variable real.
- Teorema Fundamental del Cálculo.
- Cálculo de primitivas. El Teorema del cambio de variable.
- Descomposición de funciones racionales. Primitivas de funciones racionales.
- Integrales impropias.
- Cálculo de longitudes, áreas y volúmenes de revolución.
- Cálculo con funciones de más de una variable.

**Programa detallado en inglés:**

- Series of real numbers
- Limits and continuity of one real variable functions.
- Derivable one real variable functions.
- Mean value Theorem. L'Hopital's Rule.
- Applications of derivative function. Optimization. Graphs.
- Approximation by polynomial functions.
- Riemann integral of one real variable functions.
- Fundamental theorem of calculus.
- Calculus of antiderivatives. Integration by substitution.
- Integration of rational functions.
- Improper integrals.
- Calculus of lengths, areas and volumes of revolution.
- Calculus of several variables functions.

**Competencias de la asignatura:****Generales:**

- CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.
- CG2-Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Clasificar un número limitado de problemas matemáticos y resolverlos según patrones preestablecidos. (CT2)

Clasificar un problema en función de su solución y resolverlo. (CG1)

Reconocer un problema matemático entre los propuestos en la asignatura. (CG1)

**Evaluación detallada:**

- El 20% de la nota se obtiene en las clases prácticas, evaluadas por el profesor del grupo, siempre que se cumpla la condición de haber asistido y realizado al menos el 80% de ellas (tanto en la convocatoria ordinaria como en la extraordinaria).

- El 80% restante de la nota se consigue en el examen ordinario o extraordinario de la asignatura. Cada examen se corrige de forma horizontal entre los profesores de la asignatura.

Para aprobar la suma de la nota de prácticas (hasta 2 puntos) y la del examen (hasta 8 puntos) debe ser igual o superior a 5 puntos.

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,50

Problemas: 1,50

Laboratorios: 0,00

Otras actividades:

Clases teóricas:

3 horas semanales de pizarras con desarrollos teóricos y resolución de problemas.

Clases prácticas:

1 hora a la semana de prácticas dirigidas realizada por los alumnos.

Ambos tipos de clases presenciales.

**Bibliografía:**

- BARTLE, R. y D. R. SHERBERT, "Introducción al análisis matemático de una variable," Ed. Limusa, 2ª edición (1996).
- BRADLEY, G. Y SMITH, K. : "Cálculo de una y varias variables", Volumen 1. Prentice-Hall.
- GARCÍA, A. y otros "Cálculo I. Teoría y problemas de Análisis Matemático en una variable," Ed. GLAGSA, Madrid 1993.
- Ramos A.M., Rey J.M., "Matemáticas para el acceso a la universidad", Ediciones Pirámide (Grupo ANAYA), 2015.
- SPIVAK, M. "Cálculo infinitesimal," Ed. Reverté (1994).
- STEWART, J., "Cálculo diferencial e integral", Thomson (1999).
- J. San Martín, V. Tomeo y I. Uñas "Cálculo en una variable ", Ed. Garceta, 2010.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805472 - Algebra Lineal		<b>Abrev:</b> AL	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Linear Algebra		<b>Carácter:</b> Formación básica	
<b>Materia:</b> Matemáticas		24 ECTS	
<b>Otras asignaturas en la misma materia:</b>			
Cálculo		6 ECTS	
Matemática Discreta y Lógica Matemática I		6 ECTS	
Matemática Discreta y Lógica Matemática II		6 ECTS	
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Interdepartamental AMMA/AIGeTo		<b>Coordinador:</b> Gallego Rodrigo, Francisco Javier	

**Descripción de contenidos mínimos:**

Sistemas lineales y matrices.  
Programación lineal.  
Espacios vectoriales.  
Aplicaciones lineales.  
Rangos y determinantes.  
Sistemas lineales.  
Resolución numérica de sistemas lineales.  
Diagonalización.  
El espacio euclídeo.  
Estadística descriptiva

**Programa detallado:**

- El cuerpo de los números complejos. Raíces de la unidad.
- Matrices.
- Matrices escalonadas.
- Sistemas de ecuaciones lineales. El método de Gauss
- Espacios vectoriales.
- Subespacios vectoriales.
- Dependencia lineal. Bases.
- Rangos.
- Determinantes.
- El teorema de Rouche-Frobenius.
- Diagonalización de matrices.
- Aplicaciones lineales.
- Núcleo e imagen de una aplicación lineal.
- Forma matricial de una aplicación lineal.
- Semejanzas de Matrices. autovalores y autovectores. Aplicaciones.
- Espacio euclídeo

**Programa detallado en inglés:**

- The fields of complex numbers. Roots of unity.
- Matrices
- Hermite normal form.
- System of linear equations. The method of Gaussian elimination.
- Vector spaces.
- Vector subspaces.
- Linear dependence. Bases.
- Linear maps.
- Kernel, image and the rank-nullity theorem.
- Matrix form of a linear map.
- Ranks.
- Determinants.
- Rouché-Frobenius's Theorem.
- Matrix similarity. Eigenvalues and eigenvectors.
- Diagonalizable matrix. Applications.
- Euclidean space.

**Competencias de la asignatura:****Generales:**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.

**Específicas:**

No tiene

**Básicas y Transversales:**

CB1-Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio

CB2-Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio

CB3-Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética

CB4-Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado

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

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

**Resultados de aprendizaje:**

Clasificar un número limitado de problemas matemáticos y resolverlos según patrones preestablecidos. (CT2)

Clasificar un problema en función de su solución y resolverlo. (CG1)

Reconocer un problema matemático entre los propuestos en la asignatura. (CG1)

**Evaluación detallada:**

- El 20% de la nota se obtiene en las clases prácticas (tanto para la convocatoria ordinaria como en la extraordinaria), evaluadas por el profesor del grupo, siempre que se cumplan las condiciones impuestas por dicho profesor al principio de la asignatura. En cualquier caso, una condición indispensable será la de haber asistido y realizado al menos el 80% de las prácticas.

- El 80% restante de la nota se consigue en el examen ordinario o extraordinario de la asignatura. Si las circunstancias lo permiten, cada examen se corregirá de forma horizontal entre los profesores de la asignatura.

Para aprobar la suma de la nota de prácticas y la del examen debe ser igual o superior a 5 puntos.

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,50

Problemas: 1,50

Laboratorios: 0,00

Otras actividades:

Clases teóricas:

3 horas semanales de pizarras con desarrollos teóricos y resolución de problemas.

Clases prácticas:

1 hora a la semana de prácticas dirigidas realizada por los alumnos.

Ambos tipos de clases presenciales.

**Bibliografía:**

Los siguientes son los libros de texto recomendados para la asignatura, aunque también se hará uso de textos, documentos y material audiovisual proporcionados a los estudiantes a través del Campus Virtual.

- BARO, E. y TOMEO, V.; "Introducción al Álgebra Lineal"; Ed. Garceta, 2014.

- DE BURGOS, J.; "Álgebra lineal y geometría cartesiana"; Segunda edición, McGraw-Hill, Madrid, 2000.

- HERNÁNDEZ, E.; "Álgebra y Geometría"; Addison-Wesley y Ed. U.A.M., 1994.

- MERINO, L. y SANTOS, E.; "Álgebra lineal con métodos elementales"; Ed. Thomson-Paraninfo, Madrid 2006 y Ed. Univ. de Granada, Granada, 1997;

- STRANG, G., "Álgebra Lineal y sus Aplicaciones", Ed. Addison-Wesley Iberoamericana, 1986.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



# UNIVERSIDAD COMPLUTENSE DE MADRID

## FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805473 - Matemática Discreta y Lógica Matemática I		<b>Abrev:</b> MDL1	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Discrete Mathematics and Mathematical Logic I		<b>Carácter:</b> Formación básica	
<b>Materia:</b> Matemáticas		<b>24 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>			
Algebra Lineal		6 ECTS	
Cálculo		6 ECTS	
Matemática Discreta y Lógica Matemática II		6 ECTS	
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Fernández Camacho, M. Inés	

### Descripción de contenidos mínimos:

Introducción a los métodos de razonamiento.  
Inducción y recursión.  
Teoría de números.  
Conjuntos y funciones.  
Relaciones y órdenes.  
Recurrencias.

### Programa detallado:

Tema 1: Introducción.  
Lógica proposicional: introducción lenguaje, tablas de verdad. Lógica de primer orden. Introducción al lenguaje de la lógica de primer orden (cuantificadores). Formalización de enunciados. Métodos de demostración: reducción al absurdo, contraejemplos, demostraciones universales.

Tema 2: Números, Inducción, recursión. Conjuntos numéricos, División entera, divisibilidad, números primos. Inducción. Definiciones recursivas y recurrencias.

Tema 3: Conjuntos, relaciones, funciones y cardinales. Conjuntos, elementos y subconjuntos, operaciones con conjuntos. Relaciones y propiedades. Funciones y propiedades. Cardinales.

Tema 4: Relaciones de equivalencia: clases de equivalencia. Relaciones de orden: conjuntos ordenados.

### Programa detallado en inglés:

Unit 1: Introduction.  
Propositional logic: language introduction, truth tables. First-order logic. Introduction to the language of first-order logic (quantifiers). Formalization of statements. Methods of proof: reductio ad absurdum, counterexamples, universal proofs.

Unit 2: Numbers, induction, recursion. Sets of numbers, integer division, divisibility, prime numbers. Induction. Recursive definitions and recurrences.

Unit 3: Sets, relations, functions and cardinality. Sets, elements and subsets, set operations. Relations and properties. Functions and properties. Cardinality.

Unit 4: Equivalence relations: equivalence classes. Order relations: ordered sets.

### Competencias de la asignatura:

#### Generales:

CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.

CG2-Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.

#### Específicas:

No tiene

#### Básicas y Transversales:

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Aprender a demostrar por inducción y a definir utilizando recursión. (CG1, CG2)

Conocer las nociones básicas de teoría de conjuntos. (CG1, CG2)

Ser capaz de aplicar los conceptos y técnicas aprendidos en el contexto de otras asignaturas. (CT2)

Combinar el uso de técnicas en la resolución de problemas. (CT3)

Realizar ejercicios. (CT1)

**Evaluación detallada:**

La nota final se calculará sumando la nota obtenida en el examen final (ya sea en la convocatoria ordinaria o en la extraordinaria), cuyo valor máximo es de 9 puntos, y la nota obtenida en otras actividades (valor máximo: 1 punto).

La nota del apartado de otras actividades se obtiene durante la convocatoria ordinaria y no es recuperable durante la convocatoria extraordinaria.

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,50

Problemas: 1,50

Laboratorios: 0,00

Otras actividades:

Actividades presenciales: Clases teóricas y clases prácticas 40%

Actividades dirigidas: Realización individual de ejercicios y problemas tutorizados 10%

Trabajo personal: 50%

**Bibliografía:**

R. Caballero, T. Hortalá, N. Martí, S. Nieva, A. Pareja, M. Rodríguez; Matemática Discreta para Informáticos. Ejercicios resueltos; Pearson, Colección Prentice Practica, 2007;

R.P. Grimaldi; Matemáticas discreta y Combinatoria. Una introducción con aplicaciones. Pearson Educación, 3ª edición, 1998

M. T. Hortalá González, J. Leach Albert, M. Rodríguez Artalejo; Matemática Discreta y Lógica Matemática; Editorial Complutense, 2001 (Segunda edición);

K.H. Rosen; Discrete Mathematics and Its Applications; McGraw-Hill, 2012 (Seventh Edition); (También en castellano: K.H. Rosen;

Matemática discreta y sus aplicaciones 5ª edición. McGraw-Hill, 2004)

K.A.Ross, C.R.B. Wright; Discrete Mathematics; Prentice Hall 1992 (Third Edition);

M. Ben-Ari; Mathematical Logic for Computer Science; Springer 2001 (Second Edition);

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:

**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805474 - Matemática Discreta y Lógica Matemática II		<b>Abrev:</b> MDL2	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Discrete Mathematics and Mathematical Logic II		<b>Carácter:</b> Formación básica	
<b>Materia:</b> Matemáticas		<b>24 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>			
Algebra Lineal		6 ECTS	
Cálculo		6 ECTS	
Matemática Discreta y Lógica Matemática I		6 ECTS	
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Palomino Tarjuelo, Miguel	

**Descripción de contenidos mínimos:**

Métodos de razonamiento.  
Formalización y deducción en lógica de proposiciones y de primer orden.  
Combinatoria.  
Grafos y árboles.

**Programa detallado:**

Tema 1: Grafos y árboles.  
Grafos no dirigidos y multigrafos. Recorridos en grafos: ciclos hamiltonianos, recorridos eulerianos. Árboles. Grafos dirigidos.

Tema 2: Combinatoria.  
Principios elementales de conteo, variaciones, permutaciones y combinaciones.

Tema 3: Lógica de proposiciones.  
Sintaxis y semántica. Validez, equivalencia y consecuencia lógica. Formas normales. Sistemas de deducción: tableaux.

Tema 4: Lógica de primer orden.  
Sintaxis y semántica. Validez, equivalencia y consecuencia lógica. Formas normales. Sistemas de deducción: tableaux.

**Programa detallado en inglés:**

Unit 1: Graphs and trees. Undirected graphs and multigraphs. Paths: Hamilton and Euler paths. Trees. Directed graphs.

Unit 2: Counting. Basic counting techniques, variations, permutations and combinations.

Unit 3: Propositional logic. Syntax and Semantics. Validity, logical equivalence and logical consequence. Normal forms. Semantic tableaux.

Unit 4: First-order logic. Syntax and Semantics. Validity, logical equivalence and logical consequence. Normal forms. Semantic tableaux.

**Competencias de la asignatura:****Generales:**

CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.

CG2-Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Conocer los principios elementales de conteo. (CG1, CG2)

Resolver problemas elementales sobre grafos. (CG1, CG2)

Utilizar las lógicas proposicional y de primer orden para formalizar y demostrar argumentaciones. (CG1, CG2)

Ser capaz de aplicar los conceptos y técnicas aprendidos en el contexto de otras asignaturas. (CT2)

Combinar el uso de técnicas en la resolución de problemas. (CT3)

Realizar ejercicios. (CT1)

**Evaluación detallada:**

La nota final se calculará sumando la nota obtenida en el examen final (ya sea en la convocatoria ordinaria o en la extraordinaria), cuyo valor máximo es de 8,5 puntos, y la nota obtenida en otras actividades (valor máximo: 1,5 puntos).

La nota del apartado de otras actividades se obtiene durante la convocatoria ordinaria y no es recuperable durante la convocatoria extraordinaria.

**Actividades docentes:**

Reparto de créditos:

Teoría: 5,00

Problemas: 1,00

Laboratorios: 0,00

Otras actividades:

Actividades presenciales: Clases teóricas y clases prácticas 40%

Actividades dirigidas: Realización individual de ejercicios y problemas tutorizados 10%

Trabajo personal: 50%

**Bibliografía:**

M. T. Hortalá González, J. Leach Albert, M. Rodríguez Artalejo; Matemática Discreta y Lógica Matemática; Editorial Complutense, 2001 (Segunda edición).

R. Caballero, T. Hortalá, N. Martí, S. Nieva, A. Pareja, M. Rodríguez; Matemática Discreta para Informáticos. Ejercicios resueltos; Pearson, Colección Prentice Practica, 2007.

T. Hortalá, N. Martí, M. Palomino, M. Rodríguez, R. del Vado.; Lógica Matemática para Informaticos. Ejercicios resueltos; Pearson, Colección Prentice Practica, 2008.

K.H. Rosen; Discrete Mathematics and Its Applications; McGraw-Hill, 2003 (Fifth Edition).

K.A.Ross, C.R.B. Wright; Discrete Mathematics; Prentice Hall 1992 (Third Edition).

M. Ben-Ari; Mathematical Logic for Computer Science; Springer 2001 (Second Edition).

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805475 - Fundamentos de la Programación I		<b>Abrev:</b> FP1	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Fundamentals of Programming I		<b>Carácter:</b> Formación básica	
<b>Materia:</b> Informática		24 ECTS	
<b>Otras asignaturas en la misma materia:</b>			
Fundamentos de Computadores I		6 ECTS	
Fundamentos de Computadores II		6 ECTS	
Fundamentos de la Programación II		6 ECTS	
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Interdepartamental ISIA / SIC		<b>Coordinador:</b> Hernández Yáñez, Luis	

**Descripción de contenidos mínimos:**

Construcciones básicas de la programación estructurada.  
Abstracciones procedimentales.  
Tipos de datos estructurados.  
Archivos de texto.  
Uso de entornos de programación y desarrollo.  
Documentación, prueba y depuración de programas.  
Realización de prácticas en laboratorio.

**Programa detallado:**

- 1.- Computadoras y programación
- 2.- Tipos simples e instrucciones I
- 3.- Tipos simples e instrucciones II
- 4.- La abstracción procedimental
- 5.- Tipos de datos estructurados

**Programa detallado en inglés:**

- 1.- Computers and Programming
- 2.- Simple Types and Instructions I
- 3.- Simple Types and Instructions II
- 4.- Procedural Abstraction
- 5.- Structured Data Types

**Competencias de la asignatura:****Generales:**

CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CB1-Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio
- CB2-Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio
- CB3-Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética
- CB4-Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado
- CB5-Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Desarrollar las prácticas y ejercicios tomando decisiones sobre su diseño. (CT3)
- Desarrollar y validar programas expresados en lenguajes de programación concretos. (CG3)
- Evaluar la eficiencia de los algoritmos para elegir el más adecuado. (CT2)
- Manejar en los programas desarrollados datos estructurados mantenidos en archivos. (CG3)
- Resolver ejercicios de programación analizando el problema y diseñando la solución. (CT2, CT3)
- Utilizar herramientas informáticas sobre sistemas operativos concretos para desarrollar programas informáticos. (CG3)

**Evaluación detallada:**

La calificación se compondrá de varias calificaciones parciales de las distintas tareas llevadas a cabo por el estudiante.  
A continuación se indican los porcentajes del peso de cada actividad en la calificación final en ambas convocatorias (ordinaria y extraordinaria):  
- Examen final (ordinario/extraordinario): 60%  
- Práctica: 20%  
- Actividades adicionales: 20% (a determinar por cada profesor)  
Se realizará 1 práctica.

Para poder aprobar en la convocatoria ordinaria se requerirá al menos una calificación de 5 sobre 10 en el examen final y haber aprobado la práctica (práctica aprobada = entregada en el plazo establecido durante el curso, satisfaciendo los requisitos establecidos en el enunciado y evaluada al menos con un 5 sobre 10).

En la convocatoria extraordinaria la evaluación se realizará de la misma forma, habiendo un plazo adicional para entregar la práctica corregida si se suspendió en la convocatoria ordinaria.

Dado que las actividades adicionales se realizan a lo largo del cuatrimestre, su calificación es usada en ambas convocatorias, sin que haya posibilidad de recuperarlas.

**Actividades docentes:**

Reparto de créditos:

- Teoría: 3,00
- Problemas: 0,00
- Laboratorios: 3,00

Otras actividades:

- Clases teóricas: 2 horas de clases teóricas a la semana.
- Clases prácticas: 2 horas de clases de problemas/prácticas a la semana. Las clases de problemas/prácticas se desarrollarán en aulas de informática.
- Las actividades presenciales se corresponden con las 4 horas semanales en aula.

**Bibliografía:**

- "C++: An Introduction to Computing". 2ª edición. J. Adams, S. Leestma, L. Nyhoff. Prentice Hall, 1998.
- "Programación y resolución de problemas con C++". N. Dale, C. Weems. McGraw-Hill, 2007.
- "Programación en C++ para ingenieros". F. Xhafa et al. Thomson, 2006.
- "Programming: Principles and Practice using C++". B. Stroustrup. Pearson/Addison-Wesley, 2009.
- "El lenguaje de programación C++". Edición especial. B. Stroustrup. Pearson Educación, 2002.
- "Resolución de problemas con C++". 2ª edición. W. Savitch. Pearson, 1999.
- "Problem Solving, Abstraction, Design Using C++". 3ª edición. F.L. Friedman, E.B. Koffman. Addison-Wesley, 2000.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805476 - Fundamentos de la Programación II		<b>Abrev:</b> FP2	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Fundamentals of Programming II		<b>Carácter:</b> Formación básica	
<b>Materia:</b> Informática		24 ECTS	
<b>Otras asignaturas en la misma materia:</b>			
Fundamentos de Computadores I		6 ECTS	
Fundamentos de Computadores II		6 ECTS	
Fundamentos de la Programación I		6 ECTS	
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Interdepartamental ISIA / SIC		<b>Coordinador:</b> Hernández Yáñez, Luis	

**Descripción de contenidos mínimos:**

Recursión.  
Tipos de datos estructurados.  
Punteros y memoria dinámica.  
Programación modular.  
Uso de entornos de programación y desarrollo.  
Documentación, prueba y depuración de programas.  
Realización de prácticas en laboratorio.

**Programa detallado:**

- 1.- Tipos de datos estructurados: Arrays multidimensionales
- 2.- Programación Modular
- 3.- Algoritmos de ordenación y gestión de datos ordenados
- 4.- Punteros y memoria dinámica
- 5.- Introducción a la recursión

**Programa detallado en inglés:**

1. Structured Data Types: Multidimensional arrays
2. Modular Programming
3. Sorting Algorithms and ordered data management
4. Pointers and Dynamic Memory
5. Introduction to Recursion

**Competencias de la asignatura:****Generales:**

CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Desarrollar las prácticas y ejercicios tomando decisiones sobre su diseño. (CT3)

Desarrollar y validar programas expresados en lenguajes de programación concretos. (CG3)

Evaluar la eficiencia de los algoritmos para elegir el más adecuado. (CT2)

Manejar en los programas desarrollados datos estructurados mantenidos en archivos. (CG3)

Resolver ejercicios de programación analizando el problema y diseñando la solución. (CT2, CT3)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Utilizar herramientas informáticas sobre sistemas operativos concretos para desarrollar programas informáticos. (CG3)

**Evaluación detallada:**

La calificación se compondrá de varias calificaciones parciales de las distintas tareas llevadas a cabo por el estudiante.

A continuación se indican los porcentajes del peso de cada actividad en la calificación final en ambas convocatorias (ordinaria y extraordinaria):

- Examen final (ordinario/extraordinario): 65%

- Práctica: 20%

- Actividad adicional: 15% (a determinar por cada profesor)

Las calificaciones de la práctica y de la actividad adicional serán aquellas con las que se haya llegado a la convocatoria ordinaria, no pudiéndose recuperar en la extraordinaria.

Para poder aprobar en la convocatoria ordinaria y extraordinaria se requerirá al menos una calificación de 5 sobre 10 en el examen final.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 1,50

Laboratorios: 1,50

Otras actividades:

Clases teóricas: 2 horas de clases teóricas a la semana.

Clases prácticas: 2 horas de clases de problemas/prácticas a la semana.

Las clases de problemas/prácticas se desarrollarán en aulas de informática.

Las actividades presenciales se corresponden con las 4 horas semanales en aula.

**Bibliografía:**

"Programación y resolución de problemas con C++". N. Dale, C. Weems. McGraw-Hill, 2007.

"Programación en C++ para ingenieros". F. Xhafa et al. Thomson, 2006.

"Programming: Principles and Practice using C++". B. Stroustrup. Pearson/Addison-Wesley, 2009.

"El lenguaje de programación C++". Edición especial. B. Stroustrup. Pearson Educación, 2002.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 1º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805477 - Fundamentos de Computadores I		<b>Abrev:</b> FC1	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Introduction to Computers I		<b>Carácter:</b> Formación básica	
<b>Materia:</b> Informática		24 ECTS	
<b>Otras asignaturas en la misma materia:</b>			
Fundamentos de Computadores II		6 ECTS	
Fundamentos de la Programación I		6 ECTS	
Fundamentos de la Programación II		6 ECTS	
<b>Módulo:</b> Materias básicas			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Hidalgo Pérez, José Ignacio	

**Descripción de contenidos mínimos:**

- Representación de la información.
- Especificación e implementación de sistemas combinacionales.
- Módulos combinacionales básicos.
- Especificación e implementación de sistemas secuenciales.
- Módulos secuenciales básicos.
- Prácticas de diseño de circuitos combinacionales y secuenciales.

**Programa detallado:**

## Tema1. Representación digital de la información

- Introducción y conceptos.
- Sistemas de numeración: binario, octal, hexadecimal.
- Aritmética binaria. Suma, resta, multiplicación, división. Conversión entre bases: división por la base, sustitución en serie, conversión entre potencias de la misma base.
- Representación de números: binario puro, magnitud y signo, complemento a uno, complemento a dos.
- Aritmética entera: magnitud y signo, complemento a dos.
- Otras codificaciones: representaciones decimales: BCD y exceso-3.

## Tema 2: especificación de sistemas combinacionales

- Especificación de alto nivel y binaria.
- Codificación.
- Funciones de conmutación. Tablas de verdad.
- Expresiones de conmutación.
- Algebra de Moore. Transformaciones algebraicas.
- Forma canónica. Suma de productos. Transformación de una expresión de conmutación a una suma de productos canónica.
- Mapas de Karnaugh. Simplificación. Otros usos de los mapas de Karnaugh

## Tema 3: Implementación de sistemas combinacionales

- Puertas lógicas. And, or, not, nand, nor, xor, nxor. Propiedades de las puertas lógicas: conmutividad, asociatividad. Puertas lógicas compuestas.
- Conjuntos universales de puertas: definición. Conjunto and-or-not. Conjunto and-not. Conjunto nand.
- Síntesis de redes de puertas. Implementación de dos niveles. Implementación multinivel. Síntesis con puertas and-or-not.
- Síntesis con puertas nand.
- Análisis de redes de puertas. Análisis de redes and-or. Análisis de redes Nand.

## Tema 4: Módulos combinacionales clásicos.

- Decodificador. Implementación directa. Implementación en árbol. Aplicaciones al diseño.
- Multiplexor. Implementación directa. Implementación en árbol. Multiplexor vectorial.
- Bus
- Codificador. Implementación directa. Codificador de prioridad. Implementación directa. Implementación en árbol. Implementación en cascada. Aplicaciones al diseño.
- ROM (Read Only Memory). PROM. EPROM. EEPROM. Flash memory Aplicaciones al diseño.
- Sumador: implementación con propagación de acarreo: full adder
- Restador: implementación con propagación de acarreo: full subtractor
- Comparador de magnitud. Implementación en serie. Implementación en árbol.
- Sumador/restador en complemento a dos. Implementación con propagación de acarreo.
- ALU (Arithmetic-Logic Unit)

## Tema 5: Especificación de sistemas secuenciales

- Definiciones.
- Especificación basada en estados.
- Diagramas de estados.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Descripción binaria.
  - Asíncronos vs síncronos.
  - Concepto de registro de estado.
  - Máquinas de Moore y Mealy.
  - Reconocedor de patrones.
- Tema 6: Implementación de sistemas secuenciales síncronos.
- Biestables SR. Asíncronos, síncronos por nivel, síncronos por flanco.
  - Biestables D. síncronos por nivel, síncronos por flanco.
  - Síntesis con biestables D.
  - Inicialización de sistemas secuenciales.
- Tema 7: Módulos secuenciales básicos
- Registros
  - Registros de desplazamiento.
  - Contadores. Ascendente, descendente, saturado no saturado. contador de carga paralela. Diseño con contadores de carga paralela.
- Aplicaciones al diseño: generadores de secuencia y patrones.
- Banco de registros. Diseño y aplicación.
  - RAM: SRAM, DRAM, SDRAM.

**Programa detallado en inglés:**

1. Representing digital information
  - Introduction and concepts.
  - Numbering systems: binary, octal, hexadecimal.
  - Binary arithmetic Addition subtraction multiplication division. Conversion between bases: division by the base, substitution in series, conversion between powers of the same base.
  - Representation of numbers: pure binary, magnitude and sign, complement to one, complement to two.
  - Whole arithmetic: magnitude and sign, complement two.
  - Other encodings: decimal representations: BCD and excess-3.
2. Specification of combinational systems
  - High level and binary specification.
  - Coding.
  - Switching functions. Truth tables.
  - Switching expressions.
  - Moore's Algebra. Algebraic transformations.
  - Canonical form. Sum of products. Transformation of a switching expression to a canonical sum of products.
  - Karnaugh maps. Simplification. Other uses of Karnaugh maps
3. Implementation of combinational systems
  - Logical doors. And, or, not, nand, nor, xor, nxor. Properties of logic doors: commutability, associativity. Composite logic doors.
  - Universal door assemblies: definition. Set and-or-not. Set and-not. Nand set
  - Synthesis of door networks. Implementation of two levels. Multilevel implementation. Synthesis with and-or-not doors.
  - Synthesis with nand doors.
  - Analysis of door networks. Analysis of and-or networks. Analysis of Nand networks.
4. Basic combinational modules
  - Decoder. Direct implementation Implementation in tree. Applications to design.
  - Multiplexer. Direct implementation Implementation in tree. Vector multiplexer.
  - Bus
  - Encoder. Direct implementation Priority encoder. Direct implementation Implementation in tree. Implementation in cascade. Applications to design.
  - ROM (Read Only Memory). PROM. EPROM EEPROM. Flash memory Applications to the design.
  - Adder: implementation with propagation of shipments: full adder
  - Subtract: implementation with propagation of carries: full subtractor
  - Comparator of magnitude. Implementation in series. Implementation in tree.
  - Adder / subtractor in two's complement. Implementation with transport propagation.
  - ALU (Arithmetic-Logic Unit)
5. Specification of sequential systems
  - Definitions.
  - Specification based on states.
  - State diagrams.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



- Binary description.
  - Asynchronous vs. synchronous.
  - Concept of state registration.
  - Moore and Mealy machines.
  - Pattern recognizer.
6. Implementation of synchronous sequential systems
- SR flip-flops Asynchronous, synchronous by level, synchronous by flank.
  - D. flip-flops synchronous by level, synchronous by flank.
  - Synthesis with bistables D.
  - Initialization of sequential systems.
7. Basic sequential modules
- Records
  - Scroll records.
  - Accountants. Ascending, descending, saturated not saturated. parallel load counter. Design with parallel load counters. Applications to design: sequence generators and patterns.
  - Bank of records. Design and application
  - RAM: SRAM, DRAM, SDRAM.

**Competencias de la asignatura:**

**Generales:**

CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Diseñar sistemas digitales. (CG4)

Justificar las decisiones de diseño aplicadas en prácticas y ejercicios. (CT1)

Resolver problemas de diseño digital seleccionando la solución más adecuada entre las posibles. (CT2, CT3)

**Evaluación detallada:**

Evaluación detallada:

Nota de clase: problemas, participación, pruebas de clase o test (NPro)

Nota de Prácticas (NPra)

Examen (NExa)

Es la mayor de las dos puntuaciones siguientes:

$NExa * 0,70 + NPra * 0,20 + NPro * 0,10$

$NExa * 0,80 + NPra * 0,20$

La nota de problemas y la nota de las prácticas es la obtenida en la convocatoria de ordinaria, es decir, no hay posibilidad de recuperar/mejorar esa calificación en la convocatoria extraordinaria

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,50

Problemas: 1,50

Laboratorios: 1,00

Otras actividades:

No tiene

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

**Bibliografía:**

Digital Design and Computer Architecture ARM Edition. Sarah Harris, David Harris. Morgan Kaufmann. 2015  
Principios de Diseño Digital, Daniel D. Gajski Ed. Prentice Hall, 1997  
Fundamentos de Computadores, Hermida, R., Sánchez, F., Pastor, E. del Corral, A.M., Ed. Síntesis. 1998.

Ficha docente guardada por última vez el 28/06/2023 9:23:00 por el usuario: Coordinador GIC

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> 1º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805478 - Fundamentos de Computadores II	<b>Abrev:</b> FC2	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Introduction to Computers II	<b>Carácter:</b> Formación básica	
<b>Materia:</b> Informática	<b>24 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>		
Fundamentos de Computadores I	6 ECTS	
Fundamentos de la Programación I	6 ECTS	
Fundamentos de la Programación II	6 ECTS	
<b>Módulo:</b> Materias básicas		
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Lanchares Dávila, Juan

**Descripción de contenidos mínimos:**

- Introducción a la estructura de un computador.
- Instrucciones del computador.
- Diseño de un computador sencillo.
- Programación en lenguaje ensamblador de un computador simple.
- Prácticas de ensamblador.

**Programa detallado:**

- Lenguaje máquina y ensamblador.
  - ¿Qué es un computador?
  - Arquitectura básica de un procesador RISC-V
  - Lenguaje ensamblador
  - Programación en ensamblador
- Diseño del procesador
  - Diseño de un procesador: monociclo, multiciclo y segmentado
  - Rendimiento
- Sistema de memoria de un computador.
  - Organización de la memoria
  - Jerarquía de memoria
  - Introducción a la memoria cache

**Programa detallado en inglés:**

- Machine and assembly language.
  - What is a computer?
  - Basic architecture of a processor RISC-V
  - Assembly language
  - Assembly programming
- Processor design
  - The design of a processor: Monocycle, Multicycle, Pipelined
  - Performance
- Memory system of a computer.
  - Memory organization
  - Memory hierarchy
  - Introduction to cache memory

**Competencias de la asignatura:****Generales:**

CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

**Específicas:**

No tiene

**Básicas y Transversales:**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Analizar el funcionamiento y la estructura básica de un computador. (CG4)

Desarrollar programas en ensamblador. (CG4)

Justificar las decisiones de diseño aplicadas en prácticas y ejercicios. (CT1)

**Evaluación detallada:**

- Nota de clase: problemas, participación, pruebas de clase o test (NPro)
- Nota de Prácticas (NPra)
- Examen (NExa)

Es la mayor de las dos puntuaciones siguientes:

- NExa \* 0,70 + NPra \* 0,20 + NPro\* 0,10

- NExa \* 0,80 + NPra\* 0,20

La nota de problemas y la nota de las prácticas es la obtenida en la convocatoria de ordinaria, es decir, no hay posibilidad de recuperar/mejorar esa calificación en la convocatoria extraordinaria

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,50

Problemas: 1,50

Laboratorios: 1,00

Otras actividades:

No tiene

**Bibliografía:**

Digital Design and Computer Architecture RISC-V Edition. Sarah Harris, David Harris. Morgan Kaufmann. 2021

Computer Organization and Design RISC-V Edition: The hardware/software Interface (2nd edition). D. A. Patterson and J. L. Hennessy.

Morgan Kaufmann. 2020

Estructura y diseño de computadores. La interfaz hardware/software (2da. edición). David A. Patterson & John L. Hennessy, Editorial Reverté. 2011

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803207 - Estructura de computadores		<b>Abrev:</b> EC	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Computer Organization		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Estructura de computadores		<b>6 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> No hay			
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Molina Prego, Mª Carmen	

**Descripción de contenidos mínimos:**

- Repertorio de instrucciones.
- El procesador: diseño de la ruta de datos y del control.
- Segmentación.
- La jerarquía de memoria: caches, memoria principal y virtual.
- Buses, Entrada/ salida y almacenamiento.
- Prácticas ensamblador, uso eficiente de la jerarquía y entrada/salida.

**Programa detallado:**

## 1. Entrada/salida

- Sistema de E/S: Estructura y función.
- Sistema de interconexión.
- E/S mediante interrupciones.
- E/S por DMA.

## 2. Procesador segmentado

- Repertorio instrucciones RISC-V.
- Segmentación.
- Operaciones multi-ciclo.
- Paralelismo a nivel de instrucción.
- Planificación de instrucciones.
- Ejecución especulativa y técnicas de predicción de saltos.
- Gestión de excepciones.
- Rendimiento.

## 3. Memoria cache

- Jerarquía de memoria.
- Memoria cache.
- Gestión de memoria cache.
- Rendimiento de la memoria cache.
- Optimización de la memoria cache.

## 4. Memoria virtual

- Memoria principal.
- Memoria virtual.
- Gestión de memoria virtual.
- Interacción memoria virtual - memoria cache.

## 5. IEEE754

- Precisión simple
- Precisión doble
- Operaciones básicas

**Programa detallado en inglés:**

## 1. Input/Output

- I/O system: Structure and function.
- Interconnection system.
- Interrupts.
- DMA.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



2. Processor Design
- RISC-V instruction set architecture.
  - Pipelining.
  - Multi-cycle operations.
  - Instruction-level parallelism.
  - Instruction scheduling.
  - Speculative execution and branch prediction techniques.
  - Exception handling.
  - Performance.

3. Cache memory
- Memory hierarchy.
  - Cache memory.
  - Cache memory management.
  - Cache memory performance.
  - Cache memory optimization

4. Virtual memory
- Main memory.
  - Virtual memory.
  - Virtual memory management.
  - Virtual memory - cache memory interaction.

5. IEEE754
- Single precision.
  - Double precision.
  - Basic arithmetic operations.

**Competencias de la asignatura:**

**Generales:**

- CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.
- CG14-Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

- Analizar la estructura, organización y funcionamiento de un computador digital a nivel hardware. (CG4, CG14)
- Analizar la organización jerárquica de las unidades de memoria de un computador. (CG14)
- Analizar la repercusión de la arquitectura del repertorio de instrucciones sobre el rendimiento y las facilidades de programación. (CG4)
- Aplicar las técnicas de segmentación interna de las instrucciones para acelerar el rendimiento de un computador. (CG14)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Comprender los procedimientos de entrada/salida de un computador. (CG14)
- Conocer la arquitectura general de los centros de procesos de datos que dan soporte a la computación en nube. (CG14)
- Conocer la estructura básica de los procesadores paralelos y analizar las técnicas de programación correspondientes. (CG14)
- Conocer las diferentes técnicas de Entrada/Salida y evaluarlas mediante el diseño de prácticas en lenguaje nativo y de alto nivel (CG14)
- Justificar las decisiones de diseño aplicadas en prácticas y ejercicios. (CT1)
- Resolver problemas de Estructura de Computadores seleccionando la solución más adecuada entre las posibles. (CT2, CT3)
- Utilizar el lenguaje nativo de un computador digital para codificar programas. (CG4)

**Evaluación detallada:**

Dos itinerarios de evaluación:

1) Examen final sobre la materia 60%. Media ponderada de las prácticas a lo largo del curso 20%. Otras actividades en el aula 20%. Asistencia mínima a las clases de teoría 66% para poder sumar el 20% de otras actividades (en otro caso se calificará esta parte con un 0).

2) Examen final sobre la materia 60%. Media ponderada de las prácticas a lo largo del curso 20%.

En ambos itinerarios la nota de las prácticas y otras actividades en el aula es la obtenida en la convocatoria ordinaria, es decir, no hay posibilidad de recuperar/mejorar esa calificación en la convocatoria extraordinaria.

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,00

Problemas: 0,75

Laboratorios: 1,25

Otras actividades:

Clases teóricas

en promedio 3 horas a la semana. Incluye teoría y problemas (en aula)

Laboratorios

en promedio 1 hora a la semana (en laboratorio)

**Bibliografía:**

- . Patterson, D. A., Hennessy, J. L., & Hennessy, J. L. (2017). Computer organization and design: the hardware-software interface: risc-v edition (Ser. The morgan kaufmann series in computer architecture and design). Morgan Kaufmann.
- . Hennessy, J. L., Patterson, D. A., & Asanovic, K. (2019). Computer architecture: a quantitative approach (6th ed.). Morgan Kaufmann.
- . Harris, S. H. D. (2022). Digital design and computer architecture: risc-v edition. Morgan Kaufmann.
- . Stallings, W. (2016). Computer organization and architecture: designing for performance (10th ed.). Pearson.
- . Cuesta, A., Hidalgo, J.I., Lanchares, J., Risco, J.L. (2009). Problemas de fundamentos y estructura de computadoras. Pearson.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:

**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803213 - Bases de datos		<b>Abrev:</b> BD	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Databases		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Desarrollo del software fundamental			<b>15 ECTS</b>
<b>Otras asignaturas en la misma materia:</b> Ingeniería del Software			<b>9 ECTS</b>
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Interdepartamental ISIA / SIC		<b>Coordinador:</b> Correas Fernández, Jesús	

**Descripción de contenidos mínimos:**

Modelos de datos.  
Lenguajes de acceso a bases de datos.  
Diseño de bases de datos relacionales.  
Transacciones y control de la concurrencia.  
Conexión a bases de datos.  
Configuración y gestión de SGBD.

**Programa detallado:**

1. Introducción a las bases de datos.
2. Diseño conceptual: modelo entidad-relación.
3. Diseño lógico: modelo relacional. Álgebra relacional.
4. SQL: Structured Query Language.
5. Introducción a PL/SQL. Disparadores.
6. Introducción a transacciones y control de concurrencia.
7. Conceptos avanzados.

**Programa detallado en inglés:**

1. Introduction to Databases.
2. Conceptual Design: The Entity Relationship Model.
3. Logical Design: The Relational Database Model. Relational Algebra.
4. SQL: Structured Query Language.
5. Introduction to PL/SQL. Triggers.
6. Introduction to Transactions and Concurrency Control.
7. Advanced Concepts.

**Competencias de la asignatura:****Generales:**

- CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- CG17-Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.
- CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

**Resultados de aprendizaje:**

- Analizar y crear instrucciones SQL para la manipulación, definición y el control de una base de datos en Sistema de Gestión de Bases de Datos Relacional. (CG17, CG18, CT2)
- Conocer y saber utilizar mecanismos de gestión de la integridad de los datos en los Sistemas de Gestión de Bases de Datos Relacional. (CG3, CG17)
- Desarrollar aplicaciones software básicas que integren un Sistema de Gestión de Bases de Datos Relacional. (CG3, CG18)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Diseñar una Base de Datos según el modelo entidad-relación. (CG17, CT2)

Implementar un diseño de una Base de Datos en un Sistema de Gestión de Bases de Datos Relacional. (CG3, CG17, CT2)

Realizar tareas de administración básica de un Sistema de Gestión de Bases de Datos Relacional. (CG3, CG17, CG18)

**Evaluación detallada:**

Examen final en las convocatorias ordinaria y extraordinaria común a todos los grupos de la asignatura. Para poder aprobar la asignatura se requerirá al menos una calificación de 5 sobre 10 en el examen final.

Otras actividades. Estará formado por al menos dos ejercicios evaluables sobre los temas de la asignatura a determinar por cada docente de forma independiente. Los alumnos conocerán a principio de curso los detalles de los tipos de actividades y valoraciones asociadas a las mismas. Las calificaciones obtenidas por el alumno durante el curso se mantendrán para la convocatoria extraordinaria.

La calificación final de la asignatura en cualquiera de las dos convocatorias es el máximo entre:

$$F*0,7 + A*0,3$$

$$F*0,9 + A*0,1$$

donde F es la nota del examen final y A es la nota de otras actividades.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Clases teóricas de exposición de los principales conceptos teóricos de la asignatura y resolución de casos prácticos. Clases prácticas y de Laboratorio, incluyendo trabajos tanto individuales como en equipo.

**Bibliografía:**

Silberschatz, H. F. Korth, S. Sudarshan. Fundamentos de bases de datos (6ª ed), McGraw-Hill, 2014 (Database Systems Concepts, 7th ed, 2019).

R. Elmasri, S.B. Navathe. Fundamentals of Database Systems (7ª ed.). Addison-Wesley, 2015.

H. Garcia Molina, J.D.Ulman, J. Widom. Database Systems: The Complete Book (2ª Ed). Prentice Hall, 2009.

J. Gallibaud; Oracle 11g -SQL, PL/SQL y SQL\*Plus. Ediciones ENI, 2010.

O. Heurtel. Oracle 11g -Administracion. Ediciones ENI, 2010.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803216 - Electrónica		<b>Abrev:</b> EL	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Electronics		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Tecnología y Arquitectura de Computadores			<b>30 ECTS</b>
<b>Otras asignaturas en la misma materia:</b>			
Arquitectura de Computadores			6 ECTS
Sistemas empotrados			6 ECTS
Tecnología de computadores			6 ECTS
Tecnología y Organización de Computadores			6 ECTS
<b>Módulo:</b> Ingeniería de computadores			
<b>Departamento:</b> Estructura de la Materia, Física Térmica y Electrónica		<b>Coordinador:</b> García Hernansanz, Rodrigo	

**Descripción de contenidos mínimos:**

- Introducción a la Física de Dispositivos.
- Dispositivos electrónicos: estructuras básicas.
- Dispositivos opto-electrónicos.
- El transistor MOSFET.
- Ecuaciones y parámetros característicos.
- Familias lógicas.
- Circuitos de memoria.

**Programa detallado:**

Tema 1: Transistor MOSFET.

- Funcionamiento y ecuaciones características: Estructura del MOSFET. Tensión umbral. Parámetro de transconductancia. Tensión de saturación. Curvas características. Ecuaciones. Símbolos. Efectos de segundo orden.
- Parámetros del modelo Spice.

Tema 2: Inversores MOS. Características estáticas.

- Introducción: Característica ideal de un inversor. Estructura de un inversor. Tipos de familias lógicas.
- El MOSFET como interruptor.
- Parámetros estáticos característicos de un inversor: Tensiones críticas. Márgenes de ruido. Potencia disipada. Abanicos de entrada y de salida. Área ocupada.
- Inversores NMOS: Inversor NMOS con carga resistiva. Inversores NMOS con carga NMOS. Inversor pseudo-NMOS.
- Inversor CMOS: Estructura y funcionamiento general. Tensiones críticas. Inversor simétrico. Potencia disipada.
- Inversor CMOS de 3 estados.

Tema 3: Circuitos MOS combinacionales.

- Circuitos NMOS y pseudo-NMOS: Estructura general. Puerta NOR. Puerta NAND. Puertas complejas.
- Circuitos CMOS: Estructura general. Puerta NOR. Puerta NAND. Puertas complejas.
- Puertas de transmisión: Transistor NMOS como puerta de transmisión. Transistor PMOS como puerta de transmisión. Puerta de transmisión CMOS. Circuitos lógicos con puertas de transmisión.

Tema 4: Características dinámicas de los circuitos MOS.

- Efectos capacitivos en los circuitos MOS: Aparición de retardos como consecuencia de la presencia de capacidades. Capacidades del MOSFET. Efecto de las interconexiones.
- Parámetros dinámicos de un inversor: Potencia disipada dinámica. Tiempos de retardo.
- Tiempos de retardo en el inversor CMOS: Capacidad de carga equivalente. Cálculo de los tiempos. Consideraciones de diseño.
- Circuitos lógicos dinámicos.

Tema 5: Circuitos secuenciales.

- Circuito biestable básico.
- Latches y flip flops S-R y J-K: Latch S-R asíncrono. Latch S-R síncrono. Latch J-K síncrono. Flip-flop J-K maestro esclavo.
- Latches y flip flops tipo D: Latch tipo D. Flip-flop tipo D maestro-esclavo. Flip-flop tipo D activado por flanco.
- Requerimientos de temporización.

Tema 6: Memorias basadas en semiconductores.

- Cuestiones generales: Parámetros característicos. Clasificación de las memorias. Organización general de una memoria.
- Memorias volátiles (SRAM y DRAM): Celda SRAM. Escritura de la celda SRAM. Estructura de una memoria SRAM. Decodificador de filas. Decodificador de columnas. Amplificador sensor y circuito de precarga. Lectura de una celda SRAM. Celda DRAM. Estructura y funcionamiento de una memoria DRAM.
- Memorias no volátiles: Memorias ROM y PROM. Memorias EPROM. Memorias EEPROM y Flash.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Programa detallado en inglés:**

Unit 1: MOSFET Transistor

- Operation and characteristic equations: MOSFET structure. Threshold voltage. Transconductance parameter. Saturation voltage. Characteristic curves. Equations. Symbols. Second order effects.
- Spice model parameters.

Unit 2: MOS inverters. Static characteristics

- Introduction. Ideal inverter characteristic. Inverter structure. Types of logic families.
- The MOSFET as a switch.
- Characteristic static parameters of an inverter. Critical voltages. Noise margins. Power dissipation. Fan-in and fan-out. Area considerations.
- NMOS inverters: resistive load inverter. NMOS loaded inverters. pseudo-NMOS inverter.
- CMOS inverter: Operation and structure. Critical voltages. Symmetric inverter. Power dissipation.
- 3-state CMOS inverter.

Unit 3: MOS Combinational circuits

- NMOS and pseudo-NMOS circuits: General structure. NOR gate. NAND gate. Complex gates.
- Transmission gates: NMOS transistor as a transmission gate. PMOS transistor as a transmission gate. CMOS transmission gate. Transmission gate logic circuits.

Unit T4: Switching characteristics of MOS circuits

- Capacitances in MOS circuits: Delay times due to capacitances. MOSFET capacitances. Interconnection effects
- Inverter switching parameters: Switching power dissipation. Delay times.
- Delay times in CMOS inverters: Equivalent load capacitance. Delay time calculation. Design considerations.
- Dynamic logic circuits.

Unit 5: Sequential circuits

- Basic bistable circuit.
- S-R and J-K latches and flip flops: asynchronous S-R latch. Synchronous S-R latch. Synchronous J-K latch. Master-slave J-K flip-flop J-K.
- D type latches and flip flops: D type latch. Master-slave D type flip-flop. Edge-triggered D flip-flop.
- Timing considerations.

Unit 6: Semiconductor memories

- General considerations: Characteristic parameters. Memory classification. General organization of a memory.
- Volatile memories (SRAM and DRAM): SRAM cell. SRAM cell writing. SRAM memory structure. Row decoder. Column decoder. Sensing amplifier and precharge circuit. SRAM cell reading. DRAM cell. Structure and operation of a DRAM memory.
- Nonvolatile memories: ROM and PROM memories. EPROM Memories. EEPROM and Flash memories.

**Competencias de la asignatura:**

**Generales:**

CG14-Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.

**Específicas:**

CE\_GIC1-Capacidad de diseñar y construir sistemas digitales, incluyendo computadores, sistemas basados en microprocesador y sistemas de comunicaciones.

**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.

**Resultados de aprendizaje:**

Relacionar la información codificada en binario con las señales eléctricas que físicamente contienen esa información en los circuitos digitales. (CG14, CE\_GIC1)

Analizar la estructura y el funcionamiento físico de los dispositivos electrónicos utilizados en la fabricación de circuitos y sistemas digitales. (CG14, CT2)

Relacionar los parámetros físicos del dispositivo, dependientes de la tecnología de fabricación, con sus parámetros eléctricos, que determinarán las características de los circuitos digitales. (CG14, CE\_GIC1)

Analizar el funcionamiento de circuitos electrónicos a nivel de dispositivo. (CG14, CT2)

Utilizar la simbología, la terminología y el lenguaje técnico específicos en el campo de los dispositivos electrónicos y de los circuitos digitales para comunicar conceptos y conocimientos con precisión. (CT1)

Calcular los parámetros que caracterizan el funcionamiento estático y dinámico de un circuito digital a partir de las características de los dispositivos que lo constituyen. (CT3, CE\_GIC1)

Analizar las características de las principales tecnologías lógicas para la realización de circuitos digitales lógicos, en base al funcionamiento de los dispositivos que los constituyen y la propia estructura del circuito. (CT2, CE\_GIC1)

Diseñar a nivel de dispositivo circuitos lógicos digitales para cumplir una función especificada con unos determinados requerimientos de parámetros de funcionamiento. (CT3, CE\_GIC1)

Evaluar la tecnología óptima para una determinada aplicación en función de sus características de funcionamiento estático y dinámico y de los requerimientos de la aplicación. (CE\_GIC1)

Comprender las necesidades, la evolución y el impacto social de la tecnología microelectrónica de fabricación de circuitos lógicos. (CG14, CT5)

Organizar y planificar el tiempo requerido para cumplir objetivos en fechas determinadas. (CT4)

**Evaluación detallada:**

Examen final 80-90%.

Realización de ejercicios en clase 10-20%

Se realizará un examen final de cuestiones y de problemas tanto en la convocatoria ordinaria como en la extraordinaria. Para la parte de problemas se facilitará un formulario.

Al finalizar el tema 2 y al finalizar el tema 4, se dedicará un día de clase para que los alumnos resuelvan ejercicios como los que se han resuelto en clase y respondan cuestiones sobre las explicaciones. También se recogerán los ejercicios realizados en clase durante alguna clase de problemas o ejercicios propuestos.

La calificación final en la convocatoria ordinaria será la mayor de las dos siguientes opciones:

$$CFINAL = 0.8CEX,ORD + 0.2CEJ \text{ o } CFINAL = 0.9CEX,ORD + 0.1CEJ.$$

La calificación final en la convocatoria extraordinaria será la mayor de las siguientes opciones:

$$CFINAL = 0.8CEX,EXT + 0.2CEJ \text{ o } CFINAL = 0.9CEX,EXT + 0.1CEJ.$$

CFINAL = Calificación final de la asignatura.

CEX,ORD = Calificación del examen de la convocatoria ordinaria (sobre una escala de 10).

CEJ = Calificación de los ejercicios realizados en clase (sobre una escala de 10). La calificación obtenida en este apartado durante el curso se conserva para las convocatorias ordinaria y extraordinaria.

CEX,EXT = Calificación del examen de la convocatoria extraordinaria (sobre una escala de 10).

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,50

Problemas: 1,50

Laboratorios: 0,00

Otras actividades:

Actividades presenciales:

Clases teóricas con ejemplos de aplicación: 30% de la dedicación total del alumno

Clases de resolución de problemas: 10%

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Actividades dirigidas:  
Tutorías y trabajos dirigidos 10%

Trabajo personal:  
Estudio, realización de ejercicios propuestos, preparación y realización del examen final: 50%

**Bibliografía:**

1. A. S. Sedra y K. C. Smith. "Circuitos Microelectrónicos". McGraw-Hill.
2. T. A. DeMassa y Z. Ciccone. "Digital integrated circuits". John Wiley & Sons.
3. J. M. Rabaey, A. Chandrakasan y B. Nikolic. "Digital integrated circuits. A design perspective". Prentice Hall.
4. S.M. Kang y Y. Leblebici. "CMOS Digital Integrated Circuits, Analysis and Design". Mc-Graw Hill.

Ficha docente guardada por última vez el 17/07/2020 17:00:00 por el departamento: Estructura de la Materia, Física Térmica y Electrónica

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803217 - Tecnología de computadores		<b>Abrev:</b> TC	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Computer Systems Technology		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Tecnología y Arquitectura de Computadores		<b>24 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>			
Arquitectura de Computadores		6 ECTS	
Electrónica		6 ECTS	
Sistemas empuotrados		6 ECTS	
<b>Módulo:</b> Ingeniería de computadores			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Mecha López, Hortensia	

**Descripción de contenidos mínimos:**

- Introducción al diseño de circuitos integrados.
- Temporización y sincronización de sistemas digitales.
- Circuitos aritméticos.
- Diseño multimódulo.
- Sistemas algorítmicos.
- Organización de la memoria.
- Lenguajes de descripción de HW.
- Diseño con FPGAs

**Programa detallado:**

## Tema 1. Diseño y modelado hardware con VHDL

- 1.1. Flujo de diseño
- 1.2. Lenguajes de descripción hardware (HDL)
- 1.3. Simulación con VHDL
- 1.4. Estructura de un modelo VHDL
- 1.5. Elementos básicos de VHDL
- 1.6. Máquina de estados finita (FSM)
- 1.7. Otros elementos de VHDL
- 1.8. Tech-bench de simulación

## Tema 2. Evaluación parámetros físicos del diseño

- 2.1. ¿Por qué evaluar?
- 2.2. Análisis estático de tiempos (STA)
- 2.3. Comportamiento dinámico
- 2.3. Análisis del área
- 2.4. Análisis del consumo

## Tema 3. Diseño combinacional avanzado

- 3.1. Conocimientos previos
- 3.2. Módulos combinacionales y diseño multimódulo
- 3.3. Unidades funcionales multi-función
- 3.5. Redes iterativas 1-D y 2-D
- 3.5. Técnicas para mejorar el rendimiento
- 3.6. Segmentación
- 3.7. Errores de diseño

## Tema 4. Diseño algorítmico

- 4.1. Introducción
- 4.2. Elementos de memoria
- 4.3. Diagrama ASM, diseño de la ruta de datos y diseño de la UC
- 4.4. Principios de diseño: top- down/bottom-up, divide y vencerá, iterativo
- 4.5. Diseño RTL

## Tema 5. Memorias

- 5.1. Jerarquía de memoria
- 5.2. Tecnologías de memoria: Memoria estática y dinámica; DRAM, EPROM, FLASH
- 5.3. Organización de la memoria principal (características y rendimiento)
- 5.4. Latencia, tiempo de ciclo, ancho de banda e interleaving
- 5.5. Memoria de acceso asociativo

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



5.6. Códigos de detección de errores

Tema 6. Aritmética

- 6.1. Sumadores rápidos (anticipación y punteo de arrastres)
- 6.2. Multiplicadores sin/con signo (secuencial y combinacional)
- 6.3. Divisor secuencial

**Programa detallado en inglés:**

- 1. Hardware Design and Modeling with VHDL
  - 1.1. Design Flow
  - 1.2. Hardware Description Language (HDL)
  - 1.3. Simulation with VHDL
  - 1.4. VHDL Modeling
  - 1.5. Basic Elements of VHDL
  - 1.6. Finite State Machine (FSM)
  - 1.7. Other Elements of VHDL
  - 1.8. Techbenches
- 2. Physical Parameter Measurement
  - 2.1. Why evaluate?
  - 2.2. Static Timing Analysis (STA)
  - 2.3. Dynamic Behavior
  - 2.3. Area Measurement
  - 2.4. Power-Consumption Measurement
- 3. Advanced Combinational Design
  - 3.1. Previous Knowledge
  - 3.2. Multimodule Design
  - 3.3. Multi-function Functional Units
  - 3.5. 1D and 2D Iterative Networks
  - 3.5. Techniques to Improve Performance
  - 3.6. Pipelining
  - 3.7. Design Errors
- 4. Algorithmic Design
  - 4.1. Introduction
  - 4.2. Storage Elements
  - 4.3. ASM diagrams, Datapath and Control Unit
  - 4.4. Principles of design
  - 4.5. RTL design
- 5. Memories
  - 5.1. Memory Hierarchy
  - 5.2. Memory Technologies
  - 5.3. Memory Organization
  - 5.4. Latency, Cycle Time, Bandwidth and Interleaving
  - 5.5. Associative Memory
  - 5.6. Error Detection Codes
- 6. Arithmetic
  - 6.1. Fast Adders
  - 6.2. Signed and Unsigned Multipliers
  - 6.3. Sequential Divider

**Competencias de la asignatura:**

**Generales:**

- CG14-Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.
- CG19-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Específicas:**

- CE\_GIC1-Capacidad de diseñar y construir sistemas digitales, incluyendo computadores, sistemas basados en microprocesador y sistemas de comunicaciones.
- CE\_GIC7-Capacidad para analizar, evaluar, seleccionar y configurar plataformas hardware para el desarrollo y ejecución de aplicaciones y servicios informáticos.

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

- Comprender e interpretar las especificaciones y los parámetros de diseño de un sistema electrónico digital. (CG14)
- Comprender los factores reales que afectan al diseño de estos sistemas y su influencia en el estilo de diseño y el resultado final. (CG14)
- Comprender y usar distintas tecnologías de fabricación de memorias. (CG14)
- Construir y evaluar, mediante las medidas oportunas en el laboratorio, diferentes sistemas digitales de complejidad media diseñados en un lenguaje de descripción hardware. (CG14)
- Decidir la estructura del sistema electrónico digital adecuada para implementar la funcionalidad especificada. (CG14)
- Diseñar distintos tipos de circuitos aritméticos y evaluar las características de la implementación física de cada uno de ellos. (CG14)
- Modificar las técnicas y recursos disponibles para adaptarlos a las necesidades específicas del diseño de sistemas digitales. (CT3)
- Planear distintas opciones de diseño y seleccionar aquellas que mejor satisfagan las especificaciones. (CG14)
- Planificar adecuadamente las etapas de desarrollo para un sistema complejo. (CT4)
- Representar e interpretar, mediante cronogramas, la respuesta en el tiempo de un sistema digital. (CG14)
- Valorar el impacto medioambiental derivado de la puesta en marcha de un sistema informático. (CT5)
- Analizar el comportamiento temporal de los circuitos y plantear hipótesis sobre las posibles causas de su comportamiento erróneo. (CG14)
- Aplicar la teoría de circuitos combinatoriales y secuenciales para diseñar y evaluar distintas opciones de diseño de una especificación dada. (CG14)
- Conocer y aplicar los métodos básicos para mejorar la temporización de un circuito digital. (CG14)
- Conocer y argumentar las ventajas e inconvenientes de distintas opciones de sistemas secuenciales. (CG14)
- Diferenciar las distintas perspectivas y necesidades de comunicación según los participantes en un proyecto de desarrollo. (CT1)
- Diseñar circuitos que satisfagan la especificación. (CT2)
- Estimar las características físicas de la implementación de un sistema electrónico digital. (CG14)
- Evaluar una especificación y justificar modificaciones basadas en la tecnología. (CT2)

**Evaluación detallada:**

- Examen final en las convocatorias ordinaria y extraordinaria
- Exámenes escritos obligatorios con cuestiones teóricas y prácticas, siendo el mismo examen en todos los grupos de la asignatura y con criterios detallados de puntuación comunes.
- Prácticas de laboratorio. Asistencia obligatoria. Mismas prácticas en todos los grupos y criterios detallados de puntuación comunes. La nota del laboratorio será la media ponderada de las calificaciones de las prácticas propuestas y de la presentación de un proyecto final. Dicho proyecto

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



supondrá entre 60-70% de la nota del laboratorio y servirá para evaluar que se han adquirido todas las destrezas relacionadas con la programación en VHDL. La nota de las prácticas de laboratorio y del proyecto es recuperable en la convocatoria extraordinaria

Calificación. Método de cálculo idéntico en las convocatorias ordinaria y extraordinaria. Si la nota del examen es igual o superior a 3.5 (sobre 10) entonces la calificación de la asignatura será la siguiente

$$0,6 * \text{Nota del examen} + 0,15 * \text{Nota del laboratorio} + 0,25 * \text{Proyecto final}$$

En el caso de que la nota del examen sea inferior a 3,5/10 entonces la calificación de la asignatura será suspenso.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,40

Problemas: 1,10

Laboratorios: 1,50

Otras actividades:

Clases teóricas: Clases magistrales de teoría en aula. En promedio 2.25 horas a la semana.

Clases prácticas: Resolución en el aula de problemas de diseño comprobando la idoneidad del sistema final obtenido. En promedio 0.75 horas a la semana.

Laboratorios: Prácticas en el laboratorio de diseño de sistemas reales utilizando VHDL e implementando el circuito sobre una plataforma FPGA. En promedio 1 hora a la semana.

Resolución de problemas de diseño comprobando la idoneidad del sistema final obtenido.

Laboratorios

Se realizarán prácticas de diseño de sistemas reales utilizando VHDL e implementando el circuito sobre una plataforma FPGA. En promedio 1 hora a la semana (en laboratorio).

**Bibliografía:**

- Gajski, D. D., González Colás, A., García Puntónet Carlos, & Valero Cortés, M. "Principios de diseño digital" (2ª reimp), Prentice-Hall, (2004).
- Brown, S., Vranesic, Z., "Fundamentos de lógica digital con diseño VHDL", Mc. Graw-Hill, 2006
- Rabaey, J.M., "Circuitos integrados digitales : una perspectiva de diseño", Prentice Hall, 2004
- Parhami, B., "Computer arithmetic : algorithms and hardware designs", Oxford University Press, 2000
- Jacob, B., Ng, S., Wang, D., "Memory systems : cache, DRAM, disk", Morgan Kaufmann Publishers, 2007
- Pong P. Chu, "RTL hardware design using VHDL: coding for efficiency, portability, and scalability", John Wiley & Sons, 2006
- Peter J. Ashenden, "The designer's guide to VHDL", Morgan Kaufmann Publishers, 2008

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803222 - Métodos Estadísticos		<b>Abrev:</b> ME	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Statistical Methods		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Métodos Estadísticos		<b>6 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> No hay			
<b>Módulo:</b> Complementario			
<b>Departamento:</b> Estadística e Investigación Operativa		<b>Coordinador:</b> Romero Ramos, Eva	

**Descripción de contenidos mínimos:**

- Análisis descriptivo de datos estadísticos.
- Regresión lineal.
- Modelos linealizables.
- Probabilidad.
- Variables aleatorias.
- Teoremas de límite.
- Muestreo.
- Inferencia paramétrica y no paramétrica.
- Software estadístico.

**Programa detallado:**

1. Estadística descriptiva en una variable estadística. Medidas y relaciones entre variables estadísticas. Regresión y correlación.
2. Probabilidad. Definición de suceso y operaciones de sucesos. Sucesos independientes. Probabilidad condicionada. Teorema de la Probabilidad Total y Teorema de Bayes.
3. Variables aleatorias . Distribución de variables aleatorias. Distribuciones notables. Convergencia: Teorema Central del Límite.
4. Inferencia Estadística. Muestreo. Distribuciones asociadas a la Normal. Estimación puntual y por intervalos. Contrastes de hipótesis. Software estadístico

**Programa detallado en inglés:**

1. One-dimensional descriptive statistics. Measures and relationships between statistical variables. Regression and correlation.
2. Probability. Events and operations with events. Independent events. Conditional probability. Total probability and Bayes theorems.
3. Random variables. Distribution of a random variable. Usual distributions. Convergence: The Central Limit Theorem.
4. Statistical inference. Sampling. Sampling from a normal distribution. Point and interval estimation. Hypothesis testing. Statistical software.

**Competencias de la asignatura:****Generales:**

CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

Reconocer un problema matemático - estadístico entre los propuestos en la asignatura (CG1)

Clasificar un problema estadístico y resolverlo (CG1)

Realizar en equipo los problemas prácticos propuestos (CT1)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Clasificar problemas estadísticos y resolverlos según las técnicas estadísticas establecidas (CT2)
- Aplicar los distintos métodos estadísticos para resolver problemas (CT2)
- Comparar las técnicas de inferencia estadística seleccionando la solución más adecuada al problema planteado (CT2, CT3)
- Resolver ejercicios con el software adecuado analizando el problema y diseñando la solución (CT2, CT3)
- Analizar la solución de un problema estadístico valorando su repercusión social (CT5)

**Evaluación detallada:**

Examen final: 80%

Otras actividades: 20%. (Realización de trabajos, entregas de ejercicios y/o exámenes parciales).

La calificación obtenida por la realización de trabajos, entregas de ejercicios o exámenes parciales obtenida durante el curso, se conservará para la convocatoria extraordinaria, donde no existirá un periodo de entrega adicional de trabajos o ejercicios.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 3,00

Laboratorios: 0,00

Otras actividades:

Actividades presenciales: Clases teóricas magistrales. Clases de problemas.

Actividades dirigidas: Entrega de ejercicios y/o trabajos dirigidos.

Trabajo personal del alumno: Estudio, preparación de exámenes, realización de ejercicios.

Realización de exámenes.

**Bibliografía:**

- Devore, J.L. Probabilidad y Estadística para Ingeniería y Ciencias. Thompson - Learning. (2001)
- García, A y otros; Estadística I (Informática de Sistemas). UNED, (1995)
- Horra Navarro, Julian; Estadística Aplicada. Díaz de Santos, (2003)
- Rodríguez L. y Tomeo V. Métodos Estadísticos para Ingeniería. Garceta Grupo Editorial. (2011)
- Spiegel, M. R., Schiler, J. Srinivasan, R.A. Probabilidad y Estadística. Mc-Graw-Hill. (2001)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805479 - Fundamentos de Algoritmia		<b>Abrev:</b> FAL	<b>4,5 ECTS</b>
<b>Asignatura en Inglés:</b> Fundamentals of Algorithms		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Programación fundamental		<b>21 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>			
Estructuras de datos		4,5 ECTS	
Tecnología de la programación I		6 ECTS	
Tecnología de la programación II		6 ECTS	
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Interdepartamental ISIA / SIC		<b>Coordinador:</b> Segura Díaz, Clara Mª	

**Descripción de contenidos mínimos:**

- Análisis de la eficiencia de los algoritmos.
- Diseño y análisis de algoritmos iterativos y recursivos.
- Algoritmos de ordenación.
- Esquemas algorítmicos de divide y vencerás y vuelta atrás.

**Programa detallado:**

- Análisis de la eficiencia de los algoritmos
- Especificación de algoritmos
- Diseño y análisis de algoritmos iterativos
- Diseño y análisis de algoritmos recursivos
- Algoritmos de divide y vencerás
- Algoritmos de vuelta atrás

**Programa detallado en inglés:**

- Analyzing the efficiency of algorithms
- Formally specifying algorithms
- Design and analysis of iterative algorithms
- Design and analysis of recursive algorithms
- Divide and conquer algorithms
- Backtracking algorithms

**Competencias de la asignatura:****Generales:**

- CG2-Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.
- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Analizar y justificar el coste de algoritmos iterativos y recursivos (CG2, CG11, CT2)
- Comparar el coste de algoritmos que resuelven el mismo problema y seleccionar el más eficiente (CG2, CG11, CT2)
- Diseñar e implementar algoritmos recursivos correctos y eficientes para resolver problemas (CG11)
- Diseñar e implementar algoritmos iterativos correctos y eficientes para resolver problemas (CG11)
- Especificar algoritmos de forma que el comportamiento esperado del mismo sea lo más claro y preciso posible (CG2)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Juzgar la corrección de un algoritmo con respecto a su especificación (CG2, CG11)

Utilizar los esquemas algorítmicos vistos en clase para resolver problemas y valorar la conveniencia de su utilización (CG11, CT2)

Utilizar técnicas de generalización para definir algoritmos recursivos (CG11)

**Evaluación detallada:**

Un 20% de la nota final se obtendrá a lo largo del curso mediante la evaluación de las actividades prácticas. Las actividades no realizadas en el tiempo indicado no podrán ser recuperadas en la convocatoria extraordinaria. La asistencia a las clases prácticas es evaluable. La no asistencia en los días indicados para evaluación continua supondrá la pérdida de la calificación de la prueba. La evaluación puede incluir además la realización individual de prácticas y problemas, la exposición en público de las soluciones propias, la realización de mini-exámenes escritos, las tutorías obligatorias y cualquier otra que permita conocer el grado de conocimientos y destreza alcanzados por el alumno.

Un 80% de la nota final se obtendrá mediante el examen de la convocatoria ordinaria o de la extraordinaria. El examen constará de dos partes: un test eliminatorio que será indispensable superar para poder aprobar y que en caso de ser superado supondrá un 10% de la nota final; y una segunda parte de resolución de problemas que supondrá un 70% de la nota final.

Para poder aprobar la asignatura se requerirá superar el test eliminatorio, y obtener al menos una calificación de 5 sobre 10 en el examen. La nota mínima necesaria para dar por superado el test se anunciará durante el periodo lectivo y será igual para todos los grupos.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,75

Laboratorios: 0,75

Otras actividades:

Clases teóricas

Enseñanza presencial teórica.

Clases prácticas

Realización individual de problemas con corrección colectiva en clase.

Laboratorios

Realización individual de prácticas tutorizadas en el laboratorio.

Otras actividades

Tutorías individuales.

**Bibliografía:**

Diseño de programas: Formalismo y abstracción.

Tercera edición, Pearson/Prentice Hall 2005.

N. Martí Oliet, Y. Ortega Mallén, J. A. Verdejo López.

Estructuras de Datos y Métodos Algorítmicos: 213 Ejercicios resueltos.

Ibergarceta Publicaciones 2013.

N. Martí Oliet, C. Segura Díaz, J. A. Verdejo López.

Algoritmos correctos y eficientes: Diseño razonado ilustrado con ejercicios.

Ibergarceta Publicaciones, 2012

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805480 - Estructuras de datos		<b>Abrev:</b> ED	<b>ECTS:</b> 4,5 ECTS
<b>Asignatura en Inglés:</b> Data structures		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Programación fundamental		21 ECTS	
<b>Otras asignaturas en la misma materia:</b>			
Fundamentos de Algoritmia		4,5 ECTS	
Tecnología de la programación I		6 ECTS	
Tecnología de la programación II		6 ECTS	
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Interdepartamental ISIA / SIC		<b>Coordinador:</b> Gómez Albarrán, M <sup>a</sup> de las Mercedes	

**Descripción de contenidos mínimos:**

Especificación e implementación de tipos abstractos de datos.  
Tipos de datos lineales y arborescentes.  
Tablas asociativas.

**Programa detallado:**

1. Diseño e implementación de tipos abstractos de datos
2. Tipos de datos lineales
3. Tipos de datos arborescentes
4. Diccionarios
5. Aplicaciones de tipos abstractos de datos

**Programa detallado en inglés:**

1. Design and implementation of abstract data types
2. Linear data types
3. Tree-like data types
4. Dictionaries
5. Applications of abstract data types

**Competencias de la asignatura:****Generales:**

- CG2-Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.
- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CG12-Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Combinar el uso de tipos abstractos de datos y de técnicas algorítmicas en la resolución de problemas (CT3)
- Conocer las estructuras de datos vistas en clase y cómo se utilizan para implementar tipos abstractos de datos (CG12)
- Conocer los tipos abstractos de datos vistos en clase, sus posibles formas de implementación y la eficiencia de las mismas (CG12)
- Diseñar e implementar tipos abstractos de datos (CG12)
- Seleccionar tipos abstractos de datos para la resolución de problemas (CG12, CT3)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Evaluación detallada:**

30% de la nota se obtendrá a lo largo del curso mediante la evaluación de las actividades prácticas. Las actividades no realizadas en el tiempo indicado no podrán ser recuperadas en la convocatoria extraordinaria. La asistencia a las clases prácticas es evaluable. La no asistencia en los días indicados para evaluación continua supondrá la pérdida de la calificación de la prueba. La evaluación puede incluir además la realización individual de prácticas y problemas, la exposición en público de las soluciones propias, la realización de mini-exámenes escritos, las tutorías obligatorias y cualquier otra que permita conocer el grado de conocimientos y destreza alcanzados por el alumno.

70% de la nota se alcanzará mediante el examen final en convocatoria ordinaria. Para poder aprobar la asignatura se requerirá al menos una calificación de 5 sobre 10 como nota del examen. Examen en convocatoria extraordinaria para los alumnos que no hayan aprobado en la convocatoria ordinaria. Para poder aprobar en la convocatoria extraordinaria se requerirá al menos una calificación de 5 sobre 10 como nota del examen. Si en una convocatoria no se alcanza el 5 en la nota del examen, esta será la nota que aparezca en el acta.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,75

Laboratorios: 0,75

Otras actividades:

Clases teóricas

Clases prácticas

Realización individual de problemas con corrección colectiva en clase.

Laboratorios

Realización individual de prácticas tutorizadas en el laboratorio.

Otras actividades

Tutorías individuales.

**Bibliografía:**

R. Peña. Diseño de programas: Formalismo y abstracción. Tercera edición, Pearson/Prentice Hall 2005.

M. Rodríguez Artalejo, P. A. González Calero, M. A. Gómez Martín. Estructuras de datos: un enfoque moderno. Editorial Complutense 2011.

N. Martí Oliet, Y. Ortega Mallén, J. A. Verdejo López. Estructuras de Datos y Métodos Algorítmicos: 213 Ejercicios resueltos. Ibergarceta Publicaciones 2013.

M. A. Weiss. Data Structures and Algorithm Analysis in C++. Fourth edition. Addison-Wesley, 2014.

E. Horowitz, S. Sahni, D. Mehta. Fundamentals of Data Structures in C++. Computer Science Press 1995.

Ficha docente guardada por última vez el 30/06/2023 15:27:00 por el usuario: Coordinador GII

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID  
FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805481 - Tecnología de la programación I		<b>Abrev:</b> TP1	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Programming Technology I		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Programación fundamental		<b>21 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>			
Estructuras de datos		4,5 ECTS	
Fundamentos de Algoritmia		4,5 ECTS	
Tecnología de la programación II		6 ECTS	
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Interdepartamental ISIA / SIC		<b>Coordinador:</b> Rosa Velardo, Fernando	

**Descripción de contenidos mínimos:**

Introducción a la Programación Orientada a Objetos.  
Clases y Objetos.  
Herencia.  
Objetos y memoria dinámica.  
Polimorfismo y vinculación dinámica.  
Entornos de desarrollo, bibliotecas y marcos de aplicación.  
Entrada / salida.  
Tratamiento de excepciones.  
Realización de prácticas en laboratorio.

**Programa detallado:**

Introducción a la programación orientada a objetos (POO)  
Clases y objetos, construcción y destrucción de objetos, memoria dinámica.  
Herencia  
Polimorfismo y vinculación dinámica.  
Excepciones  
Entrada / salida

**Programa detallado en inglés:**

Introduction to Object Oriented Programming (OOP)  
Classes and objects, creation and destruction of objects, dynamic memory.  
Inheritance.  
Polymorphism and dynamic binding.  
Exceptions.  
Input / Output

**Competencias de la asignatura:**

**Generales:**

CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.  
CG13-Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.  
CT2-Capacidad de análisis y síntesis en la resolución de problemas.  
CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Argumentar las elecciones de diseño en las prácticas. (CT1)
- Desarrollar las prácticas y ejercicios tomando decisiones sobre su diseño. (CG13, CT3)
- Realizar en equipo las prácticas de programación. (CT1)
- Resolver ejercicios de programación analizando el problema y diseñando la solución. (CG13, CT2, CT3)
- Desarrollar y validar programas expresados utilizando el paradigma de la orientación a objetos en lenguajes de programación concretos. (CG3)
- Escribir y depurar programas orientados a objetos. (CG13)
- Utilizar entornos integrados de desarrollo para la construcción de aplicaciones informáticas. (CG3)

**Evaluación detallada:**

Las prácticas se realizan en grupos de dos personas, son obligatorias, tienen carácter eliminatorio y su defensa es individual. Durante el curso se irá indicando tras cada entrega si la práctica supera los mínimos exigidos o no. La creación de grupos se realizará según el criterio del profesor. Las prácticas se entregarán en el plazo y forma que disponga el profesor y siempre dentro de los plazos establecidos. Entregas fuera de plazo se consideran como NO APTAS, y por lo tanto implicarán que la asignatura está suspensa. La defensa de las prácticas se realizará en el laboratorio. Para la convocatoria extraordinaria se especificará un nuevo plazo de entrega. Las prácticas entregadas en dicho plazo contabilizarán 0 puntos en la evaluación continua.

La calificación de la asignatura en la convocatoria ordinaria, y solo en el caso de no tener ninguna práctica calificada como NO APTA (en otro caso la asignatura estará suspensa), se obtendrá de la siguiente forma:

- Un 20% en base a las prácticas entregadas en plazo en el periodo de clases.
- Un 80% en base a un examen teórico/práctico individual a realizar en los laboratorios.

La calificación de la asignatura en la convocatoria extraordinaria, y solo en el caso de haber superado todas las prácticas de la asignatura (en otro caso la asignatura estará suspensa), se obtendrá de la siguiente forma:

- Un 20% en base a las prácticas entregadas en plazo en el periodo de clases. Para la convocatoria extraordinaria se habilitará un día de entrega de prácticas, y aunque no contabilizan en la evaluación, es condición necesaria y obligatoria para poder aprobar la asignatura, haber entregado todas las prácticas y que todas alcancen los mínimos exigidos.
- Un 80% en base a un examen teórico/práctico individual a realizar en los laboratorios.

Tanto en la convocatoria ordinaria como extraordinaria, será necesario obtener al menos un 1/3 de la puntuación en la parte teórica del examen para que se le corrija la parte práctica del examen (si no se corrige, su nota de examen sería la nota de la parte teórica calculada sobre 10). A su vez, será necesario obtener como mínimo un 5 sobre 10 en el examen para poder hacer media con las prácticas.

Si un alumno no se presenta al examen de una convocatoria, que tenga o no todas las prácticas aprobadas, figurará como NO PRESENTADO en esta convocatoria.

Si un alumno, se presenta al examen de una convocatoria, teniendo las prácticas suspensas, figurará como SUSPENSO en esta convocatoria, y su nota será la del examen truncada a 4 sobre 10.

**Actividades docentes:**

Reparto de créditos:

- Teoría: 3,00
- Problemas: 0,00
- Laboratorios: 3,00

Otras actividades:

- Clases teóricas:
  - Enseñanza presencial teórica. Realización individual de problemas con corrección colectiva en clase.
- Laboratorios:
  - Realización por parejas de prácticas tuteladas en el laboratorio.
- Otras actividades:
  - Tutorías individuales.

**Bibliografía:**

- Timothy Budd: An introduction to object-oriented programming. Addison Wesley. 2002.
- David J. Barnes, Michael Kolling: Programación orientada a objetos con Java. Tercera Edición. Pearson Educación, 2007
- Bruce Eckel: Thinking in Java. Cuarta Edición. Prentice Hall, 2006.
- Benjamin Evans. Java in a Nutshell. A Desktop Quick Reference (covers Java 8), O'Reilly, 2015.
- Erich Gamma, Richard Helm, Ralph Johnson, John M. Vlissides: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> 2º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805482 - Tecnología de la programación II	<b>Abrev:</b> TP2	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Programming Technology II	<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Programación fundamental	<b>21 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>		
Estructuras de datos	4,5 ECTS	
Fundamentos de Algoritmia	4,5 ECTS	
Tecnología de la programación I	6 ECTS	
<b>Módulo:</b> Materias comunes a la rama de la informática		
<b>Departamento:</b> Interdepartamental ISIA / SIC <b>Coordinador:</b> Genaim , Samir		

**Descripción de contenidos mínimos:**

Entornos de desarrollo, bibliotecas y marcos de aplicación.  
Programación basada en eventos y componentes visuales.  
Interfaces gráficas de usuario.  
Genericidad y colecciones.  
Programación multihilo.  
Realización de prácticas en laboratorio.

**Programa detallado:**

Introducción al diseño orientado a objetos.  
Patrones de diseño  
Genericidad y colecciones  
Componentes visuales  
Modelo/vista/controlador  
Introducción a la programación con uso de hebras

**Programa detallado en inglés:**

Generics and collections  
Introduction to Object Oriented  
Design. Patterns.  
Graphical User Interface  
Model View Controller Threads  
Introduction to programming with threads

**Competencias de la asignatura:****Generales:**

CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.  
CG13-Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.  
CT2-Capacidad de análisis y síntesis en la resolución de problemas.  
CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Argumentar las elecciones de diseño en las prácticas. (CT1)  
Desarrollar las prácticas y ejercicios tomando decisiones sobre su diseño. (CG13, CT3)  
Realizar en equipo las prácticas de programación. (CT1)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Resolver ejercicios de programación analizando el problema y diseñando la solución. (CG13, CT2, CT3)

Desarrollar y validar programas expresados utilizando el paradigma de la orientación a objetos en lenguajes de programación concretos. (CG3)

Escribir y depurar programas orientados a objetos. (CG13)

Utilizar entornos integrados de desarrollo para la construcción de aplicaciones informáticas. (CG3)

**Evaluación detallada:**

Las prácticas se realizan en grupos de dos personas, son obligatorias, tienen carácter eliminatorio y su defensa es individual. Al inicio del curso se especificará el valor numérico de cada práctica y durante el curso se irá indicando tras cada entrega si la práctica supera los mínimos exigidos o no, junto con la puntuación obtenida. La creación de grupos se realizará según el criterio del profesor. Las prácticas se entregarán en el plazo y forma que disponga el profesor. Entregas fuera de plazo se consideran como NO APTAS, y por lo tanto implicarán que la asignatura está suspensa. La defensa de las prácticas se realizará en el laboratorio.

Para la convocatoria extraordinaria se especificará un nuevo plazo de entrega de prácticas pendientes (es decir, calificadas como NO APTAS o sin calificar). Las prácticas entregadas en dicho plazo contabilizarán 0 puntos en la evaluación global del curso.

La calificación de la asignatura en la convocatoria correspondiente, y solo en el caso de no tener ninguna práctica sin calificar o calificada como NO APTA (en otro caso la asignatura estará suspensa), se obtendrá de la siguiente forma:

- Un 20% en base a las prácticas entregadas en plazo en el periodo de clases.
- Un 80% en base a un examen teórico/práctico individual.

En el examen será necesario obtener como mínimo un 5 sobre 10 para poder hacer media con las prácticas.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Clases teóricas

Enseñanza presencial teórica. Realización individual de problemas con corrección colectiva en clase.

Laboratorios

Realización por parejas de prácticas tuteladas en el laboratorio.

Otras actividades

Tutorías individuales.

Presenciales

6 créditos

**Bibliografía:**

Timothy Budd: An introduction to object-oriented programming. Addison Wesley. 2002.

David J. Barnes, Michael Kolling: Programación orientada a objetos con Java. Tercera Edición. Pearson Educación, 2007

Bruce Eckel: Thinking in Java. Cuarta Edición. Prentice Hall, 2006.

Bruce Eckel: Thinking in C++. Segunda Edición. Prentice Hall, 2000.

Erich Gamma, Richard Helm, Ralph Johnson, John M. Vlissides: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:

**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805483 - Ingeniería del Software I		<b>Abrev:</b> IS1	<b>ECTS:</b> 4,5 ECTS
<b>Asignatura en Inglés:</b> Software Engineering I		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Desarrollo del software fundamental		15 ECTS	
<b>Otras asignaturas en la misma materia:</b>			
Bases de datos		6 ECTS	
Ingeniería del Software II		4,5 ECTS	
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Interdepartamental ISIA / SIC		<b>Coordinador:</b> Gómez Gauchía, Héctor	

**Descripción de contenidos mínimos:**

Introducción a la ingeniería del software.  
Lenguajes de modelado de software.  
El proceso de desarrollo de software: Modelado de flujos de trabajo.  
Planificación y gestión de proyectos.  
Análisis y especificación de requisitos: Modelado de requisitos software.

**Programa detallado:**

Introducción a la Ingeniería del Software.  
Modelos de procesos de desarrollo de software.  
Ingeniería de requisitos.  
Planificación y gestión de proyectos.

**Programa detallado en inglés:**

Introduction to Software Engineering.  
Models of software development processes.  
Requirements engineering.  
Planning and project management.

**Competencias de la asignatura:****Generales:**

CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

CG8-Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.

CG17-Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.

CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

CG20-Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.

CG22-Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.

CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Resultados de aprendizaje:**

- Analizar el impacto económico y en el cliente de un sistema informático. (CG8, CT5)
- Analizar los recursos necesarios para la ejecución de un proyecto. (CG8, CG20)
- Aplicar las capacidades de comunicación para exponer de forma organizada y clara los distintos aspectos del trabajo en el proyecto. (CG22)
- Aplicar las capacidades de comunicación, comprensión y síntesis para integrar y resolver las distintas perspectivas sobre el proyecto. (CG22, CT2)
- Diferenciar las distintas perspectivas y necesidades de comunicación según los participantes en un proyecto de desarrollo de un sistema informático. (CT1, CT2)
- Modificar las técnicas y recursos disponibles para adaptarlos a las necesidades específicas del desarrollo de un sistema informático. (CT3)
- Modificar un proceso de desarrollo software para adecuarlo a las necesidades específicas de un proyecto. (CG20)
- Planear la planificación de un proyecto en base a sus restricciones y recursos. (CG8, CT4)
- Planear las necesidades en recursos humanos de la ejecución de un proyecto de desarrollo software, así como su ejecución (CT4)
- Relacionar el éxito del proyecto con la motivación y toma decisiones adecuada del personal (CG22, CT4)
- Seleccionar y combinar críticamente las alternativas disponibles para abordar un aspecto del desarrollo de un sistema informático. (CT2, CT3)
- Usar los medios audiovisuales para expresar la información de un proyecto de sistema informático de forma apropiada para su comprensión por el cliente y el equipo de desarrollo. (CT1)
- Valorar el proceso de desarrollo software más adecuado para un proyecto. (CG20)

**Evaluación detallada:**

La asignatura considera tres factores para calcular la calificación final del alumno. Estos factores son:

- proyecto: la calificación (entre 0 y 10) de un proyecto práctico realizado en equipo. Incluye las actividades llevadas a cabo por el alumno a lo largo del cuatrimestre, tales como participación en clase, liderazgo del equipo, aportaciones al proyecto, entregas del proyecto, defensa pública de las entregas del proyecto, etc. Aquellos estudiantes que no se hayan incorporado a un equipo de proyecto en forma y plazo especificado por el profesor no podrán realizar el proyecto de la asignatura.
- defensa: la calificación (entre 0 y 10) de una prueba que evalúa los conocimientos adquiridos por los alumnos a través de la realización del proyecto de la asignatura
- contenidos: la calificación (entre 0 y 10) de un examen que evalúa los contenidos de la asignatura.

Los valores de dichos factores pueden obtenerse en las siguientes convocatorias:

- proyecto: diversas entregas hasta final del cuatrimestre. Si proyecto < 5 en convocatoria ordinaria, el equipo podrá realizar una entrega del proyecto en convocatoria extraordinaria.
- contenidos, defensa: convocatoria ordinaria y extraordinaria.

La calificación final del alumno (en la convocatoria ordinaria o en la extraordinaria) será:

- $0,4 * \text{proyecto} + 0,3 * \text{defensa} + 0,3 * \text{contenidos}$ , si  $((\text{defensa} + \text{contenidos}) / 2) \geq 5$  y  $\text{proyecto} \geq 5$
- suspenso, si  $((\text{defensa} + \text{contenidos}) / 2) < 5$  o  $\text{proyecto} < 5$

**Actividades docentes:**

Reparto de créditos:

- Teoría: 2,50
- Problemas: 0,50
- Laboratorios: 1,50

Otras actividades:

- Clases teóricas
- Sí, de exposición de los principales conceptos teóricos de la asignatura y resolución de casos prácticos.
- Seminarios
- Sí, sobre temas de especial relevancia e impartidos por expertos del área.
- Clases prácticas
- Sí
- Laboratorios
- Sí, incluyendo trabajos tanto individuales como en equipo.
- Exposiciones
- Sí, a determinar.
- Presentaciones
- Sí, a determinar.
- Presenciales.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

**Bibliografía:**

- Pressman, R.S., Maxim, B.R. Software Engineering. A Practitioner's Approach, McGraw-Hill, 2015.  
Sommerville I., Software Engineering. Tenth Edition. Pearson Education, 2016.  
R. Pressman: Ingeniería del Software - Un enfoque práctico, 7ª edición. McGraw-Hill, 2010.  
I. Sommerville: Ingeniería del Software, 9ª edición. Addison Wesley, 2016.  
J. Arlow, I. Neudstadt: UML 2. Anaya Multimedia, 2006.  
I. Jacobson, G. Booch, J. Rumbaugh: El proceso unificado de desarrollo de software. Addison-Wesley, 2000.  
R. C. Martin: Agile Software Development - Principles, Patterns, and Practices. Pearson Education, 2011.

Ficha docente guardada por última vez el 22/06/2023 16:00:00 por el usuario: Coordinador GIS

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:

**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 2º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805484 - Ingeniería del Software II		<b>Abrev:</b> IS2	<b>4,5 ECTS</b>
<b>Asignatura en Inglés:</b> Software Engineering II		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Desarrollo del software fundamental		<b>15 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>			
Bases de datos		6 ECTS	
Ingeniería del Software I		4,5 ECTS	
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Interdepartamental ISIA / SIC		<b>Coordinador:</b> Rabanal Basalo, Pablo Manuel	

**Descripción de contenidos mínimos:**

Diseño de software: Modelado estructural y modelado del comportamiento.  
Implementación y validación.  
Mantenimiento de aplicaciones.  
Práctica de la ingeniería del software.

**Programa detallado:**

Modelado de software. Introducción a UML.  
Análisis de software.  
Diseño de software. Patrones de diseño.  
Implementación y validación.  
Mantenimiento y evolución del software.

**Programa detallado en inglés:**

Modeling software. Introduction to UML.  
Software analysis.  
Software design. Design patterns.  
Implementation and validation.  
Software maintenance and evolution.

**Competencias de la asignatura:****Generales:**

- CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- CG8-Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.
- CG17-Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.
- CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
- CG20-Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.
- CG22-Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Resultados de aprendizaje:**

- Analizar el impacto económico y en el cliente de un sistema informático. (CG8, CT5)
- Analizar los recursos necesarios para la ejecución de un proyecto. (CG8, CG20)
- Aplicar las capacidades de comunicación para exponer de forma organizada y clara los distintos aspectos del trabajo en el proyecto. (CG22)
- Aplicar las capacidades de comunicación, comprensión y síntesis para integrar y resolver las distintas perspectivas sobre el proyecto. (CG22, CT2)
- Conocer los servicios básicos que un sistema gestor de bases de datos puede prestar a una aplicación. (CG3, CG17, CG18)
- Conocer los servicios básicos que un sistema operativo puede prestar a una aplicación. (CG3)
- Diferenciar las distintas perspectivas y necesidades de comunicación según los participantes en un proyecto de desarrollo de un sistema informático. (CT1, CT2)
- Dominar la abstracción procedimental (CG3)
- Dominar la orientación a objetos, herencia, polimorfismo y vinculación dinámica. (CG3)
- Modificar las técnicas y recursos disponibles para adaptarlos a las necesidades específicas del desarrollo de un sistema informático. (CT3)
- Modificar un proceso de desarrollo software para adecuarlo a las necesidades específicas de un proyecto. (CG20)
- Planear la planificación de un proyecto en base a sus restricciones y recursos. (CG8, CT4)
- Planear las necesidades en recursos humanos de la ejecución de un proyecto de desarrollo software, así como su ejecución (CT4)
- Realizar el desarrollo de un sistema informático en un proyecto de varios meses de duración en un equipo de desarrollo con más de 5 personas. (CG17, CG20, CG22, CT1, CT2, CT3, CT4, CT5)
- Relacionar el éxito del proyecto con la motivación y toma decisiones adecuada del personal (CG22, CT4)
- Seleccionar y combinar críticamente las alternativas disponibles para abordar un aspecto del desarrollo de un sistema informático. (CT2, CT3)
- Usar los medios audiovisuales para expresar la información de un proyecto de sistema informático de forma apropiada para su comprensión por el cliente y el equipo de desarrollo. (CT1)
- Valorar el proceso de desarrollo software más adecuado para un proyecto. (CG20)

**Evaluación detallada:**

- La asignatura considera tres factores para calcular la calificación final del alumno. Estos factores son:
- proyecto: la calificación (entre 0 y 10) de un proyecto práctico realizado en equipo. Incluye las actividades llevadas a cabo por el alumno a lo largo del cuatrimestre, tales como participación en clase, liderazgo del equipo, aportaciones al proyecto, entregas del proyecto, defensa pública de las entregas del proyecto, etc. Aquellos estudiantes que no se hayan incorporado a un equipo de proyecto en forma y plazo especificado por el profesor no podrán realizar el proyecto de la asignatura.
  - defensa: la calificación (entre 0 y 10) de una prueba que evalúa los conocimientos adquiridos por los alumnos a través de la realización del proyecto de la asignatura
  - contenidos: la calificación (entre 0 y 10) de un examen que evalúa los contenidos de la asignatura.

Los valores de dichos factores pueden obtenerse en las siguientes convocatorias:

- proyecto: diversas entregas hasta final del cuatrimestre. Si proyecto < 5 en convocatoria ordinaria, el equipo podrá realizar una entrega del proyecto en convocatoria extraordinaria.
- contenidos, defensa: convocatoria ordinaria y extraordinaria.

La calificación final del alumno (en la convocatoria ordinaria o en la extraordinaria) será:

- $0,4 * \text{proyecto} + 0,3 * \text{defensa} + 0,3 * \text{contenidos}$ , si  $((\text{defensa} + \text{contenidos}) / 2) \geq 5$  y  $\text{proyecto} \geq 5$
- suspenso, si  $((\text{defensa} + \text{contenidos}) / 2) < 5$  o  $\text{proyecto} < 5$

**Actividades docentes:**

Reparto de créditos:  
Teoría: 1,25  
Problemas: 0,25  
Laboratorios: 3,00

Otras actividades:  
Clases teóricas  
Sí, de exposición de los principales conceptos teóricos de la asignatura y resolución de casos prácticos.  
Seminarios  
Sí, sobre temas de especial relevancia e impartidos por expertos del área.  
Clases prácticas  
Sí

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Laboratorios  
Sí, incluyendo trabajos tanto individuales como en equipo.  
Exposiciones  
Sí, a determinar.  
Presentaciones  
Sí, a determinar.  
Presenciales.

**Bibliografía:**

- Pressman, R.S., Maxim, B.R. Software Engineering. A Practitioner's Approach, McGraw-Hill, 2015.  
Sommerville I., Software Engineering. Tenth Edition. Pearson Education, 2016.  
R. Pressman: Ingeniería del Software - Un enfoque práctico, 7ª edición. McGraw-Hill, 2010.  
I. Sommerville: Ingeniería del Software, 9ª edición. Addison Wesley, 2016.  
J. Arlow, I. Neudstadt: UML 2. Anaya Multimedia, 2006.  
I. Jacobson, G. Booch, J. Rumbaugh: El proceso unificado de desarrollo de software. Addison-Wesley, 2000.  
R. C. Martin: Agile Software Development - Principles, Patterns, and Practices. Pearson Education, 2011.  
D. Alur, J. Crupi, D. Malks: Core J2EE Patterns: Best Practices and Design Strategies, 2nd Edition. Prentice-Hall PTR, 2007.  
E. Gamma, R. Helm, R. Johnson, J. Vlissides: Patrones de diseño. Addison Wesley, 2003.

Ficha docente guardada por última vez el 23/06/2023 16:06:00 por el usuario: **Coordinador GIS**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID  
FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 3º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803208 - Sistemas operativos		<b>Abrev:</b> SO	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Operating Systems		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Sistemas operativos y redes fundamentales		<b>12 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Redes		<b>6 ECTS</b>	
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Tenllado Van der Reijden, Christian	

**Descripción de contenidos mínimos:**

Gestión de Procesos: planificación y comunicación.  
Asignación de memoria dinámica y memoria virtual.  
Arquitectura del Sistema de E/S y drivers.  
Sistemas de ficheros y directorios.  
Interfaz de usuario y lenguajes de script.

**Programa detallado:**

TEORÍA

=====

Módulo 1. Introducción

- 1.1 Qué es un SO
- 1.2 Componentes del SO
- 1.3 Concepto de llamada al sistema
- 1.4 Arranque del SO
- 1.5 El shell Bash. Introducción a Bash scripting.

Módulo 2. Gestión de Ficheros

- 2.1 Ficheros
  - 2.1.1 Concepto de ficheros.
  - 2.1.2 Denominación. Estructura. Tipos. Atributos
  - 2.1.3 Operaciones sobre ficheros
- 2.2 Directorios
  - 2.2.1 Concepto de directorio
  - 2.2.2 Jerarquía. Ruta absoluta y relativa
  - 2.2.3 Operaciones sobre directorios
- 2.3 Sistema de Ficheros
  - 2.3.1 Estructura de un Sistema de Ficheros
  - 2.3.2 Tablas de acceso y relación con descriptor de fichero
  - 2.3.3 Administración del espacio de disco
  - 2.3.4 Rendimiento. Cache de bloques

Módulo 3. Gestión de Procesos

- 3.1 Concepto de proceso.
  - 3.1.1 Creación y finalización
  - 3.1.2 Modelo Jerárquico
  - 3.1.3 Estados de un proceso
  - 3.1.4 Estructuras de datos básicas para su gestión
- 3.2 Planificación
  - 3.2.1 Concepto de planificador
  - 3.2.2 Algoritmos básicos: FCFS, con prioridad, round-robin
- 3.3 Threads
  - 3.3.1 Concepto de thread
  - 3.3.2 Estructura de una aplicación multithread
  - 3.3.3 Implementación de los threads. Biblioteca POSIX Threads
- 3.4 Sincronización y Comunicación
  - 3.4.1 Concepto de carrera y definición de sección crítica
  - 3.4.2 Exclusión mutua
  - 3.4.3 Problemas clásicos de programación concurrente
  - 3.4.4 Semáforos, cerrojos y variables condicionales
  - 3.4.5 Soporte hardware para implementación de primitivas de sincronización

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Módulo 4. Gestión de entrada/salida

4.1 Arquitectura del sistema de E/S

4.1.1 Recordatorio de técnicas HW de E/S

4.1.2 Modelo de un dispositivo en LINUX. Anatomía de un driver

4.1.3 Caso de dispositivos: de bloque (disco), de caracteres (terminal, impresora)

Módulo 5. Gestión de memoria

5.1 Introducción a la gestión de memoria

5.1.1 Espacios de direcciones lógico y físico

5.1.2 Reubicación

5.2 Multiprogramación. Particiones fijas y particiones variables

5.3 Memoria Virtual (MV)

5.3.1 Concepto de MV paginada. Políticas de carga y sustitución

5.3.2 Diseño de sistemas paginados. Políticas de asignación y carga

5.3.3 Implementación de sistemas paginados. Excepción de "fallo de página"

5.4 Regiones de memoria de un proceso

5.4.1 Estructura y generación de un ejecutable

5.4.2 Operaciones sobre regiones

LABORATORIO

=====

Se realizarán 7 sesiones laboratorio en las que los alumnos afianzarán su conocimiento sobre el API POSIX y la programación de sistemas, realizando prácticas sobre sistemas de ficheros, gestión de procesos e hilos, sincronización y E/S.

**Programa detallado en inglés:**

THEORY

=====

Unit 1. Introduction

1.1 What is an OS

1.2 Components of an operating system

1.3 System calls

1.4 Booting up the system

1.5 The BASH shell. Introduction to BASH scripting

Unit 2. File Management

2.1 Files

2.1.1 Concept of file

2.1.2 Naming. Structure. File types. File attributes.

2.1.3 File operations

2.2 Directories

2.2.1 Concept of directory

2.2.2 Hierarchy. Absolute and relative paths

2.2.3 Operations on directories

2.3 File Systems

2.3.1 Structure of a file system.

2.3.2 Tables in a file system and file descriptors

2.3.3 Disk space management

2.3.4 Performance. Buffer Cache

Unit 3. Process management

3.1 Concept of process

3.1.1 Process life cycle

3.1.2 Hierarchical Model

3.1.3 States of a process

3.1.4 Basic data structures for process management

3.2. Process Scheduling

3.2.1 Introduction to the OS scheduler

3.2.2 Scheduling algorithms: FCFS. priority, round-robin

3.3 Threads

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



- 3.3.1 Concept of thread
- 3.3.2 Structure of a multithreaded application
- 3.3.3 Implementation of threads. POSIX Threads Library
- 3.4 Synchronization and Communication
  - 3.4.1 Race conditions and definition of critical section
  - 3.4.2 Mutual Exclusion
  - 3.4.3 Classical problems in concurrent programming
  - 3.4.4 Semaphores, locks and condition variables
  - 3.4.5 Hardware support for the implementation of synchronization primitives

Unit 4. Input / Output management

- 4.1 Architecture of the I/O system
  - 4.1.1 Technical reminder of I/O hardware
  - 4.1.2 The LINUX device model. Anatomy of a device driver
  - 4.1.3 Types of devices: block (disk), character (terminal, printer)

Unit 5. Memory Management

- 5.1 Introduction to memory management
  - 5.1.1 Logical and physical addresses
  - 5.1.2 Relocation
- 5.2 Multiprogramming. Fixed and variable partitions
- 5.3 Virtual Memory
  - 5.3.1 Paging. Replacement policies
  - 5.3.2. Design of paging systems
  - 5.3.3 Implementation of paging systems. Handling page faults
- 5.4 Memory regions of a process
  - 5.4.1 Structure and generation of an executable file
  - 5.4.2 Operations on regions

LAB

====

The course will have 7 lab sessions in which the students will get familiar with the POSIX API and system programming in general, working on assignments on several topics: file systems, process and thread management, synchronization and I/O.

**Competencias de la asignatura:**

**Generales:**

- CG10-Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.
- CG15-Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.

**Resultados de aprendizaje:**

- Comprender la estructura de un sistema de ficheros y diseñar uno simple (CG15, CT2, CT4)
- Conocer el papel del sistema operativo en un sistema digital (CG10, CG15)
- Conocer las herramientas administrativas para gestión de drivers y módulos del kernel (CG10)
- Diferenciar los conceptos de proceso e hilo (CG15)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMÁTICA**

- Diseñar un planificador de tareas (CG15, CT2, CT3, CT4)
- Entender y saber aplicar los mecanismos de sincronización y comunicación entre procesos e hilos (CG15, CT2)
- Implementar aplicaciones usando llamadas al sistema POSIX (CG15)
- Implementar un módulo de kernel capaz de interactuar con dispositivos de E/S (CG10, CG15, CT1, CT3)
- Utilizar lenguajes de scripting (bash) para la automatización de tareas (CG10)

**Evaluación detallada:**

Para la evaluación se tienen en cuenta los siguientes elementos:

1. Nota de Laboratorio: habrá un examen de laboratorio en cada convocatoria, el mismo día que el examen de teoría. Se realizarán diferentes ejercicios prácticos.
2. Examen de Teoría. Habrá examen final teórico en ambas convocatorias, común para todos los grupos de la asignatura, que será escrito y estará formado por cuestiones teórico-prácticas.
3. Pruebas de clase: realización de actividades propuestas por el profesor en clase, como la resolución de problemas, la realización de partes opcionales de las prácticas, controles, etc.

La nota final será la mayor de las dos puntuaciones siguientes:

- Nota del examen de teoría \* 0,60 + Nota del examen de laboratorio \* 0,30 + Nota de pruebas de clase \* 0,10
- Nota del examen de teoría \* 0,70 + Nota del examen de laboratorio \* 0,30

La asignatura se considerará aprobada si la nota final es igual o superior a 5 y además se ha obtenido una nota igual o superior a 4 (sobre 10) en ambos exámenes. En caso de haber obtenido una nota inferior a 4 en cualquiera de los exámenes, la media saturará en 4 y se considerará suspenso.

**Actividades docentes:**

Reparto de créditos:

- Teoría: 3,00
- Problemas: 1,50
- Laboratorios: 1,50

Otras actividades:

- Clases teóricas en promedio 3 horas a la semana. Incluye teoría y problemas (en aula).
- Clases prácticas en promedio 1 hora a la semana (en laboratorio)

**Bibliografía:**

Bibliografía Básica

- Jesus Carretero, Sistemas Operativos – una visión aplicada. McGraw-Hill. 2007
- Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau. Operating Systems: Three Easy Pieces. Arpaci-Dusseau Books. <http://pages.cs.wisc.edu/~remzi/OSTEP>. 2015
- W. Stallings. Operating Systems. Internals and Design Principles. 7th Ed. Prentice Hall. 2012
- Andrew S. Tanenbaum, Albert S. Woodhull. Modern Operating Systems, 3rd Ed. Prentice Hall. 2006

Bibliografía Complementaria

- Abraham Silberschatz, Greg Gagne, Peter B. Galvin. Operating System Concepts. 8th Ed. Wiley. 2011<sup>[1]</sup><sub>[SEP]</sub>
- Neil Matthew, Richard Stones. Beginning Linux Programming. 4th Ed. Wiley. 2007
- Mark Mitchell et al. Advanced Linux Programming. New Riders Publishing. [http://richard.esplins.org/static/downloads/linux\\_book.pdf](http://richard.esplins.org/static/downloads/linux_book.pdf). 2001.
- Machtelt Garrels. Bash Guide for Beginners. <http://www.tldp.org/LDP/Bash-Beginners-Guide/Bash-Beginners-Guide.pdf>. 2008

Ficha docente guardada por última vez el 28/06/2022 10:34:00 por el usuario: Coordinador GIC

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 3º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803209 - Redes		<b>Abrev:</b> RED	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Computer Networks		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Sistemas operativos y redes fundamentales		<b>12 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Sistemas operativos		<b>6 ECTS</b>	
<b>Módulo:</b> Materias comunes a la rama de la informática			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Moreno Vozmediano, Rafael	

**Descripción de contenidos mínimos:**

Técnicas y medios de transmisión de datos.  
Protocolos de enlace y redes de área local.  
Protocolos de red y encaminamiento.  
Protocolos de transporte.  
Arquitectura TCP/IP e Internet.

**Programa detallado:**

Módulo 1. Introducción a las redes  
1.1. Tipos de redes  
1.2. Arquitectura de red  
1.3. Ejemplos de arquitecturas de red: Modelo OSI y TCP/IP

Módulo 2. Conceptos de transmisión de datos  
2.1. Datos y señales  
2.2. Ancho de banda y velocidad de transmisión  
2.3. Transmisión analógica y digital  
2.4. Multiplexación  
2.5. Medios de transmisión

Módulo 3. Infraestructuras de red  
3.1. Conexiones punto a punto  
3.2. Redes de área local (LAN)  
3.3. Redes de área extensa (WAN)  
3.4. Tecnologías de acceso residencial

Módulo 4. La capa de red. Protocolo IP  
4.1. Introducción a las funciones de red: encaminamiento y congestión  
4.2. Protocolo IP  
4.3. Redes, subredes y superredes  
4.4. Protocolo ARP  
4.5. Protocolo ICMP  
4.6. Introducción al encaminamiento en IP

Módulo 5. La capa de transporte e introducción a los servicios y aplicaciones de red  
5.1. Modelo cliente-servidor  
5.2. Los protocolos de transporte TCP y UDP  
5.3. Introducción a los servicios básicos de red y protocolos de aplicación  
5.4. Introducción a la seguridad  
5.5. Introducción a los sistemas distribuidos

**Programa detallado en inglés:**

Module 1. Introduction to computer networks  
1.1. Types of computer networks  
1.2. Network architectures  
1.3. Examples of network architectures: OSI and TCP/IP

Module 2. Data transmission concepts  
2.1. Data and signals  
2.2. Bandwidth and bit rate  
2.3. Analog and digital transmission  
2.4. Multiplexing  
2.5. Transmission media

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Módulo 3. Network infrastructures

- 3.1. Point to point connections
- 3.2. Local area networks (LAN)
- 3.2. Wide area networks (WAN)
- 3.3. Residential access technologies

Module 4. Network layer. Internet Protocol (IP)

- 4.1. Introduction to network layer functions: routing and congestion control
- 4.2. Internet Protocol (IP)
- 4.3. Networks, subnetworks and supernetworks
- 4.4. Address Resolution Protocol (ARP)
- 4.5. Internet Control Message Protocol (ICMP)
- 4.6. Introduction to IP routing

Module 5. Transport layer and introduction to network services and applications

- 5.1. Client-server model
- 5.2. User Datagram Protocol (UDP) and transmission Control Protocol (TCP)
- 5.3. Introduction to basic network services and application protocols
- 5.4. Introduction to security
- 5.5. Introduction to distributed systems

**Competencias de la asignatura:**

**Generales:**

CG10-Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.

CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Analizar y comparar distintas configuraciones de red, seleccionando la configuración más adecuada entre las posibles (CT3)

Analizar y resolver problemas teóricos y prácticos de redes de computadores (CT2, CT3)

Comprender la función de los elementos de la arquitectura de una red (CG16)

Conocer los conceptos básicos de transmisión de datos en redes de computadores (CG16)

Conocer los principales protocolos de red y los servicios y aplicaciones básicas ofrecidos por las redes de computadores (CG10, CG16)

Diseñar y administrar una configuración básica de red (CG10)

**Evaluación detallada:**

En ambas convocatorias (ordinaria y extraordinaria) se calculará la nota como resultado de la evaluación de dos partes:

- Teoría (85% de la nota): Se evaluará mediante un examen final que incluirá cuestiones teóricas y problemas, sobre los contenidos explicados en clase.

- Laboratorio (15% de la nota): Para aprobar esta parte, el alumno deberá asistir de forma obligatoria a todas las sesiones de laboratorio y realizar correctamente una práctica final. En caso de no asistir al laboratorio o no superar la práctica final, el alumno podrá recuperar la nota mediante un examen escrito que incluirá cuestiones relacionadas con las prácticas de laboratorio y que se realizará a continuación del examen de teoría.

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,30

Problemas: 0,70

Laboratorios: 1,00

Otras actividades:

- Enseñanza presencial teórica

- Enseñanza presencial de ejercicios y supuestos prácticos

- Enseñanza presencial de prácticas de laboratorio

- Realización de prácticas no tutorizadas

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





**Bibliografía:**

- Forouzan, B., "Transmisión de datos y redes de comunicaciones", MacGraw-Hill, 4ª ed., 2006
- Forouzan B. "TCP/IP Protocol Suite", McGraw-Hill, 4ª ed., 2010
- Stallings, William, "Comunicaciones y Redes de Computadores", Pearson-Prentice Hall, 7a ed. , 2004 (En Inglés, "Data and Computer Communications", 9th edition)
- Halsall, Fred, "Redes de Computadores e Internet", Pearson-Addison Wesley, 5ª ed., 2006
- Kurose, J., Ross, K., "Redes de computadoras, un enfoque descendente", Pearson, 5ª ed., 2010

Ficha docente guardada por última vez el 22/06/2023 12:58:00 por el usuario: Coordinador GIC

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> 3º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803214 - Ampliación de Sistemas Operativos	<b>Abrev:</b> ASO	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Advanced Topics in Operating Systems	<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Sistemas Operativos y Redes Avanzados	<b>12 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Ampliación de Redes	<b>6 ECTS</b>	
<b>Módulo:</b> Ingeniería de computadores		
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Prieto Matias, Manuel

**Descripción de contenidos mínimos:**

Seguridad y protección.  
Tipos de sistemas operativos: servidor, desktop, empotrado.  
Proceso de arranque y configuración del sistema.  
Diseño e implementación de software de sistema.  
Aspectos multicore del sistema operativo.

**Programa detallado:**

Teoría

=====

## Módulo 1: Introducción

- 1.- Estructura y recursos del sistema
- 2.- Arquitectura del sistema operativo Linux
- 3.- Interfaz de llamadas al sistema
- 4.- Códigos de error y gestión de errores
- 5.- Llamadas al sistema vs funciones de biblioteca
- 6.- Introducción a la seguridad y la protección

## Módulo 2: Gestión Avanzada de Sistemas de Ficheros

- 1.- Arquitectura del sistema de ficheros
- 2.- Manejo avanzado de ficheros
- 3.- Manejo avanzado de directorios

## Módulo 3: Gestión Avanzada de Procesos y Memoria

- 1.- Estructura e Información de procesos
- 2.- Ejecución de Programas
- 3.- Control de procesos y Planificación
- 4.- Gestión de memoria
- 5.- Gestión de señales y temporizadores

## Módulo 4: Comunicación entre Procesos

- 1.- Comunicación mediante tuberías (pipes)
- 2.- Mecanismos IPC del UNIX System V
- 3.- Semáforos
- 4.- Memoria Compartida
- 5.- Colas de Mensajes

## Módulo 5: Kernel

- 1.- Visión global del Kernel

Laboratorio

=====

## Módulo 1. Programación de sistemas

## Módulo 2. Gestión avanzada de sistemas de ficheros

## Módulo 3. Gestión avanzada de procesos y señales, y diseño de un shell

## Módulo 4. Comunicación y sincronización con tuberías e IPC

**Programa detallado en inglés:**

Contents

=====

## Module 1: Introduction

- 1.- System organization and resources
- 2.- Linux operating system architecture
- 3.- System call interface
- 4.- Error management
- 5.- System calls vs library functions
- 6.- An introduction to system security and protection

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Module 2: Advanced File System Management

- 1.- File system architecture
- 2.- Advanced file management
- 3.- Advanced directory management

Module 3: Advanced Process and Memory Management

- 1.- Process structure
- 2.- Program execution
- 3.- Process management and scheduling
- 4.- Memory management
- 5.- Signal and timer management

Module 4: Interprocess Communication

- 1.- Pipes
- 2.- IPC Mechanisms
- 3.- Semaphores
- 4.- Shared Memory
- 5.- Message Queues

Module 5: Kernel

- 1.- Kernel internals

Lab

=====

Module 1. System programming

Module 2. Advanced management of file systems

Module 3. Advanced management of processes and signals, and shell design

Module 4. Interprocess communication with IPCs

**Competencias de la asignatura:**

**Generales:**

CG10-Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.

CG15-Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios.

CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

**Específicas:**

CE\_GIC4-Capacidad de diseñar e implementar software de sistema y de comunicaciones.

CE\_GIC6-Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos.

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.

**Resultados de aprendizaje:**

Conocer las estructuras internas del kernel Linux para la gestión de ficheros, proceoss y dispositivos. (CG15, CT1)

Conocer los mecanismos para la gestión eficiente y robusta de procesos e hilos en un sistema POSIX. (CT3)

Conocer y hacer uso de las distintas alternativas ofrecidas por los sistemas POSIX para la comunicación y sincronización de procesos. (CT2, CT3, CE\_GIC4)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Diseñar e implementar un sistema gestor de procesos en un sistema POSIX. (CT2, CT3, CT4)
- Dominar las herramientas administrativas existentes para la gestión de sistemas de ficheros, procesos y mecanismos de comunicación entre procesos. (CG10)
- Entender y saber aplicar los principales mecanismos de gestión y control de errores en el desarrollo de software de sistema robusto y seguro. (CT2, CE\_GIC6)
- Implementar aplicaciones cooperativas usando mecanismos de comunicación/sincronización entre procesos IPC. (CG15, CT3, CE\_GIC4)
- Implementar una aplicación completa para la interacción con el sistema de ficheros basada en llamadas al sistema. (CT3, CE\_GIC4)
- Realizar prácticas de programación en equipo. (CT1)

**Evaluación detallada:**

Asistencia a laboratorio obligatoria (70% mínimo de asistencia)

Convocatoria ordinaria:

Prácticas y trabajo personal: 40%

Examen final: 60%

Convocatoria extraordinaria:

Prácticas y trabajo personal: 30% (calificación obtenida en la convocatoria ordinaria.)

Examen final: 70%

La asignatura se considerará aprobada si la media ponderada de prácticas y examen final es igual o superior a 5 y además se ha obtenido una nota igual o superior a 4 en el examen final.

**Actividades docentes:**

Reparto de créditos:

Teoría: 1,50

Problemas: 0,00

Laboratorios: 4,50

Otras actividades:

- Enseñanza presencial teórica

- Enseñanza presencial de ejercicios y supuestos prácticos

- Enseñanza presencial de prácticas de laboratorio

- Realización de prácticas no tutorizadas

**Bibliografía:**

- Mark Mitchell, Jeffrey Oldham, and Alex Samuel, "Advanced Linux Programming", CodeSourcery LLC

(<https://mentoreembedded.github.io/advancedlinuxprogramming/>)

- Robert Love, "Linux System Programming", 2ª edición O'Reilly Media, Inc (disponible online en la biblioteca UCM)

- Richard Stevens, Stephen A. Rago, "Advanced Programming in the UNIX Environment", 2ª Edición, Addison-Wesley

- Robert Love; Linux Kernel Development. Addison-Wesley. 3rd Edition. Julio 2010.

- Michael Kerrisk. The Linux Programming Interface. A Linux and UNIX System Programming Handbook. No Starch Press, ISBN 978-1-59327-220-3

- Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau. Operating Systems: Three Easy Pieces. Arpaci-Dusseau Books.

<http://pages.cs.wisc.edu/~remzi/OSTEP>

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> 3º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803215 - Ampliación de Redes	<b>Abrev:</b> AR	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Advanced Computer Networks	<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Sistemas Operativos y Redes Avanzados	<b>12 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Ampliación de Sistemas Operativos	<b>6 ECTS</b>	
<b>Módulo:</b> Ingeniería de computadores		
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Fabero Jiménez, Juan Carlos

**Descripción de contenidos mínimos:**

Internet de nueva generación (IPv6).  
Protocolos de encaminamiento en Internet.  
Protocolos y servicios de red avanzados.  
Configuración y evaluación de servidores.  
Seguridad en redes.

**Programa detallado:**

## Módulo 1. Conceptos avanzados de TCP

- 1.1. Repaso de TCP.
- 1.2. Control de errores en TCP y temporizadores de retransmisión.
- 1.3. Control de flujo en TCP.
- 1.4. Control de congestión en TCP.
- 1.5. Ajuste de parámetros de TCP.
- 1.6. Programación con sockets.

## Módulo 2. Internet de nueva generación: IPv6

- 2.1. Repaso de IPv4 y comparación con IPv6.
- 2.2. Formato del datagrama IPv6. Cabeceras de extensión.
- 2.3. Direccionamiento IPv6.
- 2.4. ICMPv6.
- 2.5. Mecanismos de transición de IPv4 a IPv6.

## Módulo 3. Protocolos de encaminamiento en IPv4 e IPv6

- 3.1. Sistemas autónomos.
  - 3.1. RIP.
  - 3.2. OSPF.
  - 3.3. BGP.

## Módulo 4. Configuración de servicios de red

- 4.1. Configuración del servicio DHCP
- 4.2. Configuración de NAT
- 4.3. Configuración del servicio DNS
- 4.4. Configuración y evaluación de servidores

## Módulo 5. Introducción a la seguridad

- 5.1. Conceptos básicos sobre seguridad.
- 5.2. Técnicas de cifrado.
- 5.3. Firmas y certificados digitales. PKI.
- 5.4. Cortafuegos.
- 5.5. Redes Privadas Virtuales (VPN)

## PRÁCTICAS DE LABORATORIO

## Prácticas del Módulo 1

- Configuración de puertos TCP y técnicas de escaneo.
- Ajuste de parámetros de TCP
- Programación con sockets de aplicaciones cliente/servidor

## Prácticas del Módulo 2

- Configuración de IPv6.
- Uso de direcciones IPv6.
- Anuncio de prefijos.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Autoconfiguración.
- Túneles

Prácticas del Módulo 3

- Configuración de un Sistema Autónomo con encaminadores RIP y OSPF
- Configuración de encaminadores BGP para el intercambio de información de encaminamiento entre varios SA.

Prácticas del Módulo 4

- Configuración de servidores y clientes DHCP
- Configuración de un router con NAT y port forwarding
- Configuración de servidores DNS
- Configuración y monitorización de un servidor Web

Prácticas del Módulo 5

- Creación de una Autoridad Certificadora y uso de certificados
- Configuración de un cortafuegos.
- Configuración de una VPN

**Programa detallado en inglés:**

Module 1. TCP advanced concepts

- 1.1. Review of TCP basics.
- 1.2. TCP error control and retransmission timers.
- 1.3. TCP flow control.
- 1.4. TCP congestion control.
- 1.5. Tuning TCP parameters.
- 1.6. Socket programming.

Module 2. New generation Internet: IPv6

- 2.1. Review of IPv4 basics and comparison with IPv6.
- 2.2. IPv6 packet format. Extension headers
- 2.3. IPv6 addressing.
- 2.4. ICMPv6.
- 2.5. Transition mechanisms IPv4-IPv6.

Module 3. Routing protocols in IPv4/IPv6

- 3.1. Autonomous systems.
  - 3.1. RIP.
  - 3.2. OSPF.
  - 3.3. BGP.

Module 4. Configuration of network services

- 4.1. DHCP service configuration
- 4.2. NAT configuration
- 4.3. DNS service configuration
- 4.4. Configuration and evaluation of servers

Module 5. Introduction to security

- 5.1. Security basics.
- 5.2. Encryption techniques.
- 5.3. Digital signatures and certificates. PKI.
- 5.4. Firewalls.
- 5.5. Virtual Private Networks (VPN)

**Competencias de la asignatura:**

**Generales:**

CG10-Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.

CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

**Específicas:**

CE\_GIC8-Capacidad para diseñar, desplegar, administrar y gestionar redes de computadores. y del resto de las actividades evaluables

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Analizar y resolver problemas y supuestos prácticos de configuración avanzada de redes (CT2, CE\_GIC8)

Conocer, usar y configurar los protocolos avanzados de red (IPv6) (CG10, CG16)

Conocer, usar y configurar los protocolos de encaminamiento en Internet (RIP, OSPF, BGP) (CG10, CG16)

Conocer, usar y configurar los protocolos de transporte en Internet (TCP) (CG10, CG16)

Conocer, usar y configurar los servicios fundamentales de Internet (DHCP, DNS, NAT, seguridad, ...) (CG10, CG16)

Diseñar, construir y administrar configuraciones complejas de red (CG10, CT3)

Experimentar con distintas configuraciones de red, comparar las distintas soluciones y seleccionar la más adecuada (CT3)

**Evaluación detallada:**

La realización de prácticas durante el curso (10%) sólo se tendrá en cuenta en la convocatoria ordinaria.

Convocatoria ordinaria = Examen final 90% (50% teoría, 40% práctica) + 10% asistencia al laboratorio y entregas

Convocatoria extraordinaria = Examen final 90% (45% teoría, 45% práctica).

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

- Enseñanza presencial teórica

- Enseñanza presencial de ejercicios y supuestos prácticos

- Enseñanza presencial de prácticas de laboratorio

- Realización de prácticas no tutorizadas

**Bibliografía:**

- B. Forouzan. "TCP/IP Protocol Suite", 4th ed. McGrawHill, 2010
- L. Parziale. "TCP/IP Tutorial and Technical Overview", 8th ed. IBM RedBooks. 2006.
- C. M. Kozierok. "The TCP/IP Guide", Versión 3.0. Recurso on-line (<http://www.tcpipguide.com>). 2005
- F. Halsall. "Redes de Computadores e Internet". 5ª ed. Addison-Wesley. 2006.
- B. Sosinsky. "Networking Bible". 1ª ed. Wiley Publishing. 2009.
- E. Cole. "Network Security Bible". 2ª ed. John Wiley & Sons. 2009.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 3º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803218 - Arquitectura de Computadores		<b>Abrev:</b> AC	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Computer Architecture		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Tecnología y Arquitectura de Computadores		<b>24 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b>			
Electrónica		6 ECTS	
Sistemas empotrados		6 ECTS	
Tecnología de computadores		6 ECTS	
<b>Módulo:</b> Ingeniería de computadores			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Clemente Barreira, Juan Antonio	

**Descripción de contenidos mínimos:**

- Paralelismo a nivel de instrucción.
- Ejecución de varios threads.
- Arquitectura de multiprocesadores.
- Introducción a la programación de sistemas multiprocesador.
- Sincronización.
- Coherencia.
- Consistencia E/S y sistemas de almacenamiento.

**Programa detallado:**

## Módulo 1. Introducción y tendencias en arquitectura de computadores.

- Contexto de la asignatura
- Evolución tecnológica, binomio arquitectura-tecnología
- Consumo de energía
- Factores determinantes del coste
- Medidas de rendimiento

## Módulo 2. Paralelismo a nivel de instrucción y multithreading.

- Técnicas de compilación básicas.
- Planificación dinámica de instrucciones: dependencias y renombramiento de registros
- Técnicas de predicción de saltos.
- Ejecución especulativa.
- Técnicas de lanzamiento múltiple de instrucciones
- Arquitectura de procesadores superescalares fuera-de-orden
- Límites del paralelismo a nivel de instrucción.
- Multithreading: concepto y tipos

## Módulo 3. Paralelismo a nivel de datos.

- Concepto de arquitectura vectorial
- Instrucciones SIMD para procesamiento multimedia
- Unidades para procesamiento gráfico (GPUs)

## Módulo 4. Multiprocesadores

- Conceptos básicos de multiprocesamiento
- La red de interconexión
- Arquitecturas de memoria compartida centralizada.
- Coherencia de cache: protocolos.
- Arquitectura de memoria compartida distribuida.
- Coherencia basada en directorio.
- Sincronización. Primitivas de sincronización.
- Concepto de consistencia de memoria: modelos.

**Programa detallado en inglés:**

- o Module 1. Introduction and trends in computer architecture
  - Context of the course
  - Technological evolution. The technology-architecture interaction.
  - Energy consumption
  - Key components of cost.
  - Measuring performance

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- o Module 2. Instruction-level parallelism (ILP) and multithreading.  
Basic compilation techniques  
Dynamic instruction scheduling: dependences and register renaming  
Branch prediction  
Speculative execution  
Multiple issue techniques  
Limits of ILP  
Architecture of superscalar out-of-order processors  
Multithreading: concept and types
- o Module 3. Data-level parallelism  
Vector architecture  
SIMD instruction set extensions for multimedia  
Graphics processing units (GPUs)
- o Module 4. Multiprocessors  
Basic concepts of multiprocessing  
The interconnection network  
Centralized shared memory architectures  
Cache coherence: protocols.  
Distributed shared memory architectures  
Directory-based cache coherence  
Synchronization: primitives  
Concept of memory consistency: models

**Competencias de la asignatura:**

**Generales:**

CG19-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.

**Específicas:**

CE\_GIC3-Capacidad de analizar y evaluar arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, así como desarrollar y optimizar software para las mismas.

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Analizar las limitaciones de paralelismo a nivel de instrucciones y estudiar alternativas arquitectónicas para superar dichas limitaciones (CG19, CE\_GIC3)

Comprender el paralelismo a nivel de datos y las arquitecturas que lo explotan, así como las instrucciones vectoriales y los mecanismos de vectorización (CG19, CE\_GIC3)

Comprender la influencia mutua entre tecnología de circuitos integrados y diseño arquitectónico, así como sus repercusiones sociales y económicas (CE\_GIC3)

Comprender la organización de las arquitecturas con paralelismo a nivel de "thread", y analizar los diferentes mecanismos de cambio de "thread" (CE\_GIC3)

Comprender los mecanismos para gestionar la ejecución de instrucciones en desorden y la especulación de saltos, así como el incremento de la capacidad de procesamiento que se deriva de dichas técnicas (CE\_GIC3)

Conocer la organización de sistemas multiprocesador, y comprender los mecanismos para el intercambio de información entre los diferentes procesadores, así como analizar las alternativas para el diseño de su jerarquía de memoria (CE\_GIC3)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



Conocer la terminología propia de la arquitectura de computadores en lengua inglesa y manejar fuentes bibliográficas en dicha lengua (CT1, CE\_GIC3)

Conocer las medidas para expresar el rendimiento de computadores en diferentes contextos, con inclusión de los aspectos térmicos y económicos (CE\_GIC3)

Resolver problemas de Arquitectura de Computadores seleccionando la solución más adecuada entre las posibles. (CT2, CT3)

**Evaluación detallada:**

A lo largo del curso habrá una serie de entregas de ejercicios que servirán para complementar las sesiones de teoría en el aula.

\* Convocatoria ordinaria: Examen final obligatorio y escrito. La nota de esta convocatoria será: Nota examen final x 0,9 + Nota entrega ejercicios x 0,1.

\* Convocatoria extraordinaria: Examen final escrito. La nota de esta convocatoria será: Nota examen final x 0,9 + Nota entrega ejercicios x 0,1. La nota entrega de ejercicios será la obtenida en la convocatoria ordinaria.

Cualquier estudiante puede ser llamado por el profesor para que explique individualmente la solución aportada a un determinado ejercicio. En caso de observarse un notorio desconocimiento del razonamiento que lleva a la solución aportada, o de detectarse graves incoherencias en la explicación, el hecho podrá trasladarse al comité de actuación ante copias.

Mismo examen (prácticas y otros elementos de evaluación, en su caso) en todos los grupos y criterios detallados de puntuación comunes.

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,50

Problemas: 1,50

Laboratorios: 0,00

Otras actividades:

Clases teóricas: 3 horas de clases teóricas a la semana en aula.

Clases prácticas: 1 hora de clase práctica (problemas, discusión, ...) a la semana en aula.

Total horas presenciales: 4h. (40%)

**Bibliografía:**

Básica:

- Hennessy, J. L., Patterson, D.A., "Computer Architecture: A Quantitative Approach", 6ª Ed., Elsevier, 2018.

- Culler, D., Singh, J., "Parallel Computer Architecture: A Hardware/Software Approach", Morgan Kaufmann, 1999

Complementaria:

- Dubois, M., Annavaram, M., Stenström, P. "Parallel Computer Organization and Design", Cambridge University Press, 2012

- González, A., Latorre, F., Magklis, G., "Processor Microarchitecture - An Implementation Perspective", Morgan & Claypool Publishers, 2011.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> 3º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803221 - Programación de sistemas y dispositivos	<b>Abrev:</b> PSyD	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Embedded Systems Programming	<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Software de Sistemas		<b>6 ECTS</b>
<b>Otras asignaturas en la misma materia:</b> No hay		
<b>Módulo:</b> Ingeniería de computadores		
<b>Departamento:</b> Arquitectura de Computadores y Automática		
<b>Coordinador:</b> Mendías Cuadros, José Manuel		

**Descripción de contenidos mínimos:**

- Programación de controladores de dispositivos.
- Desarrollo de software empujado.
- Desarrollo de software para dispositivos móviles.
- Desarrollo de software para sistemas dedicados o especializados.
- Introducción a los sistemas en tiempo real.
- Programación práctica de sistemas y dispositivos.

**Programa detallado:**

Introducción a la programación de sistemas empujados. Descripción del puesto de trabajo: el microcontrolador S3C44BOX, la placa de prototipado Embest S3CEV40 y el entorno de desarrollo Eclipse IDE for C/C++ Developers. Programación a bajo nivel en C. Desarrollo de firmware y bootstrapping. Modelos de programación de software empujado: sistemas multi-estado, sistemas muestreados, sistemas de control, sistemas guiados por eventos y sistemas guiados por tiempo. Micro-kernels de tiempo real. Prácticas de laboratorio.

**Programa detallado en inglés:**

Introduction to Embedded Systems Programming. Working framework description: the S3C44BOX microcontroller, the Embest S3CEV40 prototyping board and the Eclipse IDE for C/C++ Developers. Low-level C programming. Firmware development and bootstrapping. Programming models for embedded software: multi-state systems, sampled systems, control systems, event-driven systems, time-driven systems. Real time Micro-kernels. Labs.

**Competencias de la asignatura:****Generales:**

CG19-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.

**Específicas:**

- CE\_GIC2-Capacidad de desarrollar procesadores específicos y sistemas empujados, así como desarrollar y optimizar el software de dichos sistemas.
- CE\_GIC5-Capacidad de analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empujadas y de tiempo real.

**Básicas y Transversales:**

- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

- Analizar, evaluar y seleccionar la arquitectura software más adecuada para una aplicación empujada de tiempo real. (CE\_GIC5)
- Desarrollar aplicaciones empujadas combinando C y ensamblador. (CE\_GIC2)
- Evaluar las prestaciones de una plataforma hardware-software. (CT2, CT5, CE\_GIC5)
- Justificar las decisiones de diseño aplicadas en la optimización de una aplicación empujada de tiempo real. (CG19, CE\_GIC2)
- Resolver ejercicios de programación de sistemas empujados de tiempo real. (CG19, CT2, CT3)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Seleccionar los elementos más idóneos a usar de una plataforma hardware para el desarrollo de una aplicación empotrada de tiempo real.  
(CE\_GIC2)

**Evaluación detallada:**

La evaluación es continua y la nota final se obtendrá sumando ponderadamente la nota obtenida por las siguientes actividades: prácticas (30% de la nota), proyecto (30% de la nota) y examen práctico (40% de la nota).

Las prácticas y el proyecto se realizan durante el periodo lectivo. En caso de no superar la asignatura en convocatoria ordinaria, las notas obtenidas en ambas actividades se guardan hasta la convocatoria extraordinaria.

La asistencia a clase es obligatoria (70% mínimo de asistencia). El estudiante que no cumpla este requisito será calificado con NP en ambas convocatorias, con independencia de la nota que obtenga en el resto de actividades evaluables de la asignatura.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Enseñanza presencial teórica. Enseñanza presencial de prácticas de laboratorio. Realización en grupo de proyectos tutorizados.

**Bibliografía:**

- Jonathan W. Valvano. Embedded microcomputer systems: real time interfacing. Cengage Learning, 3ª edición, 2012.
- David E. Simon. An Embedded Software Primer. Addison-Wesley, 1999
- Jean J. Labrosse. Embedded Systems Building Blocks, Complete and Ready-to-Use Modules in C. R&D Books, 2ª edición, 2000.
- Jean J. Labrosse. MicroC/OS-II. The Real-Time Kernel. CMP Books, 2ª edición, 2002
- Michael J. Pont. Embedded C. Addison-Wesley, 2002
- Michael J. Pont. Patterns for time-triggered embedded systems. Addison-Wesley, 2001
- Andrew Sloss, Dominic Symes, Chris Wright . ARM system developer's guide: designing and optimizing system software. Elsevier / Morgan Kaufman, 2004.
- Giorgio C. Buttazzo. Hard Real Time Computing Systems. Springer, 3º edición, 2011

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





# UNIVERSIDAD COMPLUTENSE DE MADRID

## FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 3º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803223 - Sistemas web		<b>Abrev:</b> SW	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Web Systems		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Desarrollo de Software Avanzado			<b>12 ECTS</b>
<b>Otras asignaturas en la misma materia:</b> Sistemas inteligentes			<b>6 ECTS</b>
<b>Módulo:</b> Complementario			
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> Bravo Agapito, Javier	

### Descripción de contenidos mínimos:

- Arquitectura de aplicaciones web.
- Lenguajes de presentación y estilo.
- Programación en el lado del cliente.
- Programación en el lado del servidor.
- Interfaces persona-computador.
- Accesibilidad y usabilidad en la web.

### Programa detallado:

- Introducción a las aplicaciones Web. Arquitectura de las aplicaciones web. Lenguajes y tecnologías de programación Web.
- Lenguajes: HTML5 Y CSS. Diseño adaptativo. Accesibilidad y usabilidad en la web.
- Programación en el lado del cliente: Javascript, jQuery, AJAX.
- Programación en el lado del servidor: PHP, acceso a bases de datos.
- Otras tecnologías web. J2EE, Bootstrap, frameworks MVC, nodejs

### Programa detallado en inglés:

- Introduction to Web Applications. Web architecture. Development languages and technologies for web-based development.
- Client side presentation: HTML, XML, CSS stylesheets. Responsive Design. Web accessibility and usability.
- Client-side programming: Javascript, jQuery, AJAX.
- Server-side programming: PHP, database access.
- Other web technologies: J2EE, Bootstrap, MVC frameworks, nodejs.

### Competencias de la asignatura:

#### Generales:

- CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
- CG21-Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.
- CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

#### Específicas:

No tiene

#### Básicas y Transversales:

- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.

#### Resultados de aprendizaje:

- Argumentar decisiones de diseño en el desarrollo de aplicaciones web complejas. (CT3)
- Combinar distintos patrones de diseño en el cliente y en el servidor para desarrollar webs que usen servicios de apoyo. (CG21, CT3)
- Comprender los riesgos de seguridad que pueden afectar a una aplicación web para poder diseñar las medidas de seguridad oportunas. (CG18)
- Conocer distintas tecnologías adicionales de servidor y cliente para poder expandir sus competencias en desarrollo web. (CG18)
- Construir documentos bien formados y correctos en HTML5 para el desarrollo de páginas web. (CG21)
- Desarrollar aplicaciones web que usen bases de datos para lograr la persistencia de los datos. (CG18)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Desarrollar páginas web completas enriquecidas con JavaScript para mejorar la interacción. (CG21)

Desarrollo en grupo de aplicaciones web complejas (CT2, CT3, CT4)

Diseñar páginas web usando CSS nivel 3 para organizar la información y modificar la apariencia de una página web. (CG21)

**Evaluación detallada:**

Para aprobar la asignatura, el alumno deberá superar distintas prácticas eliminatorias y una prueba final. La prueba final consistirá en un proyecto final desarrollado en grupo, aunque los alumnos pueden optar por realizar un examen práctico individual en lugar del proyecto en grupo (casos especiales justificados). Además, parte de la nota depende también de completar distintos ejercicios y actividades propuestos en el día a día de las clases.

La calificación final de la asignatura se calculará en base a la siguiente fórmula:

$$0.7*NPF + 0.2*NP + 0.1*NA$$

, siendo:

\* NPF: nota de la prueba final, proyecto final en grupo o nota del examen individual (casos especiales justificados).

\* NP: nota de las prácticas

\* NA: nota de actividades de participación.

**Prueba final**

\* Proyecto final en grupo: el profesor podrá requerir realizar una tutoría colectiva con los integrantes del grupo de práctica con el objetivo de evaluar la participación y contribuciones individuales de cada uno de los integrantes del grupo. De ser necesaria esta evaluación tendrá un peso del 40% en la nota individual de cada integrante del grupo y será necesario al menos obtener un 5 (sobre 10) en la entrevista.

\* Examen práctico individual: consistirá en dos pruebas, el desarrollo de un supuesto práctico (60%) y, en los casos que sea necesario, la defensa oral de la solución propuesta y de los conceptos de la asignatura que se abordan (40%). Para calcular la media es requisito necesario obtener al menos 5 puntos (sobre 10) en cada una de las pruebas del examen.

Las calificaciones obtenidas en cualquiera de las pruebas se mantendrán para la convocatoria extraordinaria. En la convocatoria extraordinaria se pueden volver a entregar las prácticas y el proyecto.

Para calcular la media es requisito necesario haber entregado y aprobado todas las prácticas y trabajos, así como la obtención de al menos un 5 (sobre 10) en la prueba final.

Para calcular la media es requisito necesario haber entregado y aprobado todas las prácticas y trabajos, así como la obtención de al menos un 5 en la prueba final.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Clases teóricas: Exposición de los principales conceptos teóricos de la asignatura y resolución de casos prácticos.

Prácticas en laboratorio conducentes al desarrollo de un proyecto de aplicación web.

Trabajo escrito y presentación del mismo en clase sobre tecnologías web.

**Bibliografía:**

**BIBLIOGRAFÍA BÁSICA**

- Recio García, Juan Antonio. "HTML5, CSS3 y JQUERY. Curso práctico". Editorial RA-MA. 2016.

- David Powers. "PHP 7 Solutions; Dynamic Web Design Made Easy". Apress, Berkeley, CA. 2019

**BIBLIOGRAFÍA COMPLEMENTARIA**

- Aubry, Christophe. "HTML5 y CSS3. Domine los estándares de las aplicaciones web". Ediciones ENI, 2020.

- Mark J. Collins. "Pro HTML5 with CSS, JavaScript, and Multimedia; Complete Website Development and Best Practices". Apress, Berkeley, CA. 2017

- Andy Budd/Emil Björklund. "CSS Mastery". Apress, Berkeley, CA. 2016

- Frank M. Kromann. "Beginning PHP and MySQL; From Novice to Professional". Apress, Berkeley, CA. 2018.

- Olivier Heurtel. "PHP y MySQL: Domine el desarrollo de un sitio web dinámico e interactivo". Ediciones ENI, 2020.

- Russ Ferguson. "Beginning JavaScript; The Ultimate Guide to Modern JavaScript Development". Apress, Berkeley, CA. 2019.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 3º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803225 - Diseño de algoritmos		<b>Abrev:</b> DA	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Algorithm design		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Complementos de Programación			<b>12 ECTS</b>
<b>Otras asignaturas en la misma materia:</b>			
Lenguajes de programación y procesadores de lenguaje			<b>6 ECTS</b>
<b>Módulo:</b> Complementario			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Fabregas Álfaro, Ignacio	

**Descripción de contenidos mínimos:**

Estructuras arbóreas avanzadas.  
Colas de prioridad y montículos.  
Grafos.  
Métodos voraces.  
Programación dinámica.  
Métodos de búsqueda en espacios de estados y en árboles de juegos.  
Algoritmos probabilísticos.  
Complejidad de problemas.

**Programa detallado:**

1. Análisis amortizado
2. Árboles de búsqueda avanzados
3. Colas con prioridad y montículos
4. Grafos
5. Estructuras de partición
6. Algoritmos voraces
7. Programación dinámica
8. Ramificación y acotación
9. Árboles de juego
10. Algoritmos probabilísticos
11. Complejidad de problemas

**Programa detallado en inglés:**

1. Amortized analysis
2. Advanced search trees
3. Priority queues and heaps
4. Graphs
5. Disjoint sets
6. Greedy algorithms
7. Dynamic programming
8. Branch and bound
9. Game trees
10. Probabilistic algorithms
11. Computational complexity

**Competencias de la asignatura:****Generales:**

CG12-Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.  
CG13-Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.  
CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

- Analizar un problema e inferir las estructuras de datos más apropiadas para representarlo (CG12, CG13, CT3)
- Analizar el coste computacional de las operaciones sobre estructuras de datos avanzadas (CG12)
- Diseñar soluciones a problemas utilizando estructuras de datos y métodos algorítmicos avanzados y analizar su coste (CG13)
- Seleccionar el método algorítmico más apropiado para resolver un problema y justificar la elección (CG13)
- Argumentar la idoneidad de las soluciones propuestas y valorar posibles mejoras (CG13, CT2, CT5)

**Evaluación detallada:**

Un 30% de la nota se obtendrá a lo largo del curso mediante la evaluación de las actividades prácticas. Las actividades propuestas para evaluación tendrán una fecha límite de entrega o realización. No habrá un periodo de entrega nuevo para la convocatoria extraordinaria. La asistencia a las clases prácticas es evaluable. La no asistencia en los días indicados para evaluación continua supondrá la pérdida de la calificación de la prueba. La evaluación puede incluir además la realización individual de prácticas y problemas, la exposición en público de las soluciones propias, la realización de mini-exámenes escritos, las tutorías obligatorias y cualquier otra que permita conocer el grado de conocimientos y destreza alcanzados por el alumno.

Un 70% de la nota se obtendrá mediante el examen de la convocatoria ordinaria o de la extraordinaria. Para poder aprobar la asignatura se requerirá al menos una calificación de 5 sobre 10 en dicho examen.

**Actividades docentes:**

- |                      |   |
|----------------------|---|
| Reparto de créditos: | Otras actividades:  |
| Teoría: 3,00         | Enseñanza presencial teórica.                               |
| Problemas: 1,50      | Realización de problemas con corrección colectiva en clase. |
| Laboratorios: 1,50   | Realización de prácticas tutorizadas en el laboratorio.     |
|                      | Tutorías individuales.                                      |

**Bibliografía:**

- G. Brassard, P. Bradley. Fundamentos de algoritmia, Prentice Hall, 1997.
- T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. Introduction to Algorithms, Second Edition, The MIT Press, 2001. Third Edition, The MIT Press, 2009.
- E. Horowitz, S. Sahni, D. Mehta. Fundamentals of Data Structures in C++, Computer Science Press, 1995.
- E. Horowitz, S. Sahni, S. Rajasekaran. Computer Algorithms, Computer Science Press, 1998. Second Edition, Silicon Press, 2007.
- N. Martí Oliet, Y. Ortega Mallén, J. A. Verdejo López. Estructuras de datos y métodos algorítmicos: ejercicios resueltos, Pearson/Prentice Hall, 2003. Segunda edición, Garceta, 2013.
- R. Neapolitan. Foundations of Algorithms, Fifth Edition, Jones and Bartlett Publishers, 2014.
- R. Peña, Algoritmos y estructuras de datos: Con programas verificados en Dafny, Garceta, 2019.
- M. A. Weiss. Estructuras de datos en Java., Cuarta edición, Pearson, 2013.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 3º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803226 - Lenguajes de programación y procesadores de lenguaje	<b>Abrev:</b> LPP	<b>6 ECTS</b>	
<b>Asignatura en Inglés:</b> Programming Languages and Compilers	<b>Carácter:</b> Obligatoria		
<b>Materia:</b> Complementos de Programación		<b>12 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Diseño de algoritmos		<b>6 ECTS</b>	
<b>Módulo:</b> Complementario			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Vado Vírveda, Rafael del	

**Descripción de contenidos mínimos:**

- Conceptos básicos del procesamiento de lenguajes.
- Sintaxis de los lenguajes de programación.
- Expresiones regulares y gramáticas.
- Análisis léxico y sintáctico.
- Autómatas finitos y con pila.
- Estructuras de control de flujo.
- Sistemas de tipos y tipos de datos.
- Abstracción de control y abstracción de datos.
- Generación de código: código nativo, máquinas virtuales, compiladores e intérpretes.
- Paradigmas de programación: imperativo, orientado a objetos, funcional, lógico, concurrente, de scripting.

**Programa detallado:**

1. Introducción a los lenguajes de programación y a los compiladores.
2. Autómatas finitos y análisis léxico.
3. Autómatas con pila, gramáticas y análisis sintáctico
4. Análisis de la semántica estática: ámbitos de definición. Sistemas de tipos.
5. Generación de código.
6. Máquinas virtuales

**Programa detallado en inglés:**

1. An introduction to programming languages and compilers.
2. Finite automatas and lexical analysis.
3. Pushdown automatas, grammars, and syntax analysis.
4. Semantic analysis, scopes, and types.
5. Code generation.
6. Virtual machines.

**Competencias de la asignatura:****Generales:**

- CG12-Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.
- CG13-Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Ser capaz de aplicar los conceptos y técnicas aprendidos en el contexto de otras asignaturas. (CT2)
- Combinar el uso de técnicas en la resolución de problemas. (CT3)
- Realizar ejercicios. (CT1)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Conocer el proceso de generación y optimización de código. (CG12, CG13)
- Conocer las fases necesarias para la construcción de un procesador de lenguaje. (CG12, CG13)
- Conocer los fundamentos de los lenguajes de programación, su evolución y paradigmas. (CG12, CG13)
- Incorporar a un analizador sintáctico la gestión de errores y de la tabla de símbolos. (CG12, CG13)
- Usar herramientas automáticas de generación de analizadores. (CG12, CG13)

**Evaluación detallada:**

El 75% de la nota se obtendrá en el examen final. El 25% restante se obtendrá por la realización individual de hojas de ejercicios y prácticas de laboratorio, y su supervisión en las clases de problemas y sesiones de laboratorio. Estos ejercicios y prácticas solo podrán realizarse durante el periodo en el que se imparte la asignatura, y por tanto, no habrá ninguna oportunidad adicional para recuperar este 25% para la convocatoria extraordinaria.

En el caso de reducirse la presencialidad de la docencia en mas de un 50% de las horas inicialmente previstas, los porcentajes se mantendrían: 75% examen final y 25% entrega de ejercicios y prácticas.

**Actividades docentes:**

Reparto de créditos:	Otras actividades:
Teoría: 3,00	Clases teóricas magistrales.
Problemas: 2,00	Estudio
Laboratorios: 1,00	Realización individual de ejercicios
	Tutorías
	Clases de problemas.
	Prácticas de laboratorio.
	Realización de exámenes.

**Bibliografía:**

M.L. Scott. Programming Language Pragmatics. Third Edition. Elsevier 2009.  
R. Wilhelm; D. Maurer. Compiler Design. Addison-Wesley, 1995  
Alfred V. Aho, Monica S. Lam, Ravi Sethi y Jeffrey D. Ullman. Compiladores. Principios, Técnicas y Herramientas. Segunda edición. Ed. Addison Wesley, 2008.  
Kenneth C. Louden. Construcción de Compiladores. Ed. Thomson, 2004.  
M. Alfonseca, M. de la Cruz, A. Ortega y E. Pulido. Compiladores e Intérpretes: Teoría y Práctica. Ed. Pearson Prentice Hall, 2006.  
G. Sánchez y J.A. Valverde. Compiladores e intérpretes: un enfoque pragmático. Ed. Díaz de Santos, 1989.  
F.J. Sanchís y C. Galán. Compiladores: teoría y construcción. Ed. Paraninfo, 1986.  
A. Garrido, J.M. Iñesta, F. Moreno y J.A. Pérez. Diseño de Compiladores. Publicaciones Universidad de Alicante, 2002.  
D. Grune, H. Bal, C. Jacobs y K. Langendoen. Modern Compiler Design. John Wiley & Sons. 2000.  
T. Pittman, J. Peters. The Art of Compiler Design: Theory and Practice. Ed. Prentice-Hall, 1992.  
J.P. Bennett. Introduction to Compiling Techniques: A First Course Using ANSI C, LEX & YACC. Ed. McGraw-Hill, 1990.  
J.R. Levine, T. Mason y D. Brown. Lex & yacc. Ed. O'Reilly, 1995.  
T. Parr. The Definitive ANTLR 4 Reference. The Pragmatic Bookshelf, 2012

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> 4º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803206 - Ética, legislación y profesión	<b>Abrev:</b> ELP	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Ethics, Legislation and Profession	<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Ética, legislación y profesión		<b>6 ECTS</b>
<b>Otras asignaturas en la misma materia:</b> No hay		
<b>Módulo:</b> Materias comunes a la rama de la informática		
<b>Departamento:</b> Interdepartamental ACYA / ISIA	<b>Coordinador:</b> Sánchez-Elez Martín, Marcos	

**Descripción de contenidos mínimos:**

- Introducción a la ética.
- Privacidad.
- Libertad de expresión.
- Propiedad intelectual.
- Delitos informáticos.
- Seguridad en el trabajo.
- Uso responsable de la tecnología.
- Control de la tecnología.
- Fiabilidad y responsabilidad.
- Códigos éticos profesionales.

**Programa detallado:**

- Tema 1. Introducción a la ética y la legislación
  - La reflexión y el debate como herramientas de la ética
  - Uso responsable de la tecnología
  - Diseño ético
- Tema 2. Privacidad
  - Vigilancia
  - Redes sociales.
  - Reglamento General de Protección de Datos
- Tema 3. Derechos digitales y desigualdad
  - Libertad de expresión.
  - Anonimato. Censura.
  - Privilegios y desigualdad.
  - Brechas digitales (por edad, género, raza, nivel económico...)
  - Sesgos en el software
- Tema 4. Derechos de autor
  - Modelos de protección de derechos de autor. Ley de Propiedad Intelectual
  - Software libre y licencias libres
  - Cultura libre
- Tema 5. Responsabilidad informática
  - Delitos informáticos, seguridad y control
  - Implicaciones de tecnologías emergentes
- Tema 6. Profesión
  - Qué es ser informático/a.
  - Opciones profesionales y empleadores.
  - Códigos éticos profesionales
- Tema 7 Conferencias
  - Conferencias de profesionales de reconocido prestigio sobre temas relacionados directamente con la asignatura.

**Programa detallado en inglés:**

- Lesson 1: Introduction to ethics and legislation
  - Reflection and debate as tools of ethics
  - Responsible use of technology
  - Ethical design
- Lesson 2. Privacy
  - Surveillance

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Social networks.
- General Data Protection Regulation

Lesson 3. Digital rights and inequality

- Freedom of expression.
- Anonymity. Censorship.
- Privileges and inequality.
- Digital gaps (by age, gender, race, economic level...)
- Software biases

Lesson 4. Copyright

- Copyright protection models. Intellectual Property Law
- Free software and free licenses
- Free culture

Lesson 5. Computer responsibility

- Computer crime, security and control
- Implications of emerging technologies

Lesson 6. Profession

- What it is to be a computer scientist.
- Professional options and employers.
- Professional ethical codes

Lesson 7 Lectures

Lectures by renowned professionals on topics directly related to the subject.

**Competencias de la asignatura:**

**Generales:**

- CG7-Capacidad para diseñar, desarrollar, seleccionar y evaluar, aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a los principios éticos y a la legislación y normativa vigente.
- CG9-Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- CG24-Conocimiento de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

- Analizar los aspectos relacionados en un caso de conflicto ético: alcance, colectivos afectados, posibles escenarios futuros etc. (CG9, CT2)
- Analizar y comprender los procesos de transformación social producidos por las nuevas tecnologías y sus implicaciones éticas. (CG9, CT5)
- Analizar, evaluar y prever las repercusiones sociales de los proyectos informáticos. (CG7, CG9)
- Aplicar sus conocimientos técnicos a un proyecto en equipo que resulte útil para la sociedad (CG24, CT4)
- Aprender a diseñar soluciones tecnológicas adaptables a las necesidades de individuos y grupos sociales. (CG24, CT5)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





- Comprender la importancia de la brecha digital y aprender a utilizar y a diseñar mecanismos tecnológicos que fomenten la igualdad y participación. (CG7, CT5)
- Conocer los principios de la ética informática y la importancia de la disciplina en la sociedad de la información. (CG9)
- Conocer los principios éticos, identificarlos en los códigos éticos y aplicarlos en la concepción y desarrollo de sistemas informáticos. (CG7, CG9)
- Conocer y aplicar los mecanismos tecnológicos disponibles para garantizar los principios éticos. (CG7, CT5)
- Debatir en público con argumentos y datos objetivos, defendiendo su propia posición frente a un tema y también reflexionar sobre las implicaciones éticas de su profesión y el uso de la tecnología habiendo sido capaz de plasmar los puntos de vista de los diferentes actores presentes en la sociedad en el debate (CT1)
- Debatir razonadamente sobre un caso ético y llegar a alguna conclusión sobre lo que es correcto en dicha situación. (CG9, CT2)
- Fomentar el espíritu crítico en el desarrollo de las actividades profesionales. (CG9, CT3)
- Inventar mecanismos tecnológicos que fomenten los principios éticos y garanticen los códigos éticos. (CG9, CT5)
- Participar activamente en la identificación de violaciones de los principios éticos y proporcionar a los usuarios conocimiento y herramientas para paliarlas violaciones de estos principios. (CG7, CT3)
- Presentar en público una breve exposición sobre un tema relacionado con el desarrollo de su profesión, y las implicaciones éticas y sociales de las tecnologías TICs. (CG9, CT1)

**Evaluación detallada:**

La asignatura se califica mediante un examen (70%) tanto en la convocatoria ordinaria como en la extraordinaria, y otras actividades de evaluación continua (30%), que no se podrán recuperar en la convocatoria extraordinaria.

El examen teórico consta de: una parte teórica (60%), que consiste en una evaluación sobre los contenidos desarrollados a lo largo de la asignatura, y una parte de conferencias (10%).

El 30% de otras actividades reflejará la evaluación de la participación y el trabajo del/la estudiante en las diferentes actividades diarias de clase, tales como debates, foros, ejercicios, entre otras. Para obtener este 30% es necesario asistir al 66% de las clases, en caso contrario se calificará con 0 esta parte.

**Actividades docentes:**

Reparto de créditos:

Teoría: 6,00

Problemas: 0,00

Laboratorios: 0,00

Otras actividades:

Clases teóricas participativas, debates, presentaciones de alumnos/as, trabajos individuales y grupales, charlas de conferenciantes, ejercicios prácticos grupales e individuales.

**Bibliografía:**

- o "Software libre para una sociedad libre", Richard Stallman, Traficantes de Sueños, 2002, 978-84-933555-1-7
- o "Copyleft. Manual de uso", VVAA, Traficantes de Sueños, 2006. 978-84-96453-14-6
- o "No Place to Hide: Edward Snowden, the NSA, and the U.S. Surveillance State", Glenn Greenwald, Metropolitan Books, 2014. 978-1627790734
- o "La ética del hacker y el espíritu de la era de la información", Pekka Himanen, Destino, 2004, 978-8423336371
- o "A Gift of Fire: Social, Legal, and Ethical Issues for Computers and the Internet", Sara Baase, 3rd Edition, Prentice Hall, 2008. 978-0136008484
- o "The Wealth of Networks", Yochai Benkler, Yale University Press, 2006, 978-0300125771
- o "Ethical and Social Issues in the Information Age", Joseph Migga Kizza, Text in Computer Science, Springer 2015, 978-1447149903
- o "Female Innovators at Work: women on top tech", Danielle Newnham, Ed. Apress, 2016
- o "Ciberguerra", Yolanda Quintana, Ed. Los Libros de la Catarata, 2016
- o "El kit de la lucha en internet", Margarita Padilla, Ed. Traficantes de Sueños, 2012
- o "Género, Ciencia y Tecnologías de la Información", Cecilia Castaño y Juliette Webster, Ed. Aresta, 2014
- o "Hackstory.es: la historia nunca contada del underground hacker de la Península Ibérica", Mercé Molist ISBN 978-84-616-8055-9 Edición digital con licencia CC NC ND

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 4º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803219 - Sistemas empotrados		<b>Abrev:</b> SE	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Embedded Systems		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Tecnología y Arquitectura de Computadores			<b>24 ECTS</b>
<b>Otras asignaturas en la misma materia:</b>			
Arquitectura de Computadores			6 ECTS
Electrónica			6 ECTS
Tecnología de computadores			6 ECTS
<b>Módulo:</b> Ingeniería de computadores			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Mecha López, Hortensia	

**Descripción de contenidos mínimos:**

Introducción a los sistemas empotrados y aplicaciones en tiempo real.  
Microprocesadores, microcontroladores y procesadores de propósito específico.  
Subsistema de memoria en sistemas empotrados.  
Sistemas-en-chip.  
Diseño automático y codiseño HW/SW sobre plataformas reconfigurables.  
Optimización de prestaciones, consumo de potencia y fiabilidad en sistemas empotrados

**Programa detallado:**

1. Introducción a los sistemas empotrados y aplicaciones en tiempo real. (2 horas teóricas)
  - 1.1 ¿Qué es un sistema empotrado?
  - 1.2 Características y ámbitos de aplicación de los sistemas empotrados
  - 1.3 Flujo de diseño
  - 1.4 Componentes de un sistema empotrado
2. Microprocesadores, microcontroladores y procesadores de señal digital (4 horas teóricas +1 hora prácticas)
  - 2.1 Modelos de Arquitectura ISA
  - 2.2 Modelos ISA específicos para aplicaciones
  - 2.3 Diseño de un procesador
  - 2.4 Rendimiento del procesador
3. Subsistema de memoria en sistemas empotrados. (4 horas teóricas +2 horas problemas)
  - 3.1 Espacio de memoria
  - 3.2 Memoria Cache
  - 3.3 Memoria principal
  - 3.4 Memoria auxiliar o de almacenamiento
  - 3.5 Acceso directo a memoria
  - 3.6 Rendimiento y memoria
4. Buses industriales. (2 horas teóricas +2 horas prácticas)
  - 4.1 Protocolos
  - 4.2 Comunicación serie
  - 4.3 Comunicación paralela
  - 4.4 Rendimiento del bus
  - 4.5 Ejemplos: I2C, USB, SPI, PCI, CAN Bus, usb, JTAG,
5. Periféricos: sensores y actuadores. (3 horas teóricas +1 hora prácticas)
  - 5.1 Interfaz digital
  - 5.2 Interfaz analógico
    - 5.2.1 Sensores: temperatura, humedad, giróscopo, aceleración, luz, humo, proximidad, etc
    - 5.2.2 Actuadores: leds, displays, motor continua, motor paso a paso, servomotor. lcd
    - 5.2.3 Conversores analógico/digitales y digitales/analógicos. PWM
6. Integración, coste y prestaciones. (7 horas teóricas)
  - 6.1 Ejecución en tiempo real: ligaduras
  - 6.2 Herramientas de análisis de tiempos de ejecución
  - 6.3 Rendimiento en Sistemas Empotrados Distribuidos
  - 6.4 Diseño de bajo consumo
  - 6.5 Diseño de sistemas fiables
7. Casos prácticos. (8 horas teóricas )
  - 7.1 Control teclado PS2

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- 7.2 Control teclado matricial
- 7.3 Control matriz de puntos
- 7.4 Control VGA
- 7.5 Control lcd
- 7.6 Control zumbador y altavoz
- 7.7 Control LED RGB a través de un PWM
- 7.8 Control emisor/receptor de infrarrojos
- 7.9 Conversores analógico/digitales
  - 7.9.1 Control sensor temperatura
  - 7.9.2 Control fotorresistencias
- 7.10 Control motor paso a paso

Prácticas: 6 prácticas con el entorno EDK Xilinx y placas de Spartan 3  
(4 horas de prácticas en aula +14 horas en laboratorio = 18 horas)

**Programa detallado en inglés:**

1. Embedded Systems: fields of application and design flow
  - 1.1 What is an embedded system?
  - 1.2 Embedded system features and fields of applications.
  - 1.3 Design flow
  - 1.4 Embedded system Components
2. Microprocessors, microcontrollers and digital signal processors
  - 2.1 ISA Architecture models
  - 2.2 ISA architecture for specific applications
  - 2.3 Processor Design
  - 2.4 Processor performance
3. Memory subsystem in embedded systems
  - 3.1 Memory space
  - 3.2 Cache
  - 3.3 Main memory
  - 3.4 Secondary memory
  - 3.5 DMA
  - 3.6 Memory performance
4. Industrial buses.
  - 4.1 Protocols
  - 4.2 Serial communication
  - 4.3 Parallel Communication
  - 4.4 Bus performance
  - 4.5 Examples: I2C, USB, SPI, PCI, USB, JTAG, NFC y RFID
5. Peripherals: sensors and actuators.
  - 5.1 Digital Interface
  - 5.2 Analog Interface
    - 5.2.1 Sensors: temperature, humidity, gyro, acceleration, light, hume, proximity
    - 5.2.2 Actuator: leds, displays, motors, lcd
    - 5.2.3 Analog/digital and Digital/Analog converters. PWM
6. Integration, cost and performance.
  - 6.1 Real time execution
  - 6.2 Execution time analysis tools
  - 6.3 Distributed Embedded system performance.
  - 6.4 Low power dissipation design
  - 6.5 Reliable system design
7. Case studies Laboratories: Six practical labs using EDK Xilinx tool and Spartan 3 based platforms
  - 7.1 PS2 keyboard control
  - 7.2 Keypad control
  - 7.3 Banner control
  - 7.4 VGA control
  - 7.5 LCD control

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

- 7.6 Buzzer and speaker control
- 7.7 RGB LED control by PWM
- 7.8 Infrared Sender/receiver control
- 7.9 Analog/digital converters  
LDR control
- 7.10 Motor step control

Laboratory: 6 lab work with EDK Xilinx and Spartan 3

**Competencias de la asignatura:**

**Generales:**

- CG14-Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

**Específicas:**

- CE\_GIC1-Capacidad de diseñar y construir sistemas digitales, incluyendo computadores, sistemas basados en microprocesador y sistemas de comunicaciones.
- CE\_GIC2-Capacidad de desarrollar procesadores específicos y sistemas empotrados, así como desarrollar y optimizar el software de dichos sistemas.
- CE\_GIC5-Capacidad de analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empotradas y de tiempo real.
- CE\_GIC7-Capacidad para analizar, evaluar, seleccionar y configurar plataformas hardware para el desarrollo y ejecución de aplicaciones y servicios informáticos.

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

- Analizar distintas plataformas hardware y seleccionar la mejor para implementar aplicaciones en tiempo real (CE\_GIC5)
- Analizar distintas plataformas hardware y seleccionar la mejor para implementar un sistema empotrado basándose en una especificación dada (CE\_GIC5)
- Analizar y diseñar la estructura de un sistema distribuido de complejidad baja. (CG16)
- Analizar y evaluar distintas soluciones software en función de las necesidades de consumo (CE\_GIC5)
- Analizar y evaluar distintas soluciones software en función de las necesidades de seguridad (CE\_GIC5)
- Aplicar los conocimientos de estructura de computadores y de programación para diseñar y evaluar distintas opciones de diseño de una especificación dada. (CG14)
- Comprender e interpretar las especificaciones y los parámetros de diseño de un sistema empotrado. (CG14)
- Comprender los factores reales que afectan al diseño de estos sistemas y su influencia en el estilo de diseño y el resultado final. (CG14)
- Configurar una plataforma hardware basada en hardware reconfigurable para implementar un sistema empotrado con distintos sensores y actuadores (CE\_GIC7)
- Configurar una plataforma hardware basada en hardware reconfigurable para implementar un sistema empotrado que ejecute aplicaciones en tiempo real (CE\_GIC7)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Conocer y aplicar los métodos básicos para mejorar la temporización y sincronización de un sistema empotrado. (CG14)
- Conocer y argumentar las ventajas e inconvenientes de distintas opciones de sistemas empotrados. (CG14)
- Construir y evaluar, mediante las medidas oportunas, diferentes sistemas empotrados de complejidad media. (CE\_GIC1)
- Desarrollar el software de un sistema empotrado (CE\_GIC2)
- Diferenciar las distintas perspectivas y necesidades de comunicación según los participantes en un proyecto de desarrollo de un sistema empotrado. (CT1)
- Estudiar distintas alternativas de diseño de un sistema empotrado (CE\_GIC1)
- Experimentar el desarrollo de sistemas empotrados de complejidad media-alta. (CT3)
- Planear distintas opciones de diseño y seleccionar aquellas que mejor satisfagan las especificaciones. (CG14)
- Usar los medios audiovisuales para expresar la información de un proyecto de sistema empotrado de forma apropiada para su comprensión por el cliente y el entorno de desarrollo. (CT1)
- Utilizar una placa de expansión con distintos sensores/actuadores para desarrollar aplicaciones en tiempo real (CE\_GIC2)
- Valorar el impacto medioambiental derivado de la puesta en marcha de un sistema empotrado (CT5)
- Valorar el impacto y cambio social que puede involucrar la puesta en marcha de un sistema empotrado. (CT5)
- Valorar y seleccionar la arquitectura de memoria adecuada para la aplicación objetivo. (CG14)
- Analizar distintas plataformas hardware y seleccionar la mejor para implementar un sistema empotrado en función de la aplicación que vaya a ejecutar (CE\_GIC7)
- Analizar el comportamiento temporal de los sistemas empotrados y plantear hipótesis sobre las posibles causas de su comportamiento erróneo. (CG14)
- Analizar y evaluar distintas plataformas hardware en función de las necesidades de consumo (CE\_GIC5)
- Analizar y evaluar distintas plataformas hardware en función de las necesidades de seguridad (CE\_GIC5)
- Decidir la estructura del sistema empotrado adecuada para implementar la funcionalidad especificada. (CG14)
- Desarrollar un sistema empotrado basándose en una especificación dada (CE\_GIC2)
- Diseñar e implementar sistemas empotrados basándose en una especificación dada (CE\_GIC1)
- Diseñar sistemas empotrados que satisfagan la especificación. (CT2)
- Estimar las características físicas de la implementación de un sistema empotrado. (CG14)
- Evaluar una especificación de un sistema empotrado y justificar modificaciones basadas en la tecnología objetivo. (CT2)
- Modificar las técnicas y recursos disponibles para adaptarlos a las necesidades específicas del diseño de sistemas empotrados. (CT3)
- Optimizar el software de un sistema empotrado en función de las necesidades de consumo. (CE\_GIC2)
- Optimizar el software de un sistema empotrado en función de las necesidades de seguridad (CE\_GIC2)

**Evaluación detallada:**

Asistencia a clase obligatoria (70% mínimo de asistencia), de no cumplirse este mínimo el estudiante no podrá optar a la nota de prácticas.

Examen teórico: 50% en aula (nota mínima 2 sobre 5)

Examen sobre un caso práctico: 10% en aula

Nota de prácticas: 40% (15% prácticas obligatorias, 25% proyecto) podrán recuperarse las prácticas y el proyecto en la convocatoria extraordinaria siempre que el estudiante haya cumplido el mínimo de asistencia.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,30

Problemas: 0,70

Laboratorios: 2,00

Otras actividades:

Los créditos de prácticas de laboratorio se realizarán en 6 prácticas con el entorno EDK Xilinx y placas de Spartan 3

(4 horas de prácticas en aula +14 horas en laboratorio = 18 horas)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

**Bibliografía:**

**Bibliografía básica**

- Embedded hardware., know it all / Jack Ganssle, Tammy Noergaard, Fred Eady, Lewin Edwards, David J. Katz, Amsterdam, Elsevier/Newnes, cop. 2008
- Embedded Systems Handbook. Richard Zurawski. Industrial Information Technology Series
- Embedded Microcomputer Systems . Real Time Interfacing Jonathan W. Valvano International Edition
- Embedded Systems Design, Steve Heath, Ed Newnes, 2005

**Bibliografía complementaria**

- Lluís Terés, Yago Torroja, Serafín Olcoz, Eugenio Villar; VHDL: Lenguaje estándar de diseño electrónico; McGraw Hill, 1998
- Embedded System Design. Peter Marwedel. Ed. Springer
- A Comprehensive Guide for Engineers and Programmers. Tammy Noergaard. Ed Elsevier, 2005
- Computers as components : principles of embedded computing system design / Wayne Wolf. San Francisco, CA : Morgan Kaufmann Publishers, 2001
- Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers. Tammy Noergaard. Ed Elsevier, 2005
- James K. Peckol. Embedded Systems: A Contemporary Design Tool. Wiley. ISBN: 0471721808
- Sam Siewert. Real-Time Embedded Components and Systems. Charles River Media. ISBN: 1584504684

Ficha docente guardada por última vez el 23/06/2021 10:46:00 por el usuario: Coordinador GIC

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> 4º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803220 - Programación de sistemas distribuidos	<b>Abrev:</b> PSD	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Programming of Distributed Systems	<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Sistemas distribuidos		<b>6 ECTS</b>
<b>Otras asignaturas en la misma materia:</b> No hay		
<b>Módulo:</b> Ingeniería de computadores		
<b>Departamento:</b> Sistemas Informáticos y Computación		
<b>Coordinador:</b> Núñez Covarrubias, Alberto		

**Descripción de contenidos mínimos:**

Protocolos de comunicación.  
Mecanismos de comunicación y sincronización.  
Bibliotecas genéricas de paso de mensajes.  
Programación concurrente distribuida.  
Control de procesos.  
Middleware.  
Algoritmos paralelos.  
Programación en la GRID.  
Terminación distribuida.  
Validación y verificación de sistemas distribuidos.

**Programa detallado:**

1. Introducción a los sistemas distribuidos: conceptos básicos.
2. Paradigmas de comunicación: sockets, procedimientos remotos, invocación remota y servicios Web.
3. Arquitecturas de sistemas distribuidos: computación en grid, cluster, P2P y cloud.
4. Modelos, lenguajes y técnicas para la programación de sistemas distribuidos.
5. Sincronización en sistemas distribuidos: relojes físicos y lógicos, estados globales coherentes.
6. Algoritmos distribuidos: algoritmos de exclusión mutua, consenso y terminación, algoritmos tolerantes a fallos.
7. Análisis de rendimiento en sistemas altamente distribuidos utilizando técnicas de modelado y simulación.
8. Verificación y validación de sistemas distribuidos.

**Programa detallado en inglés:**

1. Introduction to distributed systems: basic concepts.
2. Communication paradigms: sockets, remote procedure calls, remote method invocation and Web services.
3. Distributed system architectures: grid, cluster, P2P and cloud computing.
4. Models, languages and techniques for distributed systems programming.
5. Synchronisation in distributed systems: logical and physical clocks, coherent global states.
6. Distributed algorithms: mutual exclusion, consensus and termination algorithms; fault-tolerant algorithms.
7. Performance analysis of highly distributed systems using modelling and simulation techniques.
8. Verification and validation of distributed systems.

**Competencias de la asignatura:****Generales:**

CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

**Específicas:**

CE\_GIC3-Capacidad de analizar y evaluar arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, así como desarrollar y optimizar software para las mismas.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Analizar mediante simulación el rendimiento y la escalabilidad de aplicaciones distribuidas conformes con diferentes arquitecturas. (CT2, CT3)

Definir los conceptos básicos de los sistemas distribuidos. (CE\_GIC3)

Discriminar los paradigmas, las arquitecturas y las tecnologías más comunes en la programación distribuida, y poder compararlos y contrastarlos. (CE\_GIC3)

Distinguir las técnicas básicas de verificación y validación de las aplicaciones distribuidas. (CT2)

Emplear algunos de los modelos, lenguajes y técnicas más conocidos de la programación distribuida. (CT2)

Identificar las propiedades más importantes de los sistemas y las aplicaciones distribuidos. (CT2)

Manejar algunos de los algoritmos distribuidos más conocidos e identificar sus propiedades. (CT2)

Solucionar problemas informáticos mediante la realización de una aplicación distribuida conforme con algunos de los paradigmas y arquitecturas, construida con algunas de las tecnologías, y empleando algunos de los modelos, lenguajes y técnicas, estudiados. (CT3)

**Evaluación detallada:**

Examen final: 70%

Realización de prácticas obligatorias: 30%.

Para poder aprobar la asignatura se requerirá, al menos, una calificación de 4 sobre 10 en el examen final y será necesario tener todas las prácticas aprobadas: entregadas en plazo y evaluadas al menos con un 5 sobre 10.

La evaluación de cada práctica se realizará mediante una defensa individual. La fecha de la defensa se publicará con antelación suficiente.

La nota final, para ambas convocatorias, se compone del 70% de la calificación obtenida en el examen y del 30% de la calificación obtenida en las prácticas. Para aprobar la asignatura es obligatorio obtener, al menos, un 5 sobre 10 en la nota final.

En la convocatoria extraordinaria, además de realizar un examen final, se abrirá un plazo para la entrega y defensa de las prácticas. Las prácticas serán las mismas para ambas convocatorias.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 1,00

Laboratorios: 2,00

Otras actividades:

Actividades presenciales: Clases teóricas donde se presentarán los conocimientos que los alumnos deben adquirir. Además, se resolverán ejercicios por parte del profesor explicando los conceptos impartidos en las clases magistrales de teoría.

Actividades dirigidas: Realización de ejercicios prácticos tutorizados en laboratorio por parte del alumno.

**Bibliografía:**

Sistemas distribuidos. Conceptos y diseño. G. Coulouris, J. Dollimore, T. Kindberg. Addison-Wesley, 2005, 4ª edición

Computación distribuida. Fundamentos y Aplicaciones. M. L. Liu. Pearson Educación, 2004

Pacheco, Peter S. Parallel programming with MPI / Peter S. Pacheco. cop. 1997

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803224 - Sistemas inteligentes		<b>Abrev:</b> SI	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Smart Systems		<b>Carácter:</b> Obligatoria	
<b>Materia:</b> Desarrollo de Software Avanzado		12 ECTS	
<b>Otras asignaturas en la misma materia:</b> Sistemas web		6 ECTS	
<b>Módulo:</b> Complementario			
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> Sagredo Olivenza, Ismael	

**Descripción de contenidos mínimos:**

- Conceptos básicos de inteligencia artificial.
- Agentes software y sistemas multiagente.
- Estándares para el desarrollo de sistemas multiagente.
- Lenguajes de comunicación entre agentes.
- Programación de sistemas multiagente - plataformas de desarrollo.
- Metodologías de desarrollo de sistemas multiagente.

**Programa detallado:**

- Inteligencia Artificial. Evolución histórica. Técnicas básicas. Aplicaciones. Retos y características de los problemas de la IA. Tipos de IA según el conocimiento y métodos de razonamiento.
- Agentes Software para el diseño de Sistemas Inteligentes. Teorías, modelos y arquitecturas. IA cognitiva basada en agentes. Sistemas multiagente y entornos de simulación. Aplicaciones. Estándares para el desarrollo de sistemas multiagente. Lenguajes de comunicación entre agentes. Metodologías de desarrollo de sistemas multiagente.
- Paradigmas de resolución de problemas
  - Resolución de problemas con búsqueda heurística
  - Búsqueda local y algoritmos genéticos
  - Razonamiento con reglas y razonamiento aproximado
- Conceptos básicos de aprendizaje automático
- Conceptos básicos de procesamiento del lenguaje natural

**Programa detallado en inglés:**

- Artificial Intelligence. Historical evolution. Basic techniques. Applications. Challenges and characteristics of AI problems. Types of AI according to knowledge and reasoning methods.
- Software Agents for the design of Intelligent Systems. Theories, models and architectures. Agent-based cognitive AI. Multi-agent systems and simulation environments. Applications. Standards for the development of multi-agent systems. Communication languages between agents. Multi-agent systems development methodologies.
- Problem solving paradigms.
  - Heuristic search
  - Local search and genetic algorithms
  - Rule based reasoning
  - Approximate reasoning
- Machine learning (an introductory view)
- Natural language processing basics.

**Competencias de la asignatura:****Generales:**

- CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
- CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Conocer, aplicar y evaluar algoritmos de búsquedas informadas, no informadas, local y con adversario. (CG23, CT2)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Diseñar e implementar un sistema basado en agentes sobre herramientas de uso generalizado. (CG23, CT2, CT3)
- Elegir representaciones del problema más adecuadas para aplicar las técnicas de inteligencia artificial para resolver problemas (CG18)
- Entender las componentes básicas de un Sistema Inteligente a través de la descripción de la resolución de un problema mediante técnicas de resolución de problemas y uso eficiente de conocimiento (CG23)
- Estudiar el problema de la interoperabilidad entre sistemas inteligentes y cómo se resuelve usando lenguajes de comunicación de agentes (CG23)
- Evaluar qué tipos de inteligencia son necesarias y cómo lograr ese tipo de inteligencia mediante búsquedas, satisfacción de restricciones, algoritmos genéticos, redes neuronales y sistemas de reglas. (CG23)
- Integración de técnicas de Inteligencia Artificial en el diseño de sistemas inteligentes mediante el concepto de agente inteligente. (CG23)
- Integrar los conocimientos adquiridos en el contexto de un desarrollo industrial usando metodologías de diseño orientado a agentes (CG23, CT2, CT3)
- Representar conocimiento usando métodos de inteligencia artificial (CG18, CG23, CT3)
- Tratar con la incertidumbre en la codificación de la información adquirida por el sistema (CG18, CT3)

**Evaluación detallada:**

La nota de la asignatura se calcula: Examen 60%, Otras actividades 40% (prácticas y ejercicios)

Es necesario aprobar el examen final en convocatoria ordinaria o extraordinaria.

Las prácticas y ejercicios se realizan en las sesiones prácticas reservadas y se entregan en la fecha indicada durante el curso. No se permite entregar las actividades adicionales en la convocatoria extraordinaria.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Clases teóricas, clases de problemas y laboratorios

**Bibliografía:**

Russell, S., Norvig, P., Inteligencia Artificial: Un enfoque moderno. Prentice Hall, 2004, segunda edición.

Russell, S., Norvig, P., Artificial Intelligence: A Modern Approach. Prentice Hall, 2010, third edition.

Mas, A., Agentes Software y Sistemas Multiagente: conceptos, arquitecturas y aplicaciones. Prentice Hall, 2005.

Wooldridge, M., An Introduction to MultiAgent Systems. John Wiley & Sons Ltd, 2006.

Brian Henderson-Sellers, Paolo Giorgini: Agent-Oriented Methodologies, Idea Group Inc (IGI), 2005.

Federico Bergenti, Marie-Pierre Gleizes, Franco Zambonelli: Methodologies and Software Engineering for Agent Systems, The Agent-Oriented Software Engineering Handbook. Springer, 2004.

Michael Huhns, Munindar P. Singh: Readings in Agents, Morgan Kaufmann, 1998.

Rafael H. Bordini, Mehdi Dastani, Jürgen Dix, Amal El Fallah Seghrouchni: Multi-Agent Programming: Languages, Platforms and Applications, Springer-Verlag, 2005.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803235 - Arquitectura Interna de Linux y Android <b>Asignatura en Inglés:</b> Linux and Android Internals		<b>Abrev:</b> LIN <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de arquitectura y tecnología de computadores		<b>30 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Cloud y Big Data Diseño automático de sistemas Programación de GPUs y aceleradores Robótica		6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Sáez Alcaide, Juan Carlos	

**Descripción de contenidos mínimos:**

Para cursar esta asignatura es preciso haber cursado (y preferiblemente aprobado) la asignatura "Sistemas Operativos".

1. Introducción: historia, Android vs. GNU/Linux, uso avanzado de módulos del kernel
2. Llamadas al sistema y drivers: compilación del kernel, implementación de llamadas al sistema y drivers
3. Gestión de procesos en Linux: Estructuras de datos para gestión de procesos, sincronización en el kernel
4. Interrupciones y trabajos diferidos: softirqs, tasklets y workqueues. Temporizadores del kernel
5. Arquitectura Interna de Android: componentes de Android, procesos en Android, kernel y native userspace, compilación

**Programa detallado:**

## Módulo 1. Introducción

- 1.1. Evolución histórica
- 1.2. Uso avanzado de módulos del kernel. Sistema de ficheros /proc. Listas enlazadas del kernel.

## Módulo 2. Llamadas al sistema y drivers

- 2.1. Compilación del kernel
- 2.2. Implementación de llamadas al sistema
- 2.3. Drivers en Linux. Linux Device Model (LDM)
- 2.4. Gestión de dispositivos USB. Anatomía de un driver USB

## Módulo 3. Gestión de procesos en Linux

- 3.1. Fundamentos de gestión de procesos en Linux. Kernel Threads
- 3.2. Mecanismos de sincronización del kernel

## Módulo 4. Interrupciones y trabajos diferidos

- 4.1. Gestión de interrupciones
- 4.2. Softirqs, tasklets y workqueues
- 4.3. Temporizadores del kernel

## Módulo 5. Arquitectura Interna de Android

- 5.1. Arquitectura y componentes de Android
- 5.2. Android vs. GNU/Linux
- 5.3. Procesos en Android
- 5.4. Kernel y Native Userspace
- 5.5. Sistema de compilación

## Prácticas de laboratorio

- Uso avanzado de módulos del kernel
- Implementación de llamadas al sistema
- Fundamentos de drivers en Linux. Desarrollo de driver para un dispositivo USB
- Procesos y sincronización en el kernel
- Gestión de interrupciones y trabajos diferidos

**Programa detallado en inglés:**

## Unit 1. Introduction

- 1.1. Background and overview
- 1.2. Advanced kernel module programming. The /proc file system. Linked lists in the kernel.

## Unit 2. System calls and device drivers

Fecha: \_\_\_\_ de \_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- 2.1. Building the kernel
- 2.2. System call implementation
- 2.3. Linux device drivers. Linux Device Model (LDM)
- 2.4. USB device management. Anatomy of a USB driver

Unit 3. Process management

- 3.1. Fundamentals of Linux process management. Kernel Threads
- 3.2. Kernel synchronization methods

Unit 4. Interrupts and deferring work

- 4.1. Interrupt handling
- 4.2. Softirqs, tasklets and workqueues
- 4.3. Kernel timers

Unit 5. Android Internals

- 5.1. Android architecture and components
- 5.2. Android vs. GNU/Linux
- 5.3. Processes in Android
- 5.4. Kernel and Native Userspace
- 5.5. Android build system

Lab Assignments

- Advanced kernel module programming
- Implementation of system calls
- Fundamentals of Linux device drivers. Creating a USB device driver
- Process management and kernel synchronization
- Interrupt handling and deferring work

**Competencias de la asignatura:**

**Generales:**

- CG10-Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.
- CG14-Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- CG15-Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios.

**Específicas:**

- CE\_GIC4-Capacidad de diseñar e implementar software de sistema y de comunicaciones.
- CE\_GIC6-Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos.
- CE\_GIC7-Capacidad para analizar, evaluar, seleccionar y configurar plataformas hardware para el desarrollo y ejecución de aplicaciones y servicios informáticos.

**Básicas y Transversales:**

- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Conocer las principales ventajas e inconvenientes de distintas estrategias de diseño del sistema operativo. (CG15)
- Utilizar herramientas administrativas para gestión de drivers y módulos del kernel. (CG10)
- Implementar llamadas al sistema sencillas en el kernel Linux. (CG15, CE\_GIC4, CE\_GIC7)
- Decidir qué mecanismo de sincronización del kernel del sistema operativo resulta más adecuado en distintos contextos. (CG15, CT2, CT3, CE\_GIC4)
- Conocer la estructura y la administración de los paquetes de una distribución de Linux. (CG10)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Comprender el proceso de configuración, compilación e instalación del kernel Linux. (CG10)
- Utilizar distintos mecanismos para diferir trabajo en el sistema operativo. (CG15, CT2, CT3, CE\_GIC4)
- Comprender la relación existente entre los distintos componentes del sistema operativo Android. (CG15, CT3)
- Construir ejecutables de programas en C/C++ y módulos del kernel para Android. (CG10, CG15, CE\_GIC4)

**Evaluación detallada:**

Dos modalidades a elegir: evaluación continua o práctica especial

Modalidad de evaluación continua (aplicable solo a la convocatoria ordinaria):

Asistencia al laboratorio y realización de prácticas propuestas durante el semestre = 60%

Trabajos y presentaciones = 15%

Realización de práctica final = 25%

Modalidad de práctica especial (aplicable a las dos convocatorias de la asignatura):

Práctica especial de laboratorio (se realizarán diferentes ejercicios teórico-prácticos el día asignado por el profesor) = 100%

**Actividades docentes:**

Reparto de créditos:

Teoría: 2,00

Problemas: 0,00

Laboratorios: 4,00

Otras actividades:

Enseñanza presencial teórica. Realización de prácticas de laboratorio.

**Bibliografía:**

Bibliografía Básica

- Kaiwan N. Billimoria. Linux Kernel Programming. Packt Publishing. 1st edition. 2021

- Kaiwan N. Billimoria. Linux Kernel Programming. Part 2 - Char Device Drivers and Kernel Synchronization. Packt Publishing. 1st edition. 2021

- Robert Love. Linux Kernel Development. 3rd Edition. Addison Wesley. 2010

- Jonathan Levin. Android Internals - A Confectioner's Cookbook - Volume I - The Power User's View. Technogeeks.com. 1st edition. 2015

- Karim Yaghmour. Embedded Android. 1st Edition. O'Reilly. 2013

- Wolfgang Mauerer. Professional Linux Kernel Architecture. Wiley Publishing. 2008

Bibliografía Complementaria

- Jonathan Corbet. Linux Device Drivers. 3rd Edition. O'Reilly. Febrero 2005

- Daniel P. Bovet. Understanding the Linux Kernel. 3rd Edition. O'Reilly. 2005

- Marko Gargenta. Learning Android. 2nd Edition. O'Reilly. 2014

- Earlece Fernandes. Instant Android Systems Development How To. Packt Publishing. 2013

- A.S. Tanenbaum y A.S. Woodhull. Operating Systems Design and Implementation. 3rd Edition. Prentice Hall. 2006

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803237 - Programación de GPUs y aceleradores <b>Asignatura en Inglés:</b> GPUs and accelerators programming	<b>Abrev:</b> GPU <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de arquitectura y tecnología de computadores	<b>36 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Arquitectura Interna de Linux y Android Arquitecturas y Programación de Computadores Cuánticos Cloud y Big Data Diseño automático de sistemas Robótica	6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> García Sánchez, Carlos

**Descripción de contenidos mínimos:**

Arquitecturas Especializadas  
Procesadores Gráficos  
Lenguajes de programación de GPUs y aceleradores

**Programa detallado:**

- 1.- Introducción
  - 2.- Procesadores gráficos
    - 2.1.- Historia
    - 2.2.- Programación de GPUs con CUDA
    - 2.3.- Programación de GPUs y Aceleradores con SYCL/DPC++
    - 2.4.- Librería optimizadas
    - 2.5.- Programación basada en directivas
- Práctica:
- Programación GPU básica en CUDA y SYCL
  - Programación GPU mediante directivas: OpenACC

**Programa detallado en inglés:**

- 1.- Introduction
  - 2.- Graphic Processor Units (GPUs)
    - 2.1.- History
    - 2.2.- GPUs programming with CUDA
    - 2.3.- GPUs programming with SYCL/DPC++ paradigm
    - 2.4.- Optimized libraries on GPUs
    - 2.5.- GPUs programming with directives
- Laboratory:
- GPUs programming with CUDA and SYCL
  - GPUs programming with OpenACC directive model

**Competencias de la asignatura:****Generales:**

- CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.
- CG2-Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.
- CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.
- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CG14-Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

**Específicas:**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





# UNIVERSIDAD COMPLUTENSE DE MADRID

## FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803238 - Diseño automático de sistemas <b>Asignatura en Inglés:</b> Digital Systems Design	<b>Abrev:</b> DAS <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de arquitectura y tecnología de computadores	<b>30 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Arquitectura Interna de Linux y Android Cloud y Big Data Programación de GPUs y aceleradores Robótica	6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Mendías Cuadros, José Manuel

### Descripción de contenidos mínimos:

Diseño Automático de Hardware  
Síntesis de Sistemas  
Especificación y análisis a nivel RT

### Programa detallado:

Introducción al diseño automático de sistemas digitales.  
Descripción del puesto de trabajo: la FPGA Xilinx Artix-7, la placa de prototipado Digilent Basys 3 y el entorno Xilinx Vivado ML.  
Especificación a nivel lógico-RT de sistemas digitales usando VHDL.  
Técnicas de diseño de nivel lógico-RT.  
Análisis de sistemas digitales a nivel lógico-RT.  
Síntesis y depuración sobre FPGAs.  
Prácticas de laboratorio.

### Programa detallado en inglés:

Introduction to design automation of digital systems.  
Working framework description: the Xilinx Artix-7 FPGA, the Digilent Basys 3Tprototyping board and the Xilinx Vivado ML Design Suite.  
Logic-RT level specification of digital systems using VHDL.  
Logic-RT level design techniques.  
Logic-RT level analysis of digital systems.  
Synthesis and debugging for FPGAs.  
Labs.

### Competencias de la asignatura:

#### Generales:

CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.  
CG14-Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.

#### Específicas:

CE\_GIC1-Capacidad de diseñar y construir sistemas digitales, incluyendo computadores, sistemas basados en microprocesador y sistemas de comunicaciones.

#### Básicas y Transversales:

CT2-Capacidad de análisis y síntesis en la resolución de problemas.  
CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

#### Resultados de aprendizaje:

Desarrollar sistemas hardware complejos. (CG4, CG14, CT2, CE\_GIC1)  
Diseñar soluciones hardware específicas. (CG4, CG14, CT3, CE\_GIC1)  
Conocer las etapas de proceso diseño automático de un sistema hardware. (CG4, CG14, CT3)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Diseñar utilizando lenguajes de descripción hardware de alto nivel. (CG4, CG14, CT2)

Comprender el funcionamiento del hardware reconfigurable. (CG4, CG14)

**Evaluación detallada:**

La evaluación es continua y la nota final se obtendrá sumando ponderadamente la nota obtenida por las siguientes actividades: prácticas (50% de la nota) y proyecto (50% de la nota).

Las prácticas se realizan durante el periodo lectivo. En caso de no superar la asignatura en convocatoria ordinaria, la nota obtenida en las prácticas se guarda hasta la convocatoria extraordinaria.

La asistencia a clase es obligatoria (70% mínimo de asistencia). El estudiante que no cumpla este requisito será calificado con NP en ambas convocatorias, con independencia de la nota que obtenga en el resto de actividades evaluables de la asignatura.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Enseñanza presencial teórica. Enseñanza presencial de prácticas de laboratorio.

**Bibliografía:**

- H. Kaeslin; Top-Down Digital VLSI Design: From Architectures to Gate-Level Circuits and FPGAs; Morgan Kaufmann, 2014
- H. Kaeslin; Digital Integrated Circuit Design: From VLSI Architectures to CMOS Fabrication; Cambridge University Press, 2008
- P.P. Chu; RTL Hardware Design Using VHDL: Coding for Efficiency, Portability, and Scalability; Cambridge University Press, 2006
- P.P. Chu; FPGA Prototyping by VHDL Examples: Xilinx Spartan-3 Version; Wiley, 2008
- G. De Micheli; Synthesis and Optimization of Digital Circuits; McGraw Hill, 1994
- W.J. Dally, J.W. Poulton; Digital Systems Engineering; Cambridge University Press, 1998
- Jan M. Rabaey; Digital Integrated Circuits; Prentice Hall, 1996

Ficha docente guardada por última vez el 25/06/2023 17:10:00 por el usuario: Coordinador GIC

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803245 - Programación de aplicaciones para dispositivos móviles	<b>Abrev:</b> PAD <b>Carácter:</b> Optativa		<b>6 ECTS</b>
<b>Asignatura en Inglés:</b> Application Programming for Mobile Devices			
<b>Materia:</b> Complementos de software de sistemas		<b>36 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Desarrollo de videojuegos mediante tecnologías web Ingeniería de Comportamientos Inteligentes Introducción a la Tecnología Blockchain y Smart Contracts Programación Competitiva Testing de Software		6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> Valiente Blázquez, Mª Cruz	

<b>Descripción de contenidos mínimos:</b> Desarrollo nativo para plataformas móviles Aplicaciones web móviles Desarrollo multiplataforma
<b>Programa detallado:</b> 1.- Introducción al desarrollo de aplicaciones para dispositivos móviles. 2.- Desarrollo de aplicaciones nativas sobre la plataforma Android. 3.- Desarrollo de aplicaciones con tecnologías Web: HTML5, CSS3 y JavaScript. 4.- Progressive Web Apps (PWA).
<b>Programa detallado en inglés:</b> 1.- Introduction to the development of applications for mobile devices. 2.- Native Android applications development. 3.- Web applications for mobile devices: HTML5, CSS3 and JavaScript. 4.- Progressive Web Apps (PWA).
<b>Competencias de la asignatura:</b> <b>Generales:</b> CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería. CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas. CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
<b>Específicas:</b> No tiene
<b>Básicas y Transversales:</b> CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales. CT2-Capacidad de análisis y síntesis en la resolución de problemas. CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
<b>Resultados de aprendizaje:</b> Conocer las diversas plataformas móviles. (CG4) Desarrollar aplicaciones móviles nativas. (CG4, CG16, CG18, CT1, CT2, CT3) Desarrollar aplicaciones web para móviles. (CG4, CG16, CG18, CT1, CT2, CT3) Desarrollar aplicaciones para móviles desde entornos de desarrollo multiplataforma. (CG4, CG16, CG18, CT1, CT2, CT3)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Evaluación detallada:**

**CONVOCATORIA ORDINARIA**

La asignatura tiene un objetivo eminentemente práctico y de evaluación continua, con un enfoque de aprendizaje basado en proyectos, por lo que la evaluación en la convocatoria ordinaria consta de las siguientes partes:

- Elaboración y exposición pública y oral en clase de un proyecto Android (aplicación nativa) realizado en grupo, que proporcionará el 60% de la nota final. Incluye las actividades llevadas a cabo por cada estudiante a lo largo del cuatrimestre, tales como liderazgo en el equipo, aportaciones al proyecto, entregas del proyecto, defensa pública de las entregas del proyecto, etc.
- Elaboración de prácticas en grupo a lo largo del cuatrimestre, que proporcionará el 30% de la nota final. En la valoración de las prácticas se tendrá en cuenta su ejecución, calidad del código, trabajo en grupo, y defensa pública individual. No se guardarán las prácticas aprobadas, ni se abrirá ningún plazo adicional para entregar dichas prácticas en la convocatoria extraordinaria.
- Realización de actividades con carácter voluntario, así como la participación activa e individual en las clases teóricas y en los foros propuestos a lo largo del cuatrimestre, durante el periodo de clases, que proporcionará el 10% de la nota final.

Tanto en las prácticas, como en el proyecto, será necesario obtener una nota mínima de 4 (sobre 10) para poder realizar la media ponderada y aprobar la asignatura.

La asistencia a las clases prácticas es obligatoria. Se requiere un 80% de asistencia para poder aprobar la asignatura.

**CONVOCATORIA EXTRAORDINARIA**

- 60% examen final.
- 30% realización de nuevos proyectos/prácticas con Android y tecnologías Web de carácter individual.
- 10% de la realización de actividades con carácter voluntario, así como la participación activa e individual en las clases teóricas y en los foros propuestos a lo largo del cuatrimestre, durante el periodo de clases (no recuperable).

Tanto en el examen, como en la parte práctica, será necesario obtener una nota mínima de 4 (sobre 10) para poder realizar la media ponderada y aprobar la asignatura.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Presentaciones y debate con los estudiantes en el aula.

Se valora la participación activa en el proceso de aprendizaje de clase.

**Bibliografía:**

- Reto Meier: Professional Android™ 4 Application Development. John Wiley & Sons, 2012.
- Frank Ableson, Robi Sen, Chris King: Android. Guía para desarrolladores (Segunda edición). Anaya Multimedia, 2011.
- Mark L. Murphy: The Busy Coder's Guide to Android Development. CommonsWare, LLC, 2009.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID  
FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803246 - Desarrollo de videojuegos mediante tecnologías web <b>Asignatura en Inglés:</b> Web technologies for game development		<b>Abrev:</b> DVI <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de software de sistemas		36 ECTS	
<b>Otras asignaturas en la misma materia:</b> Ingeniería de Comportamientos Inteligentes Introducción a la Tecnología Blockchain y Smart Contracts Programación Competitiva Programación de aplicaciones para dispositivos móviles Testing de Software		6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> León Aznar, Carlos	

**Descripción de contenidos mínimos:**

HTML5 y JavaScript  
Arquitecturas de videojuegos  
Tecnologías de videojuegos: sprites, audio, física.

**Programa detallado:**

1. Introducción a la programación juegos en HTML5.
2. Introducción al diseño de videojuegos.
3. JavaScript.
4. Arquitectura de un motor de juegos.
5. Entrada, gestión y uso recursos.
6. Motores de física.
7. Elementos del motor: texto, eventos, tiempo, mapas.
8. Animaciones basadas en sprites.
9. Videojuegos en red con HTML5.
10. Audio.
11. Despliegue de proyectos en la web.

**Programa detallado en inglés:**

1. Introduction to game programming in HTML5.
2. Introduction to game design.
3. JavaScript.
4. Game engine architecture.
5. Input, resource management, and utilization.
6. Physics engines.
7. Engine elements: text, events, time, maps.
8. Sprite-based animations.
9. HTML5 multiplayer games.
10. Audio.
11. Deploying projects on the web.

**Competencias de la asignatura:**

**Generales:**

No tiene

**Específicas:**

No tiene

**Básicas y Transversales:**

No tiene

**Resultados de aprendizaje:**

Escribir programas eficientes, correctos y mantenibles de tamaño medio que hagan un uso adecuado de bibliotecas software y marcos de aplicación desarrollados por otros.

Diseñar programas eficientes, correctos y mantenibles en lenguajes aptos para web.

Evaluar distintas alternativas arquitectónicas de los módulos que componen un motor de videojuegos.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Evaluación detallada:**

Convocatoria ordinaria:

Es obligatoria la asistencia a clase y faltar a más del 30% de las sesiones injustificadamente supondrá el suspenso en la asignatura. La evaluación se hará sobre la realización en grupo de un proyecto consistente en el desarrollo de un videojuego para poder aprobar la asignatura en convocatoria ordinaria. Asimismo, el proyecto deberá defenderse públicamente.

La evaluación se calculará según los siguientes porcentajes:

- Funcionamiento correcto del juego (calidad del código, gameplay, pulido, eficiencia, etc): 60% de la nota.
- Diseño del juego (mecánicas, dinámicas, documento de diseño): 20% de la nota.
- Presentación y comunicación del juego (presentaciones en clase, página web): 10% de la nota.
- Calidad y criterio artístico (gráficos, música, gestión de recursos): 10% de la nota.

Convocatoria extraordinaria:

La calificación de la convocatoria extraordinaria se hará con las mismas condiciones que en la convocatoria ordinaria, entregando y presentando un proyecto. La entrega puede ser una mejora del proyecto presentado en la convocatoria ordinaria

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- Tommy Leung. Infinite Jumper in Phaser 3 with Modern JavaScript. 2020
- Tommy Leung. Infinite Runner in Phaser 3 with TypeScript. 2020
- Jeremy Gibson. Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C#. Addison-Wesley, 2014
- Zachary Kessin. Programming HTML5 Applications. O'Reilly Media, 2011
- Pascal Rettig. Professional HTML5 Mobile Game Development. John Wiley & Sons, 2012
- Jesse Schell. The Art of Game Design: A book of lenses. CRC Press, 2008
- Douglas Crockford. JavaScript: The Good Parts. O'Reilly Media / Yahoo Press, 2008
- Aditya Ravi Shankar. Pro HTML5 Games. APress, 2012

Ficha docente guardada por última vez el 23/06/2023 16:15:00 por el usuario: Coordinador GIS

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803247 - Los escenarios científicos y tecnológicos emergentes y la defensa <b>Asignatura en Inglés:</b> Emergent Scientific and Technological Scenarios and the Defense	<b>Abrev:</b> ECTD <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos científico-matemáticos	18 ECTS	
<b>Otras asignaturas en la misma materia:</b> Computación Cuántica Herramientas informáticas para los juegos de azar	6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Análisis Matemático y Matemática Aplicada		<b>Coordinador:</b> Vázquez Martínez, Luis

**Descripción de contenidos mínimos:**

Entornos de investigación científica y tecnológica  
Problemas computacionales, simulaciones, y nuevos paradigmas de calculo  
Seminarios/conferencias de personal experto externo a la facultad

**Programa detallado:**

La estructura de la asignatura se articula según los siguientes módulos:  
1. Prospectiva Científica y Tecnológica. Sistemas Complejos. Bioinformática.  
2. Centro de Experiencias Hidrodinámicas del Pardo (ICTS-CEHIPAR).  
3. Telemedicina Militar (Hospital Gómez Ulla)  
4. Exploración espacial y misiones a Marte (INTA)  
5. Computación cuántica.  
6. Estrategias de Seguridad y Ciberseguridad (CESEDEN)

**Programa detallado en inglés:**

The course is structured according to the following modules:  
1. Scientific and Technological Perspectives. Complex Systems. Bioinformatics.  
2. El Pardo Hydrodynamic Experiences Center (CEHIPAR).  
3. Military Telemedicine (Gómez Ulla Hospital)  
4. Space exploration and missions to Mars (INTA)  
5. Quantum computing.  
6. Security and Cybersecurity Strategies (CESEDEN)

**Competencias de la asignatura:****Generales:**

CG8-Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.

**Específicas:**

CE\_C1-Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar, y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática.  
CE\_TI1-Capacidad para comprender el entorno de una organización y sus necesidades en el ámbito de las tecnologías de la información y las comunicaciones.

**Básicas y Transversales:**

CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

Analizar los recursos necesarios para la ejecución de un proyecto. (CG8)  
Fomentar el espíritu crítico en el desarrollo de las actividades profesionales. (CE\_C1, CE\_TI1)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Evaluación detallada:**

65% Examen en convocatoria ordinaria/extraordinaria.

35% Asistencia a las clases de teoría, en convocatoria ordinaria/extraordinaria. Esta calificación sólo podrá obtenerse durante el curso y se mantendrá para la convocatoria extraordinaria.

**Actividades docentes:**

Reparto de créditos:

Teoría: 6,00

Problemas: 0,00

Laboratorios: 0,00

Otras actividades:

Para complementar el curso y consolidar la información presentada en las charlas, se organizarán visitas a centros de investigación relacionados con el tema. Entre las posibles opciones, se incluyen el Centro de Experiencias Hidrodinámicas del Pardo (ICTS-CEHIPAR), el Hospital Central de la Defensa "Gómez Ulla", y el Instituto Nacional de Técnica Aeroespacial (INTA).

Debido a las restricciones de visita propios de estos centros, los días y el horario de las visitas se confirmarán una vez iniciado el curso. Al no poder garantizar que las visitas se realicen en los horarios establecidos de clase, la asistencia a la misma es completamente optativa y no se tendrá en cuenta a la hora de evaluar la asistencia a clase de los estudiantes.

**Bibliografía:**

Algunas referencias sobre los temas presentados. Otro material será disponible en el Campus Virtual a lo largo de curso.

1. "Prospectiva Tecnológica: Una introducción a su metodología y su aplicación en distintos países". J.A. Martín Pereda. Estudios COTEC no9 (1997).
2. "Supercomputation in nonlinear and disordered systems: algorithms, applications and architectures". L. Vázquez, F. Tirado and I. Martín (Eds.). World Scientific (1997).
3. "Handbook of medical imaging" Vol. 2. M. Sonka and J.M. Fitzpatrick (Eds.). SPIED PRESS 2000.
4. "Practical UNIX & internet security". Simson Garfinkel and Gene Spafford, O'Reilly & Associates, Inc. 1996.
5. Mars MetNet Mission <http://metnet.fmi.fi>
6. E. Cole. "Network Security Bible", 2nd Edition. Ed. John Wiley&Sons. 2009
7. "Supercomputation in Nonlinear and Disordered Systems: Algorithms, Applications and Architectures ". L. Vázquez, F. Tirado and I.Martín (Eds). World Scientific (1997).
8. "La Gran Aventura de la Exploración en Marte" Luis Vázquez Martínez, María Pilar Velasco Cebrián, José Luis Vázquez Poletti. Editor Guillermo Escolar (2021).
9. J. Tysdall, "Beginning Perl for Bioinformatics". O'Reilly Media, 2001.
10. P. Bourne and H. Weissig, "Structural Bioinformatics". John Wiley&Sons, 2003.
11. M. Nielsen and I. Chuang, "Quantum Information and Quantum Computation". Cambridge University Press, 2010
12. IBM, "Qiskit textbook". <https://qiskit.org/learn/> <https://qiskit.org/textbook/>

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803293 - Programación con restricciones <b>Asignatura en Inglés:</b> Constraint Programming		<b>Abrev:</b> PR <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de programación y lenguajes		18 ECTS	
<b>Otras asignaturas en la misma materia:</b> Especificación, validación y testing Informática gráfica		6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Rubio Gimeno, Albert	

**Descripción de contenidos mínimos:**

Codificación.  
Resolutores.  
Optimizadores.  
Dominios de aplicación.

**Programa detallado:**

- Problemas de satisfacción de restricciones
- Dominios de restricciones: dominios finitos, aritmética real y entera, dominios booleanos
- Resolutores de restricciones: Propagación, consistencia, búsqueda.
- Modelado de problemas de restricciones. Técnicas de optimización.
- Integración de resolutores en lenguajes de propósito general

**Programa detallado en inglés:**

- Constraint Satisfaction Problems
- Constraint domains: finite domains, real and integer arithmetic, boolean domains
- Constraint solvers: propagation, search, consistency
- Modeling constraint satisfaction problems: optimization techniques
- Embedding constraint programming in general purpose languages

**Competencias de la asignatura:****Generales:**

- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CG12-Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.

**Específicas:**

- CE\_C1-Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar, y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática.
- CE\_C2-Capacidad para conocer los fundamentos teóricos de los lenguajes de programación y las técnicas de procesamiento léxico, sintáctico y semántico asociadas, y saber aplicarlas para la creación, diseño y procesamiento de lenguajes.

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

**Resultados de aprendizaje:**

Conocer las características principales de los lenguajes de restricciones. (CE\_C1, CE\_C2)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Aprender a describir problemas usando un lenguaje formal de restricciones. (CT2, CT3, CE\_C1, CE\_C2)

Reconocer problemas combinatorios para los que una codificación usando restricciones es apropiada. (CT2, CT3, CE\_C1)

Conocer las diferentes técnicas para resolver restricciones y distinguir en qué casos es más conveniente usar cada una de ellas. (CT2, CE\_C1)

Organizar la información sobre el problema y detectar si se trata de un problema de satisfacción o de optimización. (CT3, CE\_C1)

Comparar y justificar las soluciones aportadas. Colaborar en la resolución de un problema. (CT1, CT2, CT4, CT5, CE\_C1)

**Evaluación detallada:**

Existen dos modalidades de evaluación. El estudiante deberá optar por una de ellas durante las dos primeras semanas de curso. Si no lo hace, se entenderá por defecto que ha elegido la modalidad B.

**Modalidad A:**

- 60%: realización de dos trabajos prácticos.
- 20%: realización de un examen.
- 20%: resolución de problemas en clase.

**Modalidad B:**

- 40%: realización de un trabajo práctico.
- 60%: realización de un examen.

La convocatoria extraordinaria seguirá la modalidad B.

**Actividades docentes:**

**Reparto de créditos:**

Teoría: 3,00  
Problemas: 0,00  
Laboratorios: 3,00

**Otras actividades:**

Actividades presenciales: 65% de dedicación. 2 horas de clase teórica y 2 de laboratorio por semana.  
Actividades dirigidas: 15%  
Trabajo personal: 20% : estudio, desarrollo de trabajos prácticos

**Bibliografía:**

**Recursos de internet:**

- A MiniZinc Tutorial "<http://www.minizinc.org/downloads/doc-latest/minizinc-tute.pdf>". Version 1.5. Kim Marriott Peter J. Stuckey, Leslie De Koninck, Horst Samulowitz

**Libros:**

- "Programming with Constraints: An Introduction" . Kimbal Marriott and Peter Stuckey. The MIT Press. 1998.
- "Constraint Logic Programming using Eclipse". Krzysztof R. Apt and Mark Wallace. Cambridge University Press; 1 edition (January 15, 2007).

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803297 - Ingeniería web <b>Asignatura en Inglés:</b> Web engineering		<b>Abrev:</b> IW <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de ingeniería de software e inteligencia artificial		<b>18 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Aprendizaje automático y Big Data Programación evolutiva		6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> Freire Morán, Manuel	

**Descripción de contenidos mínimos:**

Tecnologías del lado cliente.  
Tecnologías del lado servidor.  
MVC web.  
Peticiones y mantenimiento de estado en el servidor.  
Seguridad web.

**Programa detallado:**

La asignatura presupone conocimientos básicos de bases de datos y tecnologías web; tiene un fuerte componente práctico, e implica la realización de proyectos en grupo que hagan uso de tecnologías web, incluyendo:

1. Tecnologías básicas: HTTP, HTML5 y CSS
2. Servidores web básicos con Java.
3. MVC en cliente, y uso de JS, JSON y AJAX
4. Persistencia en el servidor: JPA y sesiones
5. MVC en servidor con Spring MVC.
6. Seguridad web, pruebas, y websockets.
7. El ecosistema web Java

**Programa detallado en inglés:**

Students are expected to start with basic knowledge of databases and web technologies. Students will work in teams to develop sites that demonstrate correct use of web technologies, including:

1. Basic technologies: HTTP, HTML5 and CSS
2. Basic web servers with Java
3. Client-side MVC, and use of JS, JSON and AJAX
4. Server-side persistence: JPA and sessions
5. Server-side MVC with Spring MVC
6. Web security, testing, and websockets
7. The broader Java web ecosystem

**Competencias de la asignatura:****Generales:**

- CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.
- CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.
- CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMÁTICA**

**Resultados de aprendizaje:**

- Conocer las tecnologías asociadas a clientes web. (CG4)
- Conocer las tecnologías asociados a servidores web. (CG4)
- Desarrollar aplicaciones web basadas en el patrón MVC. (CG4, CG16, CG18, CT1, CT2, CT3)
- Desarrollar aplicaciones web capaces de mantener estado en el servidor. (CG4, CG16, CG18, CT1, CT2, CT3)
- Desarrollar aplicaciones web con capacidades de autenticación y autorización. (CG4, CG16, CG18, CT1, CT2, CT3)
- Desplegar aplicaciones web. (CG4, CG16, CG18, CT1, CT2, CT3)

**Evaluación detallada:**

- Convocatoria ordinaria:

Proyectos, trabajos, y presentaciones: 40%

Participación en clase: 10%

Examen final: 50%

- Convocatoria extraordinaria:

Proyectos, trabajos, y presentaciones: 40%

Examen final: 60%

- En ambas convocatorias:

Será obligatoria la realización de un proyecto en grupo, con varias entregas intermedias y una presentación.

El examen será individual sobre el proyecto entregado por cada grupo, y su nota se sumará a la del proyecto presentado.

No se puede aprobar el examen sobre el proyecto, si el proyecto está suspenso.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- Spring Mvc (2ª ed.) Ganeshan, Amuthan. Packt Publishing, 2016. Disponible como libro electrónico en la biblioteca.
- Eloquent JavaScript: A Modern Introduction to Programming (3ª ed.). Haverbeke, Marijn. 2018. Disponible en <http://eloquentjavascript.net>.
- Beginning Responsive Web Design with Html5 and Css3. Fielding, Jonathan. Apress, 2014. Disponible como libro electrónico en la biblioteca.

Ficha docente guardada por última vez el 23/06/2022 17:19:00 por el usuario: Coordinador GIS

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803299 - Programación evolutiva <b>Asignatura en Inglés:</b> Evolutionary Computation	<b>Abrev:</b> PEV <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de ingeniería de software e inteligencia artificial	18 ECTS	
<b>Otras asignaturas en la misma materia:</b> Aprendizaje automático y Big Data Ingeniería web	6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> Cervigon Rückauer, Carlos

**Descripción de contenidos mínimos:**

- Algoritmos genéticos
- Algoritmos evolutivos
- Programación genética
- Gramáticas Evolutivas
- Extensiones de algoritmos evolutivos

**Programa detallado:**

- Introducción a la Programación Evolutiva.
- Estructura y componentes básicos del algoritmo genético simple. Operadores básicos.
- Implementación del Algoritmo genético simple.
- Mejoras al esquema básico del algoritmo genético simple (representación, operadores).
- Fundamentos matemáticos.
- Algoritmos evolutivos: otras representaciones.
- Programación genética
- Gramáticas evolutivas.
- Extensiones de los algoritmos evolutivos: ACO, PSO, CRO, Multiobjetivo, Algoritmos evolutivos paralelos.

**Programa detallado en inglés:**

- Introduction to Evolutionary Computation.
- Structure and components of simple genetic algorithm.
- Implementation of simple genetic algorithm.
- Improvements to the basic outline of simple genetic algorithm.
- Mathematical Foundations.
- Evolutionary algorithms: other representations.
- Genetic programming
- Gramatical evolution.
- Extension of evolutionary algorithms: ACO, PSO, CRO, Multiobjective, Parallel evolutionary algorithms.

**Competencias de la asignatura:****Generales:**

- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CG12-Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Comprender los algoritmos evolutivos y sus variantes, profundizando en los detalles del enfoque evolutivo para la resolución de problemas. (CG11, CG12)

Desarrollar aplicaciones basadas en algoritmos genéticos. (CG11, CG12, CT1, CT2, CT3)

Desarrollar aplicaciones basadas en algoritmos evolutivos. (CG11, CG12, CT1, CT2, CT3)

Desarrollar aplicaciones basadas en programación genética y gramáticas evolutivas. (CG11, CG12, CT1, CT2, CT3)

**Evaluación detallada:**

Convocatoria ordinaria/extraordinaria:

Prácticas (75%) + Actividad adicional (25%)

SE REQUIERE NIVEL AVANZADO DE LENGUAJE JAVA Y CONOCIMIENTOS DE INTERFACES GRÁFICAS DE USUARIO PARA PODER REALIZAR LAS PRÁCTICAS.

Es necesario que todas las prácticas obtengan una calificación igual o superior a 5 sobre 10.

La calificación, tanto en la convocatoria ordinaria como extraordinaria, tiene en cuenta las siguientes actividades y pruebas a realizar por el estudiante:

A) Prácticas obligatorias se deben presentar y defender en las fechas establecidas durante el curso: Tres prácticas obligatorias realizadas en parejas, pero defendidas individualmente: 75%.

La defensa y evaluación de cada práctica puede incluir adicionalmente un examen-cuestionario sobre dicha práctica.

B) Actividades adicionales: Presentación de trabajos, exposición de temas y artículos científicos relacionados con la asignatura, uso de entornos/herramientas de computación evolutiva, presentaciones de prácticas, cuestionarios y ejercicios en el aula, participación: 25%.

La convocatoria extraordinaria dispondrá de un nuevo plazo para la defensa de las prácticas suspensas o no entregadas en plazo establecido. Dicha defensa incluye un examen sobre toda la materia relacionada con las prácticas. Las actividades adicionales se realizan durante el curso y no son recuperables en la convocatoria extraordinaria.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

Zbigniew Michalewicz.; Genetic Algorithms + Data Structures = Evolution Programs.; Springer-Verlag, 1996.;

Algoritmos Evolutivos: un enfoque práctico. Lourdes Araujo, Carlos Cervigón. RAMA-2009.

Algoritmos Evolutivos: teoría y casos prácticos. Lourdes Araujo, Carlos Cervigón. Kindle Edition Amazon.

David E. Goldberg.; Genetic Algorithms in Search, Optimization and Machine Learning; Addison Wesley Publishing Company, 1988;

John R. Koza; Genetic Programming; The MIT Press , 1993; Melanie Mitchell; An Introduction to Genetic Algorithms; The MIT Press, 1998;

Michael O'Neill, Conor Ryan. Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language (Genetic Programming). 2003. Springer.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803300 - Aprendizaje automático y Big Data <b>Asignatura en Inglés:</b> Machine learning and big data	<b>Abrev:</b> AA <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de ingeniería de software e inteligencia artificial	<b>18 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Ingeniería web Programación evolutiva	6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> González Calero, Pedro Antonio

**Descripción de contenidos mínimos:**

Minería de datos.  
Aprendizaje automático.  
Estructuras de decisión.  
Métodos de clasificación  
Redes neuronales.

**Programa detallado:**

1. Introducción a la minería de datos y el aprendizaje automático.
2. Adquisición, limpieza y análisis de datos.
3. Visualización de la información.
4. Conceptos generales de aprendizaje automático.
5. Métodos de regresión.
6. Support Vector Machines.
7. Árboles de decisión.
8. Evaluación de los sistemas de aprendizaje automático.
9. Clasificación bayesiana.
10. Redes neuronales.
11. Diseño de sistemas de aprendizaje automático.
12. Aprendizaje no supervisado.

**Programa detallado en inglés:**

1. Introduction to data mining and machine learning
2. Acquiring, cleaning and analyzing data
3. Information visualization
4. Fundamentals of machine learning
5. Regression methods
6. Support Vector Machines
7. Decision trees
8. Evaluation
9. Bayes Classifiers
10. Neural networks
11. Machine learning system design
12. Unsupervised learning

**Competencias de la asignatura:****Generales:**

- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CG12-Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.
- CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Diseñar sistemas de aprendizaje automático. (CG11, CG12, CG23, CT1, CT2, CT3)

Desarrollo de sistemas de aprendizaje automático. (CG11, CG12, CG23, CT1, CT2, CT3)

Diferenciar entre aprendizaje supervisado y no supervisado. (CG11, CG12, CG23)

**Evaluación detallada:**

La evaluación se realizará mediante una prueba final, tanto en la convocatoria ordinaria como en la extraordinaria.

La prueba, en ambas convocatorias, tendrá lugar en el laboratorio donde los estudiantes tendrán que resolver un problema utilizando el código que hayan desarrollado a lo largo del curso en las prácticas de la asignatura.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani; An Introduction to Statistical Learning with Applications in R; Springer, 2013. <http://faculty.marshall.usc.edu/gareth-james/ISL/>
- Wes McKinney; Python for Data Analysis, 2nd Edition; O'Reilly Media, 2017
- Aurélien Géron; Hands-On Machine Learning with Scikit-Learn and TensorFlow; O'Reilly Media, 2017
- Sebastian Raschka, Vahid Mirjalili; Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow, 2nd Edition; Packt Publishing, 2017
- Armando Fandango; Python Data Analysis, Second Edition; Packt Publishing, 2017
- Ian H. Witten, Eibe Frank, Mark A. Hall and Christopher J. Pal; Data Mining: Practical Machine Learning Tools and Techniques, Fourth Edition; Morgan Kaufmann, 2016.
- Trevor Hastie, Robert Tibshirani and Jerome Friedman; The Elements of Statistical Learning, 2nd edition; Springer, 2009. <https://web.stanford.edu/~hastie/ElemStatLearn/>
- Christopher Bishop; Pattern Recognition and Machine Learning; Springer, 2006.
- Tom M. Mitchell; Machine Learning; McGraw-Hill, 1997.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803304 - Seguridad en redes <b>Asignatura en Inglés:</b> Network security	<b>Abrev:</b> SER <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de computadores	12 ECTS	
<b>Otras asignaturas en la misma materia:</b> Programación paralela para móviles y multicores	6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Huedo Cuesta, Eduardo

**Descripción de contenidos mínimos:**

Redes avanzadas  
Comunicaciones seguras  
Protección redes y sistemas en red

**Programa detallado:**

TEORÍA  
1. Introducción  
2. Criptografía de clave secreta  
3. Funciones resumen  
4. Criptografía de clave pública  
5. Ataques en redes  
6. Cortafuegos  
7. Detección de intrusos  
8. Conexiones seguras  
9. Seguridad en redes inalámbricas  
10. Seguridad en la web  
11. Seguridad en DNS  
12. Seguridad en correo electrónico

**Programa detallado en inglés:**

THEORY  
1. Introduction  
2. Secret key cryptography  
3. Hash functions  
4. Public key cryptography  
5. Network attacks  
6. Firewalls  
7. Intrusion detection  
8. Secure connections  
9. Wireless networks security  
10. Web security  
11. DNS security  
12. Email security

**Competencias de la asignatura:****Generales:**

CG7-Capacidad para diseñar, desarrollar, seleccionar y evaluar, aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a los principios éticos y a la legislación y normativa vigente.

CG10-Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.

CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

**Específicas:**

CE\_GIC6-Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos.

**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Conocer los principales elementos, vulnerabilidades y mecanismos de seguridad relacionados con la privacidad, la integridad y la autenticación de la información y los sistemas en red. (CG7, CT2, CE\_GIC6)
- Conocer, entender y aplicar los principales mecanismos criptográficos utilizados en comunicaciones seguras y aplicados al cifrado, la firma digital y los certificados digitales. (CG7, CG16, CT2)
- Conocer, analizar y evaluar los principales ataques y amenazas a redes y sistemas en red, y manejar, implantar y configurar distintas herramientas y servicios de seguridad para prevenir, detectar y combatir dichos ataques y amenazas. (CG7, CG10, CG16, CT2, CT3, CE\_GIC6)
- Conocer, analizar y evaluar los problemas de seguridad de los principales servidores de Internet y diseñar, implementar y desplegar configuraciones de servidores seguros. (CG7, CG10, CG16, CT2, CT3)

**Evaluación detallada:**

- Evaluación final (convocatoria ordinaria y extraordinaria):  
Examen (60%).

- Evaluación continua (no recuperable):  
Prácticas de laboratorio (40%).

Para aprobar es necesario obtener al menos 4 puntos sobre 10 en la nota del examen.

**Actividades docentes:**

Reparto de créditos:	Otras actividades:
Teoría: 3,60	No tiene
Problemas: 0,00	
Laboratorios: 2,40	

**Bibliografía:**

- E. Cole. Network Security Bible, 2nd Edition. Ed. John Wiley & Sons. 2009
- J. Vacca. Computer and Information Security Handbook. Ed. Morgan Kaufmann. 2009
- B. Burns y otros. Security Power Tools. Ed. O'Reilly. 2007
- S. MacClure y otros. Hacking exposed 6. Ed. MacGraw Hill. 2009
- R. Johnson and M. Merkow. Security Policies and Implementation Issues. Ed. Jones & Bartlett Learning. 2010
- S. Harris, F. Maymí, Mc Graw Hill, All in one CISSP, exam guide, 7ª edición 2016
- William Stallings Network Security Essentials: Applications and Standards, Prentice Hall, 2013
- J. Michael Stewart, Jones & Bartlett Learning, Network Security, Firewalls, and VPNs, 2014
- Ruby B. Lee, Security Basics for Computer Architects, Synthesis Lectures on Computer Architecture, 2013

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:

**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803305 - Criptografía y teoría de códigos <b>Asignatura en Inglés:</b> Cryptography and code theory	<b>Abrev:</b> CTC <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de matemáticas e investigación operativa	<b>18 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Análisis numérico Investigación Operativa	6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Álgebra, Geometría y Topología		<b>Coordinador:</b> Alonso García, Mª Emilia

**Descripción de contenidos mínimos:**

Criptografía y Teoría de Códigos

**Programa detallado:**

1. Algoritmos básicos de la aritmética de enteros y anillos de polinomios con coeficientes en un cuerpo. Complejidad binaria.
2. Cuerpos finitos. Caracterización y representación.
3. Códigos correctores de errores. Distancia de Hamming y cotas.
4. Códigos lineales. Algunas familias de códigos, Problema de la decodificación de códigos lineales.
5. Códigos cíclicos. Construcción de códigos cíclicos : códigos BCH y de Reed Salomón. Decodificación de BCH con algoritmo Berlekamp-Massey.
6. Conceptos básicos y tareas de la Criptografía de Clave Pública. Criptografía clásica.
7. Cifrado en flujo, LFSR's y ataques.
8. Funciones de una dirección. Funciones resumen (" hash"). Complejidad de problemas  $P$  y  $NP$ , en la aritmética de enteros. Criptografía de Clave Pública. Autenticación. Firma digital.
9. Sistemas criptográficos basados en el problema del logaritmo discreto (DLP). Protocolo de Diffie-Hellmann-Meckle. Sistema "El Gamal", DSS y otros protocolos basados en DLP. Ataques a DLP.
10. Sistemas criptográficos basados en el problema de la factorización de enteros. RSA. Protocolos basados en RSA. Ataques a RSA. Algoritmo "Rho" de Pollard y algoritmo QS.
12. Otros protocolos: Prueba sin conocimiento, votación electrónica, dinero digital.

**Programa detallado en inglés:**

First part:

Elementary algorithms for integer arithmetic and polynomials arithmetic over a field.  
Binary complexity of EEA. Finite fields: characterization and representation.  
Libraries in Maple and SAGE. Error-correcting codes. Hamming distance. Some bounds.  
Linear codes, cyclic codes, BCH codes, Reed Salomon codes. The problem of de-codification.

Second part:

Basic concepts on Cryptography and its history. Symetric Cryptography versus public Cryptography. Stream Ciphers. Complexity of problems in Arithmetic and Combinatorics:  $P$  and  $NP$ . One way functions, hash functions.  
Public key Cryptography based on DLP. Some attacks and protocols: DSS.  
Public key Cryptography based on the factorization problem: RSA. Attacks: modern integer factorization algorithms. Zero knowledge protocols. Electronic voting, digital cash.

**Competencias de la asignatura:****Generales:**

CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

**Resultados de aprendizaje:**

Aprender y practicar los conceptos complementarios de Criptografía y teoría de códigos (CG1, CT2)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Evaluación detallada:**

La evaluación de la asignatura se realizará vía : examen, entrega de ejercicios y práctica de programación. Con los porcentajes que se expresan a continuación:

- La práctica de programación se realizará en grupos de 2 a 4 alumnos cuyo número de integrantes dependerá del número de alumnos que asistan asiduamente a clase y deseen realizarla. Ésta constituirá un 40% de la calificación final.

La evaluación de dicha práctica se hará vía su presentación en grupo al profesor, y durante la realización de la misma los integrantes del grupo deben demostrar conocer los algoritmos en que dicha práctica se basa y no solo la correcta ejecución del programa. La calificación de la práctica será individual.

- La entrega de ejercicios asidua durante el correspondiente cuatrimestre constituirá un 10% de la calificación total.

- El examen se valorará en el 50% de la calificación total.

Exámenes: En Lab Final convocatoria ordinaria y extraordinaria .

En el caso en que el alumno por circunstancias especiales no pueda asistir asiduamente a clase, no tendrá opción arealizar la práctica de programación por considerar que su realización requiere una tutorización continua por parte del profesor, y se le calificará únicamente el examen y la entrega de ejercicios siempre que se asista a corregirlos personalmente a las tutorías.

La calificación de las prácticas aprobadas se conservará para la convocatoria extraordinaria. En la convocatoria extraordinaria también existirá la posibilidad de volver a entregar prácticas para quienes no las hubieran aprobado previamente.

Las calificaciones serán sobre 10.

**Actividades docentes:**

Reparto de créditos:

Teoría: 4,00

Problemas: 2,00

Laboratorios: 0,00

Otras actividades:

--Clases magistrales, apoyadas por herramienta informática de cálculo simbólico .

--Resolución de problemas individual y presentación de estos en clase.

**Bibliografía:**

- Buchmann, J.A. : "Introduction to Cryptography". Undergraduate Texts in Maths. Springer- Verlag 2nd. de. (2004).

- Gómez-Pardo, J.L.: "Introduction to Cryptography with Maple". Springer-Verlag, 2013.

- Koblitz, N.: "A course in Number Theory and Cryptography". Springer- Verlag 2nd. ed., 1994. (GTM 1149).

- Lidl, R., Gunter, P.: "Applied Abstract Algebra". 2nd. ed. Springer 1997.

-N. P. Smart: Cryptography made simple (a través de la Bibli. de la UCM:

<http://link.springer.com/book/10.1007/978-3-319-21936-3>.

- Stinson D. R. : "Cryptography Theory and Practice. 3rd. Ed . In "Discrete Mathematics and its Applications". Taylor&Francis, LLC, CRC Press (2005).

- Trappe W. Washington L.: "Cryptography with Coding Theory". Prentice Hall; 2nd. ed. ( 2005)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803307 - Investigación Operativa <b>Asignatura en Inglés:</b> Operational Research	<b>Abrev:</b> IO <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de matemáticas e investigación operativa	18 ECTS	
<b>Otras asignaturas en la misma materia:</b> Análisis numérico Criptografía y teoría de códigos	6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Estadística e Investigación Operativa		<b>Coordinador:</b> Rodríguez Sousa, Antonio Alberto

**Descripción de contenidos mínimos:**

Introducción a la Investigación Operativa.  
Programación Lineal. Algoritmo del simplex.  
Programación entera. Algoritmos de ramificación y acotación, planos de corte y Algoritmo de Balas para problemas binarios.  
Grafos. Problemas sobre grafos.

**Programa detallado:**

- 1.- Modelización matemática de problemas de optimización. Software de optimización.
- 2.- Programación lineal:
  - 2.1.- Método Simplex.
  - 2.2.- Método Simplex Dual.
- 3.- Programación lineal entera:
  - 3.1.- Método de planos de corte.
  - 3.2.- Método de ramificación y acotación.
  - 3.3.- Método de ramificación y corte.
- 4.- Programación no lineal:
  - 4.1.- Optimalidad de Karush-Kuhn-Tucker.
  - 4.2.- Algoritmos de optimización sin restricciones sin diferenciación (algoritmos de comparación, coordenadas cíclicas, Nelder-meade, entre otros).
  - 4.3.- Algoritmos de optimización sin restricciones con diferenciación: Newton y cuasi-Newton, descenso de gradiente, gradiente conjugado, descenso de gradiente estocástico, Adam.
  - 4.4.- Optimización con restricciones: minimización secuencial no restringida.
- 5.- Optimización en redes:
  - 5.1.- Problema del árbol soporte de peso mínimo.
  - 5.2.- Problemas de camino mínimo.
  - 5.3.- Problemas de flujo óptimo.
  - 5.4.- Problemas de optimización de recorridos en redes.
- 6.- Heurísticas para resolver problemas de optimización.

**Programa detallado en inglés:**

- 1.- Mathematical modeling of optimization problems. Optimization software.
- 2.- Linear programming:
  - 2.1.- Simplex Method.
  - 2.2.- Dual Simplex Method.
- 3.- Integer linear programming:
  - 3.1.- Cutting plane method.
  - 3.2.- Branching and dimensioning method.
  - 3.3.- Branching and cutting method.
- 4.- Non-linear programming:
  - 4.1.- Karush-Kuhn-Tucker optimality.
  - 4.2.- Unconstrained optimization algorithms without differentiation (comparison algorithms, cyclic coordinates, Nelder mead, among others).
  - 4.3.- Unconstrained optimization algorithms with differentiation: Newton and quasi-Newton, gradient descent, conjugate gradient, stochastic gradient descent, Adam.
  - 4.4.- Constrained optimization: sequential unconstrained minimization.
- 5.- Optimization in networks:
  - 5.1.- Minimum weight support tree problem.
  - 5.2.- Minimum path problems.
  - 5.3.- Optimal flow problems.
  - 5.4.- Path optimization problems in networks.
- 6.- Heuristics to solve optimization problems.

**Competencias de la asignatura:****Generales:**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.
<b>Específicas:</b> No tiene
<b>Básicas y Transversales:</b> CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales. CT2-Capacidad de análisis y síntesis en la resolución de problemas. CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
<b>Resultados de aprendizaje:</b> Clasificar un problema en función de su solución y resolverlo. (CG1) Realizar en equipo los problemas prácticos propuestos (CT1) Reconocer un problema de la investigación operativa y saber resolverlo. (CG1) Identificar problemas informáticos nuevos con problemas de investigación operativa ya conocidos, utilizando las técnicas de resolución de estos últimos para resolverlos. (CT3) Aplicar los distintos métodos operativos para resolver problemas de Investigación Operativa. (CT2) Resolver ejercicios con el software adecuado analizando el problema y aplicando la solución. (CT2, CT3)
<b>Evaluación detallada:</b> Realización de una prueba final escrita: ponderación 60%. Será necesaria la obtención de una nota mínima de 4.00 sobre 10.00 para poder hacer media y superar la asignatura. Entrega de ejercicios o problemas propuestos por el profesor: Ponderación 40%. Se requerirá la nota media final de un 5.00 sobre 10.00 para poder superar la asignatura
<b>Actividades docentes:</b> Reparto de créditos:                      Otras actividades: Teoría: 3,00                                      - Clases presenciales Problemas: 2,00                                  - Clases de problemas Laboratorios: 1,00                              - Tutorías de prácticas y de programación
<b>Bibliografía:</b> 1.- Bazaraa, M.S., Jarvis, J.J., Sherali, H.D. (1998). Programación lineal y flujo en redes. Limusa 2.- Cormen, T.H., Leiserson, C.E., Rivest, R.L. (2009) Introduction to Algorithms. MIT Press, Cambridge Massachusetts. 3.- Hillier, F.S. & Lieberman, G.J. (2015). Introduction to Operations research. McGraw-Hill 4.- Kochendefor, A.J., & Wheeler, T.A. (2019). Algorithms for optimization. MIT Press 5.- Nocedal, J., & Wright, S.J. (2006). Numerical optimization. Springer 6.- Taha, H.A. (2012). Investigación en operaciones. Pearson

Ficha docente guardada por última vez el 30/06/2023 12:29:00 por el departamento: Estadística e Investigación Operativa

Fecha: ____ de _____ de ____
Firma del Director del Departamento:





Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803347 - Minería de datos y el paradigma Big Data <b>Asignatura en Inglés:</b> Data mining and the Big Data paradigm		<b>Abrev:</b> MIN <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de gestión y procesamiento de la información		24 ECTS	
<b>Otras asignaturas en la misma materia:</b> Análisis de redes sociales Bases de Datos noSQL Gestión de la información en la web		6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Requeno Jarabo, José Ignacio	

**Descripción de contenidos mínimos:**

- Conceptos básicos y/o claves de la inteligencia del negocio o inteligencia empresarial (business intelligence)
- Paradigmas complementarios al modelo relacional para la representación de datos, información y conocimiento: modelo multidimensional y almacenes de datos (data warehouse); datos abiertos y/o enlazados; datos textuales, semiestructurados y/o no estructurados; grandes volúmenes de datos ("big data").
- Principales técnicas de obtención, representación, procesamiento y explotación de los paradigmas mencionados anteriormente.
- Minería de datos: Metodologías, procesos, técnicas y modelos principales; campos de aplicación.

**Programa detallado:**

La expansión de la WWW y el crecimiento exponencial en la capacidad de almacenamiento y procesamiento de los sistemas de información actuales han abierto nuevas vías para la representación y explotación de datos impensables hasta no hace muchos años.

De esta forma, han surgido recientemente paradigmas y conceptos como "data warehouse", datos abiertos, datos enlazados, datos no estructurados o textuales, o "big data", que se han unido a los tradicionales datos del paradigma relacional en las necesidades de almacenamiento y procesamiento de los datos y la información, sobre todo del mundo empresarial y de los negocios. Estas necesidades han venido impulsadas en gran medida, por ejemplo, por los requisitos que plantea una gestión de la Inteligencia del Negocio más moderna y automatizada.

Acompañando a esta evolución en los paradigmas de representación y/o almacenamiento de los datos y la información han surgido también nuevas formas de procesamiento y/o explotación de los mismos. Entre ellas figura, de manera privilegiada, la minería de datos, con sus propias metodologías (como KDD, SEMMA o CRISP-DM) y técnicas (como clustering o asociación) relativamente consolidadas, para la deducción y/o extracción automática de información y conocimiento de estos.

La asignatura contextualiza y presenta todos estos nuevos paradigmas de representación de datos y de la información, junto con los problemas que plantean y las soluciones provistas hasta la fecha para su solución, en forma de metodologías, técnicas y buenas prácticas para el desarrollo y explotación de sistemas de información (es decir, de minería de datos y de textos) apropiados.

**OBJETIVOS:**

El objetivo de esta asignatura es familiarizar al alumnado con los distintos paradigmas actuales de representación de datos, prestando especial atención a aquellos que posibilitan o implican el manejo de grandes volúmenes de datos (big data), así como con las principales metodologías, técnicas y buenas prácticas existentes para su manejo, procesamiento y/o explotación dentro del campo de la minería de datos.

**TEMARIO:**

1. Introducción y conceptos básicos: inteligencia del negocio; paradigmas de representación de datos, información y conocimiento (evolución de las bases de datos); campos de aplicación de la minería de datos; técnicas de minería de datos.

2.- Almacenamiento de datos: paradigmas complementarios al modelo relacional.

2.1.- El paradigma de los "data warehouse".

2.1.1.- Introducción a los almacenes de datos.

2.1.2.- El modelo multidimensional: esquemas multidimensionales, cuboides y operaciones.

2.1.3.- Arquitecturas multidimensionales: OLAP versus OLTP; ROLAP y otras formas de gestión.

2.2.- El paradigma de los datos abiertos y/o enlazados: archivos CSV, XML, RDF y derivados.

2.4.- El paradigma de los datos textuales y/o no estructurados: problemas y procesos clave en el procesamiento del lenguaje natural (PLN).

2.5.- El paradigma de los grandes volúmenes de datos ("big data"). Bases de datos distribuidas y/o NoSQL: introducción a MapReduce, Hadoop, MongoDB y/o BigTable.

3.- Minería de datos.

3.1.- Introducción y conceptos básicos.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- 3.2.- Metodologías de desarrollo de sistemas de minería de datos: KDD, SEMMA y/o CRISP-DM.
- 3.3.- Procesos y técnicas claves en el desarrollo de sistemas de minería de datos.
  - 3.3.1.- Importación, preprocesamiento y exportación de datos; herramientas ETL (extract, transfer and load).
  - 3.3.2.- Exploración y visualización básica de datos.
  - 3.3.3.- Modelos de transformación y/o procesamiento de datos.
    - 3.3.3.1.- Árboles de decisión y bosques aleatorios.
    - 3.3.3.2.- Regresión.
    - 3.3.3.3.- Agrupamiento.
    - 3.3.3.4.- Reglas de asociación.
  - 3.3.4.- Minería de textos.
    - 3.3.4.1.- Adquisición de datos textuales: web crawling y minería de páginas web y redes sociales.
    - 3.3.4.2.- Procesamiento automático de textos: procesos de lematización, radicalización, etc.
  - 3.3.5.- Minería de grafos: explotación y aplicaciones de los repositorios de datos enlazados. Introducción a SPARQL.
  - 3.3.6.- Evaluación.

**Programa detallado en inglés:**

The expansion of the WWW and the exponential growth in storage capacity and processing capabilities of current information systems has given rise to the creation of new data representation formats and data exploitation means that were inconceivable some years ago.

In this way, some new data-related concepts and their related paradigms have recently emerged, such as "data warehouse", open data, linked data, unstructured or textual data, or "big data". All of them, together with the traditional relational data paradigm, are currently involved in the data and information storage and processing needs of business and enterprises. These needs have been driven, to a large extent, for example, by the requirements posed by a more modern and automated Business Intelligence management.

In order to make up for this evolution in the paradigms and formats of representation and/or storage of data and information, some new means for their processing and/or exploitation have also emerged. The main one is, most probably, data mining, which has already developed its own methodologies (such as KDD, SEMMA or CRISP-DM) and techniques (such as clustering or association rules) for the deduction and/or automatic extraction of information and knowledge from these data formats.

This course seeks to contextualize and present all these new data and information representation formats and paradigms, as well as the problems they pose and the solutions found to date to solve them, by means of suitable methodologies, techniques and good practices for the development and exploitation of information systems (i.e., data and text mining).

**GOALS:**

The main goal of this course is to make students aware of

- the current data representation paradigms, paying special attention to those that involve big data management and/or processing, as well as
- the main methodologies, techniques and good practices identified so far for their management, processing and/or exploitation within the data mining field.

**CONTENTS:**

- 1. Introduction and basic concepts: business intelligence; data, information and knowledge representation paradigms (evolution of databases); areas of data mining applications; data mining techniques.
- 2.- Data storage: relational model supplementary paradigms.
  - 2.1.- The data warehouse paradigm.
    - 2.1.1.- Introduction to data warehouses.
    - 2.1.2.- The multidimensional model: multidimensional schemes, cuboids and operations.
    - 2.1.3.- Multidimensional architectures: OLAP vs. OLTP; ROLAP and other management techniques.
  - 2.2.- The open and/or linked data paradigm: CSV, XML and RDF files (and derivatives).
  - 2.4.- The textual and/or unstructured data paradigm: problems and key processes in natural language processing (NLP).
  - 2.5.- The big data paradigm. Distributed and/or NoSQL databases: introduction to MapReduce, Hadoop, MongoDB and/or BigTable.
- 3.- Data mining.
  - 3.1.- Introduction and basic concepts.
  - 3.2.- Methodologies for data mining system development: KDD, SEMMA and/or CRISP-DM.
  - 3.3.- Key processes and techniques in data mining system development.
    - 3.3.1.- Data import, preprocessing and export; ETL tools (extract, transfer and load).
    - 3.3.2.- Data exploration and basic visualization.
    - 3.3.3.- Data transformation models and/or processing.
      - 3.3.3.1.- Decision trees and random forests.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



- 3.3.3.2.- Regression.
- 3.3.3.3.- Clustering.
- 3.3.3.4.- Association rules.
- 3.3.4.- Text mining.
- 3.3.4.1.- Text acquisition: web crawling, web page and social network mining.
- 3.3.4.2.- Automatic word and text processing: lemmatization, stemming, etc.
- 3.3.5.- Graph mining: exploitation and applications of linked data repositories. Introduction to SPARQL.
- 3.3.6.- Assessment.

**Competencias de la asignatura:**

**Generales:**

- CG6-Conocimiento adecuado del concepto de empresa, marco institucional y jurídico de la empresa. Organización y gestión de empresas.
- CG17-Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.
- CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
- CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

**Específicas:**

- CE\_C1-Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar, y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática.
- CE\_C4-Capacidad para conocer los fundamentos, paradigmas y técnicas propias de los sistemas inteligentes y analizar, diseñar y construir sistemas, servicios y aplicaciones informáticas que utilicen dichas técnicas en cualquier ámbito de aplicación.
- CE\_C5-Capacidad para adquirir, obtener, formalizar y representar el conocimiento humano en una forma computable para la resolución de problemas mediante un sistema informático en cualquier ámbito de aplicación, particularmente los relacionados con aspectos de computación, percepción y actuación en ambientes o entornos inteligentes.
- CE\_C7-Capacidad para conocer y desarrollar técnicas de aprendizaje computacional y diseñar e implementar aplicaciones y sistemas que las utilicen, incluyendo las dedicadas a extracción automática de información y conocimiento a partir de grandes volúmenes de datos.
- CE\_TI1-Capacidad para comprender el entorno de una organización y sus necesidades en el ámbito de las tecnologías de la información y las comunicaciones.

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.

**Resultados de aprendizaje:**

- Comprensión de los aspectos básicos empresariales y del negocio que afectan al diseño y desarrollo de sistemas de información y/o de gestión de (bases de) datos, información y conocimientos. (CG6)
- Capacidad para identificar y analizar los elementos de la lógica del negocio que afectan al proceso de diseño y desarrollo de un sistema de inteligencia empresarial (business intelligence), minería de datos y/o de explotación de big data e integrarlos, con una metodología adecuada, en dicho proceso. (CG6, CT2, CE\_C5, CE\_TI1)
- Representar y procesar datos, información y conocimientos de forma integrada y conveniente usando metodologías, procesos, modelos propios de las áreas de la minería de datos (incluyendo la minería de textos) y del campo de los big data. (CG17, CG18, CG23, CT2, CE\_C1, CE\_C4, CE\_C5, CE\_C7)
- Conocer, aplicar y evaluar distintas técnicas y modelos de minería de datos para resolver un problema concreto. (CG23, CT2, CE\_C1, CE\_C7)
- Entender las componentes básicas de un sistema de inteligencia empresarial en general y, más concretamente, de un sistema de minería de datos, para su aplicación en el desarrollo de sistemas y aplicaciones en este ámbito. (CG17, CG18, CE\_C5, CE\_TI1)
- Elegir la(s) representación(es) más adecuada(s) del problema para aplicar las técnicas y modelos de minería de datos que lo resuelvan. (CG17, CG18, CT2)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Diseñar e implementar (preferentemente, mediante grupos de trabajo) un sistema de minería de datos y/o de procesamiento de grandes volúmenes de datos (big data) utilizando metodologías, procesos, técnicas y modelos adecuados de estas áreas. (CG17, CG18, CG23, CT1, CT2, CE\_C1, CE\_C7)

Aumento y/o mejora de la capacidad de trabajo en equipo y de realización de presentaciones orales. (CT1, CT2)

**Evaluación detallada:**

En ambas convocatorias (ordinaria y extraordinaria) la realización de las prácticas y/o proyectos es obligatoria. La(s) práctica(s) se entregarán individualmente o en grupo. La realización y entrega de la(s) práctica(s) puede conllevar su defensa (parcial y/o total) pública o en presencia del profesor.

La nota final se calculará de acuerdo a los siguientes porcentajes:

- \* 20% participación en clase: proactividad y colaboración en el desarrollo de las sesiones presenciales y de las tareas colectivas, resolución de ejercicios y cuestionarios, etc.
- \* 80% práctica(s) obligatoria(s).

- No entregar las prácticas en el plazo establecido supondrá el suspenso en la asignatura (no se calculará la media con el resto de elementos de la evaluación) en la convocatoria ordinaria.

- Existe la posibilidad de entregar las prácticas en la convocatoria extraordinaria, manteniéndose para la convocatoria extraordinaria la calificación correspondiente a la participación en clase y el resto de tareas aprobadas.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- Russell, Matthew A. (2014) Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More (2nd Edition). Sebastopol (CA, EE.UU.): O'Reilly.

- Witten, Ian H., Frank, Eibe (2005) Data Mining: Practical Machine Learning Tools and Techniques (2nd Edition). San Francisco: Morgan Kaufmann Publishers (Elsevier).

- Zhao, Yanchang (2013) R and Data Mining: Examples and Case Studies. San Diego, Waltham, Londres, Amsterdam: Academic Press (Elsevier).

- Dietrich, D. (2015). Data science and big data analytics: discovering, analyzing, visualizing and presenting data. John Wiley & Sons.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803348 - Gestión de la información en la web <b>Asignatura en Inglés:</b> Web Information Management		<b>Abrev:</b> GIW <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de gestión y procesamiento de la información		<b>24 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Análisis de redes sociales Bases de Datos noSQL Minería de datos y el paradigma Big Data		6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Martín Martín, Enrique	

**Descripción de contenidos mínimos:**

- Formatos de intercambio de datos.
- Explotación de bases de datos.
- Obtención y extracción de datos en la web.
- Seguridad en la web.

**Programa detallado:**

1. Introducción a Python.
2. Gestión de información basada en documentos XML, CSV y JSON.
3. Acceso a bases de datos relacionales en Python.
4. Obtención de datos: web scraping, API y repositorios abiertos.
5. Desarrollo de servicios web con Python.
6. Persistencia de datos en Python: ODM.
7. Seguridad en aplicaciones web.
8. Desarrollo de aplicaciones web en Python.

**Programa detallado en inglés:**

1. Introduction to Python.
2. Information management based on XML, CSV, and JSON documents.
3. Relational databases in Python.
4. Data collection: web scraping, API, and open repositories.
5. Web services development with Python.
6. Data persistence in Python: ODM.
7. Security in web applications.
8. Web application development in Python.

**Competencias de la asignatura:****Generales:**

CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

**Específicas:**

CE\_TI6-Capacidad de concebir sistemas, aplicaciones y servicios basados en tecnologías de red, incluyendo Internet, web, comercio electrónico, multimedia, servicios interactivos y computación móvil.

CE\_TI7-Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos.

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Conocer cómo implementar mecanismos de explotación de información que se encuentra accesible en la web. (CG18)

Ser capaz de colaborar en el contexto de un equipo para desarrollar una aplicación informática. (CT1)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ser capaz de resolver problemas mediante el análisis de la información y la síntesis. (CT3)

Saber cómo crear una aplicación web que gestione información. (CE\_TI6)

Conocer e implementar mecanismos de seguridad de la información en una aplicación web. (CE\_TI7)

**Evaluación detallada:**

La nota final estará compuesta al 100% por las prácticas en laboratorio. Existirá un periodo de entrega de prácticas en la convocatoria extraordinaria para aquellas prácticas suspensas o que no fueron entregadas a tiempo.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

**BÁSICA:**

- Practical Programming: An Introduction to Computer Science Using Python 3.6 (Third Edition). Paul Gries, Jennifer Campbell, Jason Montojo. Pragmatic Bookshelf, 2017.
- Gestión de la Información Web. Antonio Sarasa Cabezuelo. Editorial UOC. 2016.
- BIG DATA con PYTHON. Recolección, almacenamiento y proceso. Rafael Caballero, Enrique Martín y Adrián Riesco. RC libros, 2018.
- Introducción a las bases de datos NoSQL usando MongoDB. Antonio Sarasa Cabezuelo. Editorial UOC. 2016
- Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement (Second Edition). Luc Perkins, Jim Wilson, Eric Redmond. Pragmatic Bookshelf, 2018.
- Hands-On RESTful API Design Patterns and Best Practices. Harihara Subramanian, Pethuru Raj. Packt Publishing, 2019.
- The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws (Second Edition). Dafydd Stuttard, Marcus Pinto. Wiley Publishing, 2011.

**COMPLEMENTARIA:**

- Introduction to Computation and Programming Using Python (Second Edition). John V. Guttag. MIT Press, 2017.
- Fundamentos de bases de datos (Sexta Edición). Abraham Silberschatz, Henry F. Korth, S. Sudarshan. McGraw-Hill, 2014.
- REST API Design Rulebook. Mark Masse. O'Reilly, 2011.
- Foundations of Security: What Every Programmer Needs to Know. Christoph Kern, Anita Kesavan, Neil Daswani. Apress, 2007.
- Big Data. Análisis de datos con Python. Yolanda García Ruiz, Antonio Sarasa Cabezuelo. Editorial Garceta. 2017.

Ficha docente guardada por última vez el 20/07/2020 17:01:00 por el usuario: **Coordinador GII**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:

**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803352 - Interfaces de usuario <b>Asignatura en Inglés:</b> User interfaces	<b>Abrev:</b> IU <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de ingeniería del software y sistemas de información	<b>6 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> No hay		
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> Freire Morán, Manuel

**Descripción de contenidos mínimos:**

- Usabilidad
- Guías, Principios y Paradigmas de Diseño
- Mecanismos y dispositivos de interacción
- Ingeniería de la Información
- Patrones de diseño

**Programa detallado:**

1. Usabilidad
2. Teorías, principios y guías de diseño
3. Reglas y paradigmas de diseño
4. Mecanismos de interacción
5. Patrones de diseño
6. Accesibilidad
7. Desarrollo de interfaces web

**Programa detallado en inglés:**

1. Usability
2. Theories, principles, and design guidelines
3. Design rules and paradigms
4. Interaction mechanisms
5. Design patterns
6. Accessibility
7. Web UI development

**Competencias de la asignatura:****Generales:**

CG21-Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Desarrollar interfaces de usuario usables. (CG21, CT1, CT2, CT3)

Desarrollar interfaces de usuario accesibles. (CG21, CT1, CT2, CT3)

Conocer los patrones de interacción persona ordenador y mecanismos de implementación adecuados de los mismos en interfaces de usuario. (CG21)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

**Evaluación detallada:**

Convocatoria ordinaria:

- Evaluación continua mediante la realización y defensa de prácticas realizadas en grupo: 100%. Para poder ser evaluado mediante prácticas el alumno debe asistir al 90% de las clases.
- Los alumnos que no aprueban las practicas o que no asistan al 90% de las clases podrán realizar una prueba adicional por el 100% de la asignatura.

Convocatoria extraordinaria:

- Prueba adicional: 100%.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Clases teóricas: Enseñanza presencial teórica.

Laboratorios: Realización de prácticas tutorizadas en el laboratorio.

Trabajos dirigidos: Realización de trabajos individuales o colectivos con presentaciones en clase.

**Bibliografía:**

- The essential guide to user interface design : an introduction to guide design principles and techniques. Wilbert Galitz. John Wiley & Sons. 2007.
- Designing Interfaces. Patterns for Effective Interaction Design (2nd Edition). Jennifer Tidwell. O'Reilly. 2011.
- About face 3 : the essentials of interaction design. Alan Cooper, Robert Reimann, and Dave Cronin. Wiley, cop. 2007

Ficha docente guardada por última vez el 26/06/2023 16:10:00 por el usuario: Coordinador GIS

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803359 - Creación de empresas <b>Asignatura en Inglés:</b> NO	<b>Abrev:</b> CE <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de estadística, finanzas y empresa	<b>6 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> No hay		
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Administración Financiera y Contabilidad		<b>Coordinador:</b> Pascual Ezama, David

**Descripción de contenidos mínimos:**

Matemática financiera.  
Creación de empresas.  
Estadística Computacional.

**Programa detallado:****TEMA I CONSTITUCIÓN DE LA EMPRESA**

Tipos de sociedades y otras alternativas; Requisitos legales; Procedimientos administrativos

**TEMA II EL PLAN DE NEGOCIO**

Introducción al plan de negocio. Equipo promotor; Descripción del negocio. Modelo de negocio; Estudio de mercado; Descripción comercial. Plan de marketing; Descripción técnica; Plan de compras ; Organización de RRHH ; Estructura legal; Estudio económico financiero; La elección de las fuentes de financiación

**TEMA III ANÁLISIS DE VIABILIDAD EMPRESARIAL: ANÁLISIS FINANCIERO**

Análisis de Balance de Situación; Cálculo del Periodo Medio de Maduración (PMM); Clasificación Funcional del Balance; Análisis del Fondo de Rotación; Condiciones del Equilibrio Financiero; Análisis complementario a través de indicadores financieros

**TEMA IV ANÁLISIS DE VIABILIDAD EMPRESARIAL: ANÁLISIS ECONÓMICO**

Análisis de la cuenta de resultados; Clasificación Funcional de la cuenta de resultados; Cálculo del Punto Muerto (PM); Análisis del Punto Muerto; Análisis de la sensibilidad del beneficio; Análisis del apalancamiento económico de la empresa

**TEMA V CALIDAD Y GESTIÓN DE LOS BENEFICIOS**

Flujos de Ingresos vs Flujos de Caja; Calidad de los Beneficios, Gestión de los Beneficios; Beneficios por Acción

**TEMA VI ANÁLISIS DE LIQUIDEZ**

Concepto de Liquidez; Liquidez y Capital Circulante; Análisis del periodo medio de maduración; Ratios de liquidez. La liquidez a corto plazo

**TEMA VII LA SOLVENCIA EMPRESARIAL**

Noción de solvencia; Endeudamiento y solvencia a largo plazo; Financiación y apalancamiento; Solvencia, beneficio y recursos generados; El proceso de fracaso empresarial; Insolvencia en las PYMES

**TEMA VIII LA RENTABILIDAD EMPRESARIAL**

Noción de rentabilidad; Rentabilidad de los activos; Rentabilidad de los fondos propios

**Programa detallado en inglés:**

No tiene

**Competencias de la asignatura:****Generales:**

CG6-Conocimiento adecuado del concepto de empresa, marco institucional y jurídico de la empresa. Organización y gestión de empresas.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:

**Resultados de aprendizaje:**

Saber cómo constituir una empresa. (CG6, CT1, CT4)

Saber llevar a cabo análisis financieros y económicos de una empresa. (CG6, CT1, CT2)

Saber gestionar los recursos económicos de una empresa para hacerla rentable. (CG6, CT1, CT2, CT4)

**Evaluación detallada:**

La asistencia será obligatoria al menos en un 80%. La calificación final tanto en convocatoria ordinaria como en convocatoria extraordinaria tendrá en cuenta:

- Exámenes sobre la materia: 50%
- Otras actividades: 50%

Para poder hacer la media es necesario obtener un mínimo de 4 en el examen final de la convocatoria ordinaria o extraordinaria.

En el apartado "Otras actividades" se podrá valorar la participación activa en el proceso de aprendizaje, la realización de prácticas y ejercicios y la realización de otras actividades dirigidas. La realización de las prácticas será obligatoria. La calificación reflejará los resultados de aprendizaje de las diferentes competencias que se adquieren en el módulo o materia. Estas competencias deberán ser suficientes para una vez superada la materia el alumno tenga los conocimientos necesarios para poder comenzar la actividad empresarial por cuenta propia.

La calificación obtenida en el apartado "Otras actividades" en la convocatoria ordinaria se mantendrá para la convocatoria extraordinaria sin posibilidad de cambio.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 3,00

Laboratorios: 0,00

Otras actividades:

En el apartado "Otras actividades" se podrá valorar la participación activa en el proceso de aprendizaje, la realización de prácticas y ejercicios y la realización de otras actividades dirigidas. La realización de las prácticas será obligatoria. La calificación reflejará los resultados de aprendizaje de las diferentes competencias que se adquieren en el módulo o materia. Estas competencias deberán ser suficientes para una vez superada la materia el alumno tenga los conocimientos necesarios para poder comenzar la actividad empresarial por cuenta propia.

**Bibliografía:**

- Pascual-Ezama D. (2010). Introducción al estudio del análisis financiero y económico. CERSA. Madrid. ISBN 13: 978-84-92943-21-0
- Pascual-Ezama D. (2010). Estudio del análisis financiero y económico: información interna vs información externa. CERSA. Madrid. ISBN 13: 978-84-92943-22-7
  - Pascual-Ezama D. (2010). Análisis de estados financieros: calidad y gestión de los beneficios. CERSA. Madrid. ISBN 13: 978-84-92539-97-0
  - Pascual-Ezama D. (2010). Análisis y viabilidad empresarial: método del caso interactivo. CERSA. Madrid. ISBN 13: 978-84-92943-20-3
  - Pascual-Ezama D. (2010). Proyección de estados contables: método del caso. CERSA. Madrid. ISBN 13: 978-84-92943-23-4
  - Goxens, M. Angeles y Gay J. María (1999). Análisis de Estados Contables, Prentice-Hall.
  - Camacho, M<sup>a</sup> del Mar y Rivero, M<sup>a</sup> José: Introducción a la Contabilidad Financiera en el Marco del EEES, Editorial Pearson, Madrid, 2010.

Enlaces de Interés:

- REGISTRO MERCANTIL: [www.rmc.es](http://www.rmc.es)
- CNMV, COMISIÓN NACIONAL DEL MERCADO DE VALORES: [www.cnmv.es](http://www.cnmv.es)
- FASB, FINANCIAL ACCOUNTING STANDARDS BOARD / US GAAP: [www.fasb.org](http://www.fasb.org)
- ICAC, INSTITUTO DE CONTABILIDAD Y AUDITORÍA DE CUENTAS: [www.icac.mineco.es](http://www.icac.mineco.es)
- [www.noticiasjuridicas.com](http://www.noticiasjuridicas.com)
- [www.cef.es](http://www.cef.es)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803364 - Análisis de redes sociales <b>Asignatura en Inglés:</b> Social Network Analysis		<b>Abrev:</b> SOC <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de gestión y procesamiento de la información		24 ECTS	
<b>Otras asignaturas en la misma materia:</b> Bases de Datos noSQL Gestión de la información en la web Minería de datos y el paradigma Big Data		6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> Arroyo Gallardo, Javier	

**Descripción de contenidos mínimos:**

Mecanismos de representación de redes sociales.  
Análisis de redes sociales.  
Modelos de redes sociales.

**Programa detallado:**

1. Introducción
2. Propiedades básicas de las redes
3. Centralidad
4. Modularidad y detección de comunidades
5. Descubrimiento de patrones
6. Modelos de redes: aleatorias y libres de escala
7. Dinámica de las redes: Propagación y difusión

**Programa detallado en inglés:**

1. Introduction
2. Network basics
3. Network centrality
4. Modularity and community detection
5. Motif discovery
6. Network models: Random networks and scale-free networks
7. Network dynamics: Propagation and diffusion

**Competencias de la asignatura:****Generales:**

CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

CG12-Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Ser capaz de representar redes sociales en estructuras de datos capaces de hacer procesamientos eficientes. (CG11, CG12, CT1, CT2, CT3)

Conocer los algoritmos fundamentales para el análisis de redes sociales. (CG11, CG12)

Conocer los distintos modelos de redes sociales con sus ventajas e inconvenientes asociados. (CG11, CG12)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**Evaluación detallada:**

Para aprobar la asignatura, el estudiante deberá superar la realización de varias prácticas en grupo y un proyecto final.

La nota final de la asignatura se calculará en base a la siguiente fórmula:  $0.4 \cdot NPO + 0.6 \cdot NPF$  siendo:

- NPO: nota de las prácticas obligatorias realizadas a lo largo del curso.
- NPF: nota del proyecto final (incluye una presentación del mismo).

Para calcular la media es requisito necesario haber entregado y obtenido una calificación mínima (5) en todas las prácticas obligatorias, así como la obtención de una calificación mínima (5) en el proyecto final.

Las calificaciones obtenidas en cualquiera de las pruebas aprobadas se mantendrán para la convocatoria extraordinaria. Tanto las prácticas suspensas como el proyecto podrán entregarse de nuevo para la convocatoria extraordinaria. En caso de entregarse el proyecto, será obligatorio realizar también una presentación del mismo.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- Network Science. Albert-Laszlo Barabasi. <http://networksciencebook.com>
- Networks, Crowds and Markets. David Easley & Jon Kleinberg, <http://www.cs.cornell.edu/home/kleinber/networks-book/>
- Matthew O. Jackson. 2010. Social and Economic Networks. Princeton University Press, Princeton, NJ, USA
- Analyzing the Social Web. Jennifer Golbeck. 2013. Morgan Kaufmann (<https://www.elsevier.com/books/analyzing-the-social-web/golbeck/978-0-12-405531-5>)
- Dynamical Systems on Networks: A Tutorial. Mason A. Porter, James P. Gleeson. 2016. Springer (<https://www.springer.com/us/book/9783319266404>)
- Barabasi Lab: <https://www.barabasilab.com/>
- Los apuntes de la asignatura del curso 2014-2015 están disponibles en <https://github.com/GuilleUCM/SOC>
- Connected: The power of six degrees: <https://www.youtube.com/watch?v=2rzxAyY7D7k>

Ficha docente guardada por última vez el 24/06/2022 12:55:00 por el usuario: Coordinador GIS

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



# UNIVERSIDAD COMPLUTENSE DE MADRID

## FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803365 - Inteligencia Artificial Aplicada al Control <b>Asignatura en Inglés:</b> Artificial Intelligences applied to control systems.	<b>Abrev:</b> IAAC <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de sistemas inteligentes	<b>12 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Percepción computacional	<b>6 ECTS</b>	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Santos Peñas, Matilde

### Descripción de contenidos mínimos:

Aplicación de técnicas de Inteligencia Artificial al control de sistemas. Control Inteligente. Sistemas Expertos. Redes Neuronales. Lógica Fuzzy. Computación evolutiva. Agentes inteligentes

### Programa detallado:

Control de Sistemas  
Control Inteligente  
Sistemas Expertos  
Redes Neuronales  
Lógica Fuzzy  
Algoritmos genéticos  
Aplicaciones

### Programa detallado en inglés:

Automatic control  
Intelligent control  
Expert Systems applied to control  
Neuro-control  
Fuzzy-control  
Genetic algorithms  
Engineering applications of intelligent control

### Competencias de la asignatura:

#### Generales:

- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CG22-Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
- CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

#### Específicas:

No tiene

#### Básicas y Transversales:

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.
- CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

#### Resultados de aprendizaje:

Conocer todo un conjunto de técnicas englobadas dentro del concepto de Inteligencia Artificial y derivar de este conocimiento sus aplicaciones en problemas relacionados con la automática y aplicaciones reales industriales y de ingeniería. (CG11, CG23, CT3)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Profundizar en las técnicas de la inteligencia artificial, en particular las que forman parte del “softcomputing”, y analizar su aplicabilidad en problemas reales. (CG23, CT3)

Destrezas y habilidades en la utilización de las herramientas y lenguajes informáticos requeridos para el desarrollo y aplicación de estas técnicas. (CT3)

Trabajo cooperativo para el desarrollo de habilidades y competencias, incluyendo la capacidad del análisis crítico de las decisiones adoptadas. (CG22, CT1)

Aplicar y experimentar nuevos sistemas de control inteligente de procesos industriales y sistemas ingenieriles. (CG23, CT2, CT3)

Aplicar y experimentar nuevos sistemas de incremento de la automatización y la productividad en la industria. (CG11, CG23, CT2, CT3, CT5)

**Evaluación detallada:**

Se realiza una evaluación basándose en los siguientes parámetros:

- Se evalúa de forma continua - exclusivamente en el laboratorio y durante los horarios asignados a las clases y en los plazos indicados- la realización, individual o en grupos de dos alumnos, de las prácticas propuestas. Para aprobar la asignatura es un requisito imprescindible realizar con una evaluación positiva las prácticas durante las clases en los plazos indicados.

- Es obligatoria la asistencia a clase (10%)

- También se pide la realización individual de unos ejercicios que se entregarán por escrito en el plazo que se indique o de algunos cuestionarios sobre la materia (10%)

- Se realizará la evaluación final mediante un examen escrito o mediante la presentación y memoria de un trabajo, según se solicite (80%).

Se podrá además valorar la exposición de trabajos en clase o la presentación y discusión de artículos científicos, así como otras actividades que puedan sugerirse durante el curso y que se podrán tener en cuenta para mejorar la nota obtenida con los procedimientos anteriores.

**Actividades docentes:**

Reparto de créditos:

Teoría: 2,00

Problemas: 1,00

Laboratorios: 3,00

Otras actividades:

Discusión de artículos científicos en clase.

**Bibliografía:**

- G. Pajares, M. Santos; Inteligencia Artificial e Ingeniería del Conocimiento; RA-MA, 2005
- A. E. Ruano, Intelligent Control using Intelligent Computational Techniques, IEE Control Series, 2005
- Zilouchian, A., Jamshidi, M., Intelligent Control Systems Using Soft Computing Methodologies, CRC Press, 2001
- Hopgood, A. A. (2021). Intelligent systems for engineers and scientists: A practical guide to artificial intelligence. CRC press.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º ( 1C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803366 - Robótica <b>Asignatura en Inglés:</b> Robotics		<b>Abrev:</b> ROB <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de arquitectura y tecnología de computadores		<b>30 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Arquitectura Interna de Linux y Android Cloud y Big Data Diseño automático de sistemas Programación de GPUs y aceleradores		6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> García Perez, Lia	

**Descripción de contenidos mínimos:**

Robótica Industrial  
Robots autónomos y navegación  
Interacción con el entorno

**Programa detallado:**

## Tema 1.- Introducción

Qué es robótica y qué se entiende por robot. Desarrollo histórico de los robots. Análisis de diferentes aplicaciones de robots manipuladores y móviles autónomos y qué tareas pueden observarse en robótica. Por último se estudiará cómo se aplica la inteligencia artificial en robótica.

## Tema 2.- Robótica industrial

Introducción a la robótica industrial y su problemática. Se aprenderá a utilizar las coordenadas homogéneas, que tipos de manipuladores existen y cómo obtener los parámetros necesarios para su modelado (algoritmo de Denavit-Hatenberg). Se trabajará con manipuladores desde el punto de vista cinemático, ya sea directo o inverso. Lenguajes de programación de manipuladores.

## Tema 3.- Detección y percepción

Se estudiarán los diferentes sensores necesarios para navegación de robots tanto de obtención de la posición (internos y externos) como detectores de obstáculos. Se hará mención especial al uso de la visión artificial en robótica.

## Tema 4.- Planificación de trayectorias

Se realizarán consideraciones generales sobre planificación de trayectorias y se mostrará cómo planificar trayectorias en un manipulador para llevar a cabo el movimiento deseado. Así mismo se realizara la planificación de trayectorias en robots móviles autónomos.

## Tema 5.- Robots autónomos y navegación

Revisión de los conceptos más utilizados y relacionados con robots autónomos. Así se estudiarán las diferentes arquitecturas de control y sus modelos cinemáticos (de ruedas independientes, de patas, aéreos, ...). Se estudiarán las particularidades de cada uno de ellos y cómo programarlos para realización de tareas de percepción y navegación.

**Programa detallado en inglés:**

## 1.- Introduction

In this subject, you study what is robotics and what is a robot. A historical review of robots is presented. You will analyse different applications of autonomous mobile robots and manipulators and what tasks can be done with robots. Finally, artificial intelligent is applied to robotics.

## 2.- Industrial robots

Kinematics of robots is the essential aspect of any industrial robot. In order to study the kinematic of manipulators, the student will learn about rotations and translations in 2-D, links and joints and the common types of robots design. The design parameters of a manipulator are obtained using the Denavit-Hatenberg algorithm. Forward and inverse kinematics will be studied. Examples of programming languages are showed.

## 3.- Sensing and Perception

Sensing and perception are to estimate characteristics of the environment, which are required to allow to a robot executed its tasks. So sensors as inertial sensors, GPS, sonar sensing or artificial vision are studied.

## 4.- Path planning

Path-planning is an important task for autonomous mobile robots that lets robots find the shortest – or otherwise optimal – path between two points. In order to plan a path, we need to know two data: a map of the environment and the robot to be aware of its location with respect to the map. So we will study several path planning algorithms when we know the position of robot. Finally, how to deal with uncertain position information is showed.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



5.- Navigation and autonomous robots

In this chapter, the basic tasks necessary to navigate are studied. We will focus the different control architectures of autonomous robots. Besides, the robot kinematic motion models are presented as wheeled robots, legged robots, UAVs, etc.

**Competencias de la asignatura:**

**Generales:**

- CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.
- CG5-Comprensión y dominio de los conceptos básicos de campos y ondas y electromagnetismo, teoría de circuitos eléctricos, circuitos electrónicos, principio físico de los semiconductores y familias lógicas, dispositivos electrónicos y fotónicos, y su aplicación para la resolución de problemas propios de la ingeniería.
- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- CG13-Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- CG19-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.
- CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Conocer los principios básicos de los robots, los distintos tipos de sensores utilizados, las aplicaciones de la robótica y ser capaz de plantear proyectos y soluciones a problemas con robots. (CG5, CG23, CT1, CT2)
- Comprensión y dominio de los conceptos básicos sobre la cinemática de los manipuladores. (CG11, CG13)
- Aplicar conceptos básicos de programación y de sistemas operativos aplicados a la programación de robots. (CG3, CG4, CG19, CT3)

**Evaluación detallada:**

Se utilizará la evaluación continua. La nota total se irá obteniendo a lo largo del curso donde se evaluarán los conocimientos teóricos y prácticos. Se realizarán distintos trabajos para profundizar en los temas propuestos. La asistencia a las sesiones prácticas es obligatoria para la obtención de la nota correspondiente a las prácticas.

La nota se irá obteniendo a lo largo del curso y consta de los siguientes conceptos:

Actividades (50%):

- Trabajos (20 %)
  - Ejercicios prácticos y problemas
  - Trabajos de profundización
  - Participación en clase y/o en el campus virtual
- Prácticas (30 %)

Conocimientos (50%):

- Evaluaciones periódicas (50%) [50% teoría – 50% Problemas]

Será necesario obtener una calificación mínima de un 4/10 en Conocimientos (total de Evaluaciones periódicas) para que se pueda hacer media con las Actividades.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





En la convocatoria extraordinaria se realizará un examen de teoría y problemas (70%) y se conservará la nota de prácticas de la asignatura (30%). No se tendrán en cuenta los trabajos realizados en la evaluación continua ni otras actividades. Será necesario obtener una calificación mínima de un 4/10 en el examen para que se pueda hacer media con la nota de prácticas.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 1,00

Laboratorios: 2,00

Otras actividades:

No tiene

**Bibliografía:**

Bibliografía básica:

Robótica, Control, Detección, Visión e Inteligencia; Fu, K.S., González, R.C. y Lee, C,S,G. Mc Graw-Hill, 1988.

Ollero, A. Robótica, Manipuladores y Robots Móviles. Marcombo, 2002.

Sensors for mobile robots. Theory and application. H.R. Everett. A.K. Peters. Wellesley, 1995.

Introduction to Robotics. P.J. McKerrow. Addison-Wesley, 1991.

Bibliografía complementaria:

Introducción a la robótica. Principios teóricos, construcción y programación de un robot educativo. J.M. Angulo Usategui, S.romero, I. A. Martínez. Ed. Thomson, 2005.

Fundamentos de Robótica A. Barrientos, L.F. Peñin, C. Balaguer, R. Aracil. Mc. Graw-Hill, 1997.

Robots y Sistemas sensoriales. Fernado Torres, Jorge Pomares y otros. Prentice Hall, 2002.

Robot motion planning. J.C. Latombe. Kluwer Academic Plublishers, 1991.

Introductory Computer Vision and Image Processing. A. Low. Mc. Graw-Hill, 1991.

Visión por computador: imágenes digitales y aplicaciones. 2ª edición. G. Pajares y J. M. de la Cruz. RA-MA, 2008.

Ejercicios resueltos de visión por computador. G. Pajares y J. M. de la Cruz. RA-MA, 2007.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803367 - Cloud y Big Data <b>Asignatura en Inglés:</b> Cloud and Big Data		<b>Abrev:</b> CLO <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de arquitectura y tecnología de computadores		<b>30 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Arquitectura Interna de Linux y Android Diseño automático de sistemas Programación de GPUs y aceleradores Robótica		6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Huedo Cuesta, Eduardo	

**Descripción de contenidos mínimos:**

1. Introducción al Cloud Computing
2. Estrategia Migración Cloud
3. Diseño de Arquitecturas Cloud
4. Programación

**Programa detallado:**

1. Ciencia de datos a gran escala
2. Procesamiento a gran escala en el Cloud
3. Fundamentos de procesamiento de datos
4. Procesamiento de datos por lotes con MapReduce
5. Procesamiento de datos por lotes con Spark
6. Procesamiento de datos en flujo con Spark
7. Aspectos prácticos de la computación Cloud
8. Temas avanzados y tendencias actuales

**Programa detallado en inglés:**

1. Large-scale Data Science
2. Large-scale Processing on the Cloud
3. Foundations of Data Processing
4. Batch Data Processing with MapReduce
5. Batch Data Processing with Spark
6. Stream Data Processing with Spark
7. Practical Aspects of Cloud Computing
8. Advanced Topics and Current Trends

**Competencias de la asignatura:****Generales:**

- CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.
- CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
- CG19-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Conocer y practicar los modelos de programación, plataformas, herramientas open-source e infraestructuras relevantes para big data. (CG19)
- Conocer y practicar el modelo de provisión cloud infraestructura como servicio. (CG16)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID  
FACULTAD DE INFORMATICA

Aplicar big data y cloud para resolver un problema de procesamiento de datos a gran escala. (CG18, CT3)

**Evaluación detallada:**

- Evaluación final (en ambas convocatorias):  
Examen teórico (40%).

- Evaluación continua (no recuperable):  
Realización y defensa de un proyecto (30%).  
Prácticas de laboratorio (20%).  
Participación en clase (10%).

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- Designing Data-Intensive Applications. The Big Ideas Behind Reliable, Scalable, and Maintainable Systems; Martin Kleppmann; O'Reilly Media, March 2017
- MapReduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop; Adam Shook and Donald Miner; O'Reilly Media, December 2012
- Data Analysis in the Cloud, Models, Techniques and Applications; Domenico Talia, Paolo Trunfio, Fabrizio Marozzo, September 2015
- Cloud Computing for Science and Engineering; Ian Foster, Dennis Gannon; The MIT Press Cambridge, 2017
- Cloudbonomics: The Business Value of Cloud Computing, Joe Weinman, Wiley, September 2012
- Hadoop MapReduce Cookbook, Srinath Perera and Thilina Gunarathne, Packt Publishing, January 2013
- OpenNebula 3 Cloud Computing, Giovanni Toraldo, Packt Publishing, May 2012

Ficha docente guardada por última vez el 27/06/2023 9:27:00 por el usuario: Coordinador GIC

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 803368 - Bases de Datos noSQL <b>Asignatura en Inglés:</b> NoSQL Databases		<b>Abrev:</b> NSQ <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de gestión y procesamiento de la información		<b>24 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Análisis de redes sociales Gestión de la información en la web Minería de datos y el paradigma Big Data		6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> García Ruiz, Yolanda	

**Descripción de contenidos mínimos:**

Distinción entre base de datos relacional y NoSQL.  
Arquitectura de bases de datos NoSQL.  
Consultas y factores que influyen en el rendimiento.

**Programa detallado:**

Tema 1: Introducción a las bases de datos NoSQL. Arquitectura y clasificación de las bases de datos NoSQL.  
Tema 2: Operaciones de consulta.  
Tema 3: Diseño de una bases de datos NoSQL .  
Tema 4: Índices, eficiencia y estadísticas de rendimiento.  
Tema 5: Sharding y replicación de datos.  
Tema 6: Conceptos avanzados.

**Programa detallado en inglés:**

Lecture 1: Introduction to NoSQL Database. NoSQL database architecture and classification  
Lecture 2: Query operations.  
Lecture 3: NoSQL database design  
Lecture 4: Indices, efficiency and performance measurements.  
Lecture 5: Sharding and replication  
Lecture 6: Advanced concepts

**Competencias de la asignatura:****Generales:**

CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.  
CG17-Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.

**Específicas:**

CE\_C7-Capacidad para conocer y desarrollar técnicas de aprendizaje computacional y diseñar e implementar aplicaciones y sistemas que las utilicen, incluyendo las dedicadas a extracción automática de información y conocimiento a partir de grandes volúmenes de datos.  
CE\_TI5-Capacidad para seleccionar, desplegar, integrar y gestionar sistemas de información que satisfagan las necesidades de la organización, con los criterios de coste y calidad identificados.

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.  
CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Evaluar, proponer y presentar adecuadamente soluciones eficientes de almacenamiento, en particular discerniendo si se debe emplear una base de datos relacional o NoSQL. (CG3, CG17, CT1, CT3, CE\_C7, CE\_TI5)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Conocer las diversas bases de datos NoSQL, y sus características. (CG3, CE\_C7, CE\_TI5)

Ser capaz de efectuar consultas sobre bases de datos NoSQL para obtener información relevante. (CG3, CE\_TI5)

**Evaluación detallada:**

- 50%: realización de trabajo práctico. El trabajo práctico estará organizado en varias entregas, para cada una de las cuales puede pedirse la presencia en clase. También pueden incluirse tests de repaso de la materia vista hasta el momento.
- 50%: realización de una prueba oral o escrita.

En la convocatoria extraordinaria se propondrá un trabajo práctico diferente, pero el esquema de evaluación es el mismo, manteniendo la realización de una prueba oral o escrita.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Actividades presenciales: 65% de dedicación. 2 horas de clase teórica y 2 de laboratorio por semana.

Actividades dirigidas: 15%

Trabajo personal: 20% : estudio, preparación de presentaciones

**Bibliografía:**

Libros:

- Kristina Chodorow & Mike Dirolf: MongoDB: The Definitive Guide (2013). O'Reilly Media

- Introducción a las bases de datos. Nosql usando MongoDB. Antonio Sarasa. Editorial UOC, S.L.; Edición: 1 (17 de mayo de 2016). ISBN. 978-8491162667

- Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. Eric Redmond Jim y R. Wilson. Pragmatic Bookshelf; Edición: 1 (11 de mayo de 2012).

- Michael Framptom: Big Data Made Easy (2014). Apress.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:

**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 803369 - Herramientas informáticas para los juegos de azar <b>Asignatura en Inglés:</b> Software tools to support online gambling		<b>Abrev:</b> HJA <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos científico-matemáticos		<b>18 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Calculabilidad y Complejidad Los escenarios científicos y tecnológicos emergentes y la defensa		6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Núñez Covarrubias, Alberto	

**Descripción de contenidos mínimos:**

Conceptos básicos relativos a juegos de azar y su relación con la Informática.  
Conceptos básicos de NLHE.  
Desarrollo de herramientas para resolver problemas sencillos en el ámbito de NLH

**Programa detallado:**

Durante los últimos años se ha generalizado el uso de herramientas informáticas para mejorar el rendimiento en juegos de azar online. De hecho, el acceso a herramientas con buenas funcionalidades, o mejor aún, a herramientas de las que otros jugadores no disponen, es fundamental para obtener buenos resultados. Una segunda vertiente en el uso de herramientas consiste en producir herramientas que nos permitan validar, mediante el uso de simulación, la bondad de diferentes estrategias en juegos de azar.

**Objetivos de la asignatura**

Los principales objetivos de esta asignatura son, por un lado, que los alumnos adquieran los conocimientos básicos sobre uso de herramientas informáticas para juegos de azar y, por otro lado, que desarrollen pequeñas herramientas que aporten funcionalidades que no están cubiertas de forma adecuada por las herramientas existentes.

1. Introducción a juegos de azar online: poker, Black-Jack y ruleta.
2. Introducción al No-limit Hold'em (preflop).
  - 2.1 Uso de herramientas, análisis de resultados y desarrollo de nuevas herramientas (preflop).
3. Introducción al No-limit Hold'em (postflop).
  - 3.1 Uso de herramientas, análisis de resultados y desarrollo de nuevas herramientas (postflop).

**Programa detallado en inglés:**

During the last years there has been a widespread use of software tools to improve the performance while gambling. Actually, the access to good tools, even better to tools that are not widely available, is a must to obtain good results. A second variant consists in developing tools that can be used to validate, by using simulation techniques, how good a certain strategy is.

**Goals of the course**

The main goals of the course are: the students should learn how to use different tools for gambling and they should be able to develop small tools to cover functionalities that are not supported by current tools.

1. Introduction to online gambling: poker, Black-Jack and roulette.
2. Introduction to No-limit Hold'em (preflop).
  - 2.1 Use of tools, analysis of results and development of new tools (preflop).
3. Introduction to No-limit Hold'em (postflop).
  - 3.1 Use of tools, analysis of results and development of new tools (postflop).

**Competencias de la asignatura:****Generales:**

CG8-Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.

CG13-Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Desarrollo de herramientas, usando técnicas y metodologías de Ingeniería del Software, para ayudar en la toma de decisiones en NLHE. (CG8, CG13)

Desarrollo de proyectos software de complejidad media en grupo. (CG8, CT1)

Búsqueda y uso adecuado de información complementaria para afrontar el desarrollo de herramientas para NLHE. (CT3)

**Evaluación detallada:**

Prácticas realizadas por grupos de 4 alumnos, con plazo de entrega (dentro del periodo de impartición de la asignatura, se fijará con suficiente antelación y se anunciará debidamente en clase y a través del campus virtual), que se calificarán entre 0 y 10 puntos teniendo en cuenta no solo la corrección del programa presentado sino su facilidad de uso, el interfaz y las características adicionales que no se exigían en el enunciado de cada práctica. La evaluación de las prácticas se realizará en el laboratorio, el día fijado para su finalización. La asistencia a los laboratorios (podrá ser "virtual" en situaciones de fuerza mayor como una pandemia) en los días de entrega de prácticas es obligatoria (obteniendo una calificación de 0 puntos aquellos alumnos ausentes y que no puedan justificar documentalmente dicha ausencia).

Se valorará la participación en las clases teóricas, respondiendo a preguntas o ejercicios planteados por el profesor.

Convocatoria ordinaria: Prácticas 90%; Participación en clase 10%.

Convocatoria extraordinaria: Examen 50%; Prácticas 40%; Participación en clase 10%.

Durante el desarrollo de la asignatura se entregarán tres prácticas obligatorias y una opcional. La primera práctica tendrá un peso del 15%, la segunda del 35% y la tercera del 20%, con respecto a la nota final. La práctica opcional tendrá un peso del 20%. Las prácticas no presentadas en la fecha fijada tendrán una puntuación de 0, tanto en convocatoria ordinaria como extraordinaria. En el caso de la convocatoria extraordinaria, se mantiene la nota que el alumno hubiera obtenido en la evaluación de las prácticas durante el periodo de impartición de la asignatura. Para aprobar la asignatura es necesario obtener, al menos, un 5 sobre 10 en la nota final.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

1. Thomas Bakker. Analytical No-limit Hold'em. Two plus Two Publishing LLC, 2010.
2. Thomas A. Bass. The Newtonian Casino. Penguin Books Ltd, 1991.
3. Will Ma. Curso 15S50 impartido en MIT. <http://web.mit.edu/willma/www/2013mit15s50.html> , 2013.
4. Collin Moshman and Douglas Zare. The Math of Hold'em. Dimat Enterprises Inc, 2011.
5. Dario de Toffoli and Margherita Bonaldi. Black Jack: A Champion's Guide. Gaming Books, 2013.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID****FACULTAD DE INFORMATICA**

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 804997 - Introducción a la Tecnología Blockchain y Smart Contracts <b>Asignatura en Inglés:</b> Introduction to Blockchain Technology and Smart Contracts		<b>Abrev:</b> TBC <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de software de sistemas		36 ECTS	
<b>Otras asignaturas en la misma materia:</b> Desarrollo de videojuegos mediante tecnologías web Ingeniería de Comportamientos Inteligentes Programación Competitiva Programación de aplicaciones para dispositivos móviles Testing de Software		6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Gordillo Alguacil, Pablo	

<b>Descripción de contenidos mínimos:</b> -Fundamentos de la tecnología Blockchain -Programación de smart contracts -Aplicaciones
<b>Programa detallado:</b> 1. Introducción a los sistemas descentralizados. 2. Mecánica de un sistema de blockchain. 3. Smart contracts: un blockchain programable. 4. El lenguaje de programación Solidity. 5. Seguridad de contratos inteligentes. Análisis de vulnerabilidades. 6. Aplicaciones y casos de uso.
<b>Programa detallado en inglés:</b> 1. Introduction to decentralized systems. 2. Mechanics of a blockchain system. 3. Smart contracts: a programmable blockchain. 4. The Solidity programming language. 5. Smart Contract Safety. Analysis of vulnerabilities. 6. Applications and case studies.
<b>Competencias de la asignatura:</b>
<b>Generales:</b> CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.  CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
<b>Específicas:</b> No tiene
<b>Básicas y Transversales:</b> CT2-Capacidad de análisis y síntesis en la resolución de problemas.
<b>Resultados de aprendizaje:</b> Conocer los conceptos básicos de la tecnología Blockchain y las aplicaciones basadas en smart contracts (CG11, CG18)  Diseñar e implementar aplicaciones sencillas basadas en smart contracts (CG11, CT2)  Realizar tareas básicas sobre sistemas de blockchain (CG11, CG18)  Conocer los conceptos básicos de seguridad de las aplicaciones basadas en smart contracts e identificar las vulnerabilidades conocidas más importantes (CG11, CT2)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID  
FACULTAD DE INFORMATICA

**Evaluación detallada:**

- Convocatoria ordinaria:

- Realización de trabajos prácticos: 60%
- Participación y realización de ejercicios en clase: 20%
- Realización de un examen: 20%

- Convocatoria extraordinaria:

- Realización de un trabajo práctico: 50%
- Realización de un examen: 50%

**Actividades docentes:**

Reparto de créditos:

- Teoría: 3,00
- Problemas: 0,00
- Laboratorios: 3,00

Otras actividades:

Clases teóricas de exposición de los principales conceptos teóricos de la asignatura y resolución de casos prácticos. Clases prácticas y de Laboratorio, incluyendo trabajos tanto individuales como en equipo.

**Bibliografía:**

- S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system (2008).
- Ethereum White Paper. <https://ethereum.org/en/whitepaper/>
- Solidity documentation. <https://solidity.readthedocs.io>
- G. Wood, Ethereum: A secure decentralised generalised transaction ledger (2014). <https://gavwood.com/paper.pdf>
- C. Dannen, Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners. Apress, 2017.
- W.-M. Lee, Beginning Ethereum Smart Contract Programming. Apress, 2019.
- M. Mukhopadhyay, Ethereum Smart Contract Development. Packt publishing, 2018.
- A. Antonopoulos, Mastering Ethereum: Implementing Digital Contracts, O'Reilly, 2018.
- A. Antonopoulos, Mastering Bitcoin: Programming the Open Blockchain. O'Reilly, 2017.

Ficha docente guardada por última vez el 21/06/2023 10:59:00 por el departamento: **Sistemas Informáticos y Computación**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 804998 - Computación Cuántica <b>Asignatura en Inglés:</b> Quantum Computing	<b>Abrev:</b> CC <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos científico-matemáticos	<b>12 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Los escenarios científicos y tecnológicos emergentes y la defensa	<b>6 ECTS</b>	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Análisis Matemático y Matemática Aplicada		<b>Coordinador:</b> Iblisdir , Sofyan

**Descripción de contenidos mínimos:**

- 1.- Postulados de la mecánica cuántica
- 2.- Circuitos cuánticos. Transformada Cuántica de Fourier
- 3.- Algoritmos de Shor y Grover

**Programa detallado:**

La computación cuántica se está consolidando como una de las tecnologías clave del futuro inmediato. Todas las grandes empresas tecnológicas están creando y agrandando sus divisiones cuánticas. Igualmente, todos los países desarrollados están invirtiendo fuertemente en el área.

Es por tanto este un momento crucial para aprender las bases de estas tecnologías emergentes y poder así liderar proyectos de futuro en las mismas.

**OBJETIVOS:**

El objetivo final de esta asignatura es que los alumnos adquieran los conocimientos necesarios para poder diseñar algoritmos cuánticos por ellos mismos. Por ello el curso tendrá un carácter teórico-práctico.

**TEMARIO**

- 1.- Álgebra lineal
  - 1.1 Notación de Dirac
  - 1.2 Repaso de álgebra lineal en notación de Dirac
- 2.- Qubits, puertas y circuitos cuánticos
  - 2.1 Postulados de la mecánica cuántica
  - 2.2 Definición de puerta y circuito cuántico
  - 2.3 Representación gráfica
  - 2.4 Ejemplos
  - 2.5 Transformada cuántica de Fourier
- 3.- Algoritmos cuánticos
  - 3.1 Puertas universales
  - 3.2 Complejidad
  - 3.3 Principales ejemplos de algoritmos cuánticos
  - 3.4 RSA y factorización
  - 3.5 Algoritmo de Shor
- 4.- Códigos cuánticos correctores de error
  - 4.1 Modelos de ruido
  - 4.2 Recordatorio de códigos correctores de error clásicos
  - 4.3 Códigos de Shor y de Steane
  - 4.4 Códigos cuánticos derivados de estabilizadores
  - 4.5 Computación cuántica robusta

**Programa detallado en inglés:**

Quantum computing is becoming a key technology for the near future. All major Tech-Companies are creating and expanding their quantum divisions. Similarly, all developed countries are investing strongly in the area.

Therefore, it is time to learn the basis of these technologies to be in the position to lead future projects around them.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**OBJECTIVES:**

The main goal is to acquire the necessary knowledge to design quantum algorithms.

**TABLE OF CONTENTS**

- 1.- Linear algebra
  - 1.1 Dirac's Notation
  - 1.2 Basics of linear algebra using Dirac's notation
- 2.- Qubits, gates and quantum circuits
  - 2.1 Postulates of quantum mechanics
  - 2.2 Definition of gate and quantum circuit
  - 2.3 Graphical representation
  - 2.4 Examples
  - 2.5 Fourier quantum transform
- 3.- Quantum algorithms
  - 3.1 Universal gate sets
  - 3.2 Complexity
  - 3.3 First examples of quantum algorithms
  - 3.4 RSA and factorization
  - 3.5 Shor's algorithm
- 4.- Quantum error correcting codes
  - 4.1 Noise models
  - 4.2 Reminder on classical error correcting codes
  - 4.3 Shor code and Steane code
  - 4.4 Quantum code from stabilisers
  - 4.5 Fault-tolerant quantum computation

**Competencias de la asignatura:**

**Generales:**

- CG1-Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.
- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Comprender los fundamentos de la computación cuántica y sus principales algoritmos (CG1, CG11)
- Desarrollar algoritmos cuánticos nuevos, o variantes de los existentes (CG1, CG11, CT1, CT2, CT3)

**Evaluación detallada:**

Pruebas parciales: 40% de la nota global  
Examen final: 60% de la nota global

**Actividades docentes:**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**

**FACULTAD DE INFORMATICA**

Reparto de créditos:

Teoría: 4,00

Problemas: 2,00

Laboratorios: 0,00

Otras actividades:

No tiene

**Bibliografía:**

- 1.- Michael. A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2000.
- 2.- Ronald de Wolf, Quantum Computing: Lecture Notes, arXiv:1907.09415

Ficha docente guardada por última vez el 30/06/2023 15:21:00 por el usuario: Coordinador GII

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º (2C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 804999 - Arquitecturas y Programación de Computadores Cuánticos <b>Asignatura en Inglés:</b> Architecture and Programming of Quantum Computers		<b>Abrev:</b> APCC <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de arquitectura y tecnología de computadores		<b>36 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Arquitectura Interna de Linux y Android Cloud y Big Data Diseño automático de sistemas Programación de GPUs y aceleradores Robótica		6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Botella Juan, Guillermo	

**Descripción de contenidos mínimos:**

1. Principios básicos de Computación Cuántica: principios matemáticos, físicos y tecnológicos. Computación reversible
2. Introducción a los circuitos cuánticos: puertas cuánticas, QFT, métricas y computadores cuánticos reales
3. Plataformas para desarrollo y simulación de circuitos cuánticos: Qiskit, Forest, MQDK, Cirq, ProjectQ
4. Problemas cuánticos clásicos: Deutsch-Jozsa, Bernstein-Vazirani, Grover, Shor
5. Simuladores de computadores cuánticos. HPC y Computación Cuántica

**Programa detallado:**

1. Principios básicos de Computación Cuántica
  - 1.1. Fundamentos matemáticos, físicos y tecnológicos
  - 1.2. Computación reversible
2. Introducción a los circuitos cuánticos
  - 2.1. Puertas cuánticas básicas
  - 2.2. Transformada Cuántica de Fourier (QFT)
  - 2.3. Métricas: puertas, profundidad, ancho, precisión
  - 2.4. Computadores cuánticos reales. Limitaciones
3. Plataformas para desarrollo y simulación de algoritmos cuánticos
  - 3.1. Qiskit de IBM
  - 3.2. Forest de Rigetti Computing. Compilación en Quil
  - 3.3. Quantum Development Kit de Microsoft. Ejemplos en Q#
  - 3.4. Cirq de Google
  - 3.5. ProjectQ de ETH Zurich. FermiLib
4. Problemas cuánticos clásicos
  - 4.1. Deutsch-Jozsa
  - 4.2. Bernstein-Vazirani
  - 4.3. Grover
  - 4.4. Shor
5. Simuladores de computadores cuánticos. HPC y Computación Cuántica
  - 5.1. QPU vs HPC
  - 5.2. Ejemplos de sistemas construidos

**Prácticas de laboratorio:**

- Puertas cuánticas y QFT
- Algoritmo de Deutsch-Jozsa
- Algoritmo de Bernstein-Vazirani
- Sumadores cuánticos: Vedral, Barenco y Eckert
- Sumadores cuánticos: Draper y Cuccaro
- Algoritmo de Grover

**Programa detallado en inglés:**

1. Basic principles of Quantum Computing
  - 1.1. Mathematical, physical and technologic fundamentals

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID  
FACULTAD DE INFORMATICA

- 1.2. Reversible computing
2. Introduction to quantum circuits
  - 2.1. Basic quantum gates
  - 2.2. Quantum Fourier Transform (QFT)
  - 2.3. Metrics: gates, depth, width and precision
  - 2.4. Real quantum computers. Limitations.
3. Frameworks for developing and simulating quantum algorithms
  - 3.1. IBM Qiskit
  - 3.2. Rigetti Forest. PyQuil
  - 3.3. Microsoft Quantum Development Kit. Q#
  - 3.4. Google Cirq
  - 3.5. ETH Zurich ProjectQ. FermiLib
4. Classical quantum problems
  - 4.1. Deutsch-Jozsa
  - 4.2. Bernstein-Vazirani
  - 4.3. Grover
  - 4.4. Shor
5. Quantum computer simulators. HPC and Quantum Computing
  - 5.1. QPU vs HPC
  - 5.2. Instances of constructed systems

Lab assignments:

- Quantum gates and QFT
- Deutsch-Jozsa algorithm
- Bernstein-Vazirani algorithm
- Quantum adders: Vedral, Barenco and Eckert
- Quantum adders: Draper and Cuccaro
- Grover's algorithm

**Competencias de la asignatura:**

**Generales:**

- CG13-Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- CG14-Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.
- CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

- Comprender los principios y bloques elementales de la computación cuántica. (CG14)
- Comprender el paralelismo inherente a la computación cuántica. (CG14)
- Analizar con sentido crítico la complejidad de un circuito cuántico. (CG13)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMÁTICA**

- Conocer distintos frameworks de programación de circuitos cuánticos. (CG13)
- Conocer las herramientas de simulación, así como los modelos de ruido de las puertas cuánticas. (CG13)
- Conocer los computadores cuánticos reales, así como su topología y sus características físicas. (CG18)
- Conocer la arquitectura de los sistemas simuladores de computadores cuánticos. (CG14)
- Realización de prácticas y proyectos sobre circuitos cuánticos. (CT1, CT2, CT3)

**Evaluación detallada:**

Dos modalidades a elegir: evaluación continua o práctica especial

Modalidad de evaluación continua (aplicable solo a la convocatoria ordinaria):

Asistencia al laboratorio y realización de prácticas propuestas durante el semestre = 30% Trabajos final = 40%

Examen = 30%

Modalidad de práctica especial (aplicable a las dos convocatorias de la asignatura):

Práctica especial en laboratorio (se realizarán diferentes ejercicios teórico-prácticos en el laboratorio el día asignado por el profesor) = 100%

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

Enseñanza presencial teórica. Realización de prácticas de laboratorio.

**Bibliografía:**

Bibliografía básica:

Eric R. Johnston, Nic Harrigan, Mercedes Gimeno-Segovia, Programming Quantum Computers, 1st edition, O'Reilly, 2019.

Jack D. Hidary, Quantum Computing: An Applied Approach, 1st edition, Springer, 2019.

Learn Quantum Computation using Qiskit. 2020. <http://community.qiskit.org/textbook>.

Bibliografía complementaria:

Robert S. Sutor, Dancing with Qubits, 1st edition, Packt Publishing, 2019.

Wolfgang Scherer, Mathematics of Quantum Computing, 1st edition, Springer, 2019.

Ficha docente guardada por última vez el 23/07/2020 9:22:00 por el usuario: Coordinador GIC

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 805371 - Programación Competitiva <b>Asignatura en Inglés:</b> Competitive Programming	<b>Abrev:</b> PCOM <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de software de sistemas	<b>36 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Desarrollo de videojuegos mediante tecnologías web Ingeniería de Comportamientos Inteligentes Juegos Serios Programación de aplicaciones para dispositivos móviles Testing de Software	6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Interdepartamental ISIA / SIC <b>Coordinador:</b> Gómez Martín, Marco Antonio		

**Descripción de contenidos mínimos:**

Introducción a la programación competitiva  
Jueces de programación competitiva y sus veredictos  
Concursos existentes. Tipos y características  
Esquemas típicos de entradas de datos  
Esquemas de soluciones en C, C++ y Java  
Estrategias para la participación en concursos  
Uso de técnicas algorítmicas básicas en programación competitiva: estructuras de datos, vuelta atrás, divide y vencerás, algoritmos voraces, programación dinámica y recorridos sobre grafos (DFS, BFS, Floyd, Dijkstra)  
Emparejamiento sobre grafos  
Network flows  
Problemas matemáticos en la programación competitiva  
Procesamiento de cadenas  
Geometría computacional  
Combinación de técnicas distintas en la resolución de problemas  
Creación de problemas para concursos de programación

**Programa detallado:**

- 1.- Introducción  
Introducción a la programación competitiva o programación rápida. Jueces automáticos y sus veredictos. Ejemplos de jueces: DOMjudge, Acepta el Reto, UVa online judge, Codeforces, CodeChef, topcoder.
- 2.- Tipos de problemas de la programación competitiva  
Tipos de problemas y su esquema de entrada. Esquema de soluciones en C, C++ y Java. Pros y contras de los distintos lenguajes de programación para su uso en concursos. Entrada/salida eficiente.
- 3.- Algoritmia básica enfocada a programación competitiva  
Estructuras de datos. Vuelta atrás. Divide y vencerás. Algoritmos voraces. Programación dinámica. DFS. BFS. Floyd. Dijkstra.
- 4.- Algoritmos avanzados sobre grafos  
Emparejamiento sobre grafos. Network flows
- 5.- Estructuras de datos avanzadas  
Árboles de intervalos. Árboles de Fenwick
- 6.- Problemas matemáticos en programación competitiva  
Primalidad y factorización. Sucesiones. Búsqueda de ciclos
- 7.- Geometría computacional en programación competitiva  
Líneas. Segmentos. Convex Hull
- 8.- Procesamiento de cadenas  
KMP. Tries. Suffix array.
- 9.- Conceptos avanzados  
Combinación de varias técnicas para resolver problemas complicados
- 10.- Elaboración de problemas para concursos de programación

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Creación de problemas para concursos de programación. Tamaños de la entrada. Generación de casos de prueba. Discriminación de soluciones poco eficientes.

11.- Concursos existentes  
Tipos de concursos. Normas. Rankings.

La mayoría de los contenidos mínimos de la asignatura pueden asociarse a un tema concreto del programa detallado. Algunos, no obstante, son transversales y aparecen durante toda la asignatura (en concreto, las estrategias para la participación en concursos). Además, las técnicas algorítmicas se enseñarán directamente de forma práctica sobre problemas de concursos anteriores (como SWERC, AdaByron, ProgramaMe o Las 12 UVas). Por último, los contenidos del tema 9 también se abordan de forma continua dentro del resto de temas.

**Programa detallado en inglés:**

1.- Introduction to competitive programming

Competitive programming introduction. Automatic online judges and their verdicts. Introduction to different jury systems: DOMjudge, Acepta el Reto, UVa online judge, Codeforces, CodeChef, topcoder.

2.- Competitive programming problems

Anatomy of a programming contest problem: typical input routines. Advantages and drawbacks of programming languages regarding competitive coding. Fast input/output.

3.- Basic algorithms for competitive programming

Data structures. Backtracking. Divide and conquer. Greedy. Dynamic programming. DFS. BFS. Floyd. Dijkstra.

4.- Graphs

Graph matching. Network flows

5.- Advanced data structures

Interval trees. Fenwick trees

6.- Mathematics problems

Prime numbers. Series. Cycle finding.

7.- Computational geometry

Lines. Segments. Convex Hull

8.- String processing

KMP. Tries. Suffix array.

9.- More advanced topics

Complex problems that require more than one algorithm technique and/or data structure.

10.- Being a problem setter

Creating programming contest problems. Input sizes. Test case generation. Discriminating inefficient solutions

11.- Programming contests

Type of contest rules. Rankings.

**Competencias de la asignatura:**

**Generales:**

CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

CG12-Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.

**Resultados de aprendizaje:**

Conocer los jueces de programación. (CT1, CT2, CT3)

Conocer los tipos de concursos de programación competitiva. (CT1, CT2, CT3)

Programar soluciones ganadoras utilizando estructuras de datos y algoritmos para problemas de programación competitiva. (CG11, CG12, CT1, CT2, CT3)

Trabajar en equipo para resolver problemas algorítmicos. (CG11, CG12, CT1, CT2, CT3, CT4)

**Evaluación detallada:**

La evaluación es continua y la nota final en la convocatoria ordinaria se obtendrá con una suma ponderada de las actividades realizadas durante el periodo de clases:

- Entregas y defensa en clase de trabajos relacionados con los contenidos de la asignatura (40%). Algunas de estas entregas y defensas podrán realizarse en grupo.
- Pruebas de evaluación desarrolladas en el laboratorio (40%). Podrán tener la forma de cuestionarios o de concursos de programación.
- Elaboración de problemas para concursos (20%).

La asistencia a clase es obligatoria; se requiere un 70% de asistencia a las clases presenciales (o las correspondientes clases a distancia si la situación sanitaria lo requiere). El estudiante que no cumpla este requisito será calificado con NP en ambas convocatorias, con independencia de la nota que obtenga en el resto de actividades evaluables de la asignatura.

Para aquellos que requieran hacer uso de la convocatoria extraordinaria, se abrirá un nuevo plazo para la elaboración de problemas para concursos (20%) y se realizará un examen en laboratorio para sustituir el 40% de la nota de las pruebas de evaluación. El 40% de la nota proveniente del primer tipo de actividades (entregas y defensa en clase de trabajos) no es recuperable.

**Actividades docentes:**

Reparto de créditos:

Teoría: 0,00

Problemas: 3,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- [1] "Competitive Programming 3: The new lower bound of programming contests". Steven Halim y Felix Halim. Ed. Lulu, 2013
- [2] "Guide to competitive programming. Learning and improving algorithms through contests". Antti Laaksonen. Springer, 2017
- [3] "Programming Challenges: The Programming Contest Training Manual". Steven S. Skiena, Miguel A. Revilla. Springer, 2003

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES		<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 805373 - Ingeniería de Comportamientos Inteligentes <b>Asignatura en Inglés:</b> Intelligent Behaviours Engineering		<b>Abrev:</b> ICI <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de software de sistemas		<b>36 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Desarrollo de videojuegos mediante tecnologías web Juegos Serios Programación Competitiva Programación de aplicaciones para dispositivos móviles Testing de Software		6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo			
<b>Departamento:</b> Ingeniería del Software e Inteligencia Artificial		<b>Coordinador:</b> Recio García, Juan Antonio	

**Descripción de contenidos mínimos:**

- Aplicación de técnicas simbólicas de Inteligencia Artificial y para la creación de comportamientos inteligentes en el dominio de videojuegos.
- Adquisición, representación y uso de conocimiento.
- Sistemas de recomendación.

**Programa detallado:**

- Tema 1. Introducción: Ingeniería del Conocimiento. Modelos de representación del conocimiento: Modelos lógicos, modelos estructurados, modelos imprecisos. Adquisición y reusabilidad del conocimiento.
- Tema 2. Repaso de conceptos de IA para videojuegos. Búsqueda de caminos y movimiento. Diseño de comportamientos y toma de decisiones. Máquinas de estados.
- Tema 3. Sistemas de reglas. Motores de inferencia. Razonamiento encadenado. Aplicación práctica.
- Tema 4. Razonamiento borroso. Introducción y fundamentos. Razonamiento en lógica borrosa. Inferencia. Aplicación práctica.
- Tema 5. Sistemas de razonamiento basado en casos. Aprendizaje y memoria. Aplicaciones a distintos tipos de videojuegos.

**Programa detallado en inglés:**

1. Introduction. Knowledge Engineering, Knowledge representation models: logical models, structured models, fuzzy models. Acquisition and reusability of knowledge.
2. Review of AI concepts for videogames. Pathfinding, behavior design and decision making. State Machines.
3. Rule-based systems. Inference Engine. Chain reasoning. Practical applications.
4. Fuzzy logic. Introduction and fundamentals. Reasoning with Fuzzy Logic, Inference. Practical applications.
5. Case-Based Reasoning. Learning and memory. Applications to videogames.

**Competencias de la asignatura:****Generales:**

CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.
- CT2-Capacidad de análisis y síntesis en la resolución de problemas.

**Resultados de aprendizaje:**

- Escribir programas eficientes, correctos y mantenibles que se encarguen de controlar el comportamiento de los personajes de un videojuego.
- Utilizar perfiles de usuario para la asignación de usuarios a equipos en videojuegos multijugador.
- Identificar y aplicar el paradigma de IA más adecuado para implementar comportamientos inteligentes en videojuegos.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMÁTICA**

**Evaluación detallada:**

La asistencia a clase es obligatoria y evaluable. Se requiere un 80% de asistencia a las clases presenciales para poder aprobar la asignatura.

**Convocatoria ordinaria:**

Evaluación continua mediante la realización y defensa de prácticas realizadas en grupo: 100%.

La nota individual de cada estudiante tendrá en cuenta sus contribuciones individuales al proyecto y su capacidad de trabajo en grupo. El estudiante que no contribuya a la realización de las prácticas grupales podrá ser expulsado del grupo asignado, teniendo que realizarlas de forma individual.

La nota final se calculará como la media de las calificaciones obtenidas en las prácticas.

Aquellos que no hayan cumplido con la presencialidad deberán entregar y aprobar todas las prácticas de forma individual además de superar un examen teórico-práctico.

**Convocatoria extraordinaria:**

Aquellos estudiantes que requieran hacer uso de la convocatoria extraordinaria deberán entregar individualmente una serie de prácticas extraordinarias fijadas por el profesor. Estas prácticas serán modificaciones y/o ampliaciones de las prácticas ordinarias. En el enunciado de las prácticas extraordinarias se indicarán los criterios para superar la convocatoria.

Aquellos que no hayan cumplido con la presencialidad deberán superar además un examen teórico-práctico.

Cualquier detalle adicional sobre la evaluación de la asignatura no contemplado en esta descripción se presentará el primer día de clase, haciéndose público a través del campus virtual.

Todas las prácticas de la asignatura se realizarán en el entorno java MsPacMan Vs. Ghosts, disponible en <https://gaia.fdi.ucm.es/research/mspacman/>.

**Actividades docentes:**

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- S. Russell y P. Norvig: Artificial Intelligence: A modern approach. Prentice Hall, 2020, cuarta edición.
- A. J. González y D. D. Dankel: The Engineering of Knowledge Based Systems. Theory and Practice; Prentice Hall, 1993
- J. Funge: Artificial Intelligence for Computer Games: An Introduction. A K Peters, Wellesley, MA. July 2004.
- E. Friedman-Hill: Jess in Action: Rule-based Systems in Java. Manning 2003.
- D. Manrique Gamio y M.C Suárez de Figueroa Baonza: Razonamiento con imprecisión: lógica borrosa: Apuntes y ejercicios. Universidad Politécnica de Madrid. 2017
- M. Richter y R. Weber: Case-Based Reasoning: A Textbook. Springer 2013

Ficha docente guardada por última vez el 27/06/2023 21:02:00 por el usuario: Vicedecanato de Estudios y Calidad

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:





UNIVERSIDAD COMPLUTENSE DE MADRID  
FACULTAD DE INFORMATICA

Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º (1C)	<b>Idioma:</b> Español
<b>Asignatura:</b> 805756 - Testing de Software <b>Asignatura en Inglés:</b> Software Testing	<b>Abrev:</b> TSW <b>Carácter:</b> Optativa	<b>6 ECTS</b>
<b>Materia:</b> Complementos de software de sistemas	<b>36 ECTS</b>	
<b>Otras asignaturas en la misma materia:</b> Desarrollo de videojuegos mediante tecnologías web Ingeniería de Comportamientos Inteligentes Juegos Serios Programación Competitiva Programación de aplicaciones para dispositivos móviles	6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> García Merayo, M <sup>a</sup> . de las Mercedes

**Descripción de contenidos mínimos:**

Conceptos básicos en testing de software.  
Testing unitario.  
Diferentes criterios de cobertura y comparación y relación entre ellos.

**Programa detallado:**

1. Introducción al testing de software.
2. Testing unitario y automatización del proceso de testing.
3. Criterios de cobertura.
4. Introducción al testing basado en modelos.
5. Otras técnicas de testing: mutaciones, metamórfico, etc.

**Programa detallado en inglés:**

1. Introduction to software testing.
2. Unit testing and automatization of the testing process.
3. Coverage criteria.
4. Introduction to model-based testing.
5. Other testing techniques: mutation, metamorphic, etc.

**Competencias de la asignatura:**

**Generales:**

CG8-Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.  
CG13-Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

**Específicas:**

No tiene

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

**Resultados de aprendizaje:**

Testeo de unidades de complejidad media. (CG13)  
Resolución de problemas prácticos tanto de forma individual como en grupo. (CT1)  
Elección del método más adecuado para afrontar el testing de un sistema específico, incrementando su fiabilidad. (CG8, CG13)

**Evaluación detallada:**

Prácticas realizadas por grupos de alumnos, con plazo de entrega (dentro del periodo de impartición de la asignatura, se fijará con suficiente antelación y se anunciará debidamente en clase y a través del campus virtual), que se calificarán entre 0 y 10 puntos. La evaluación de las prácticas se realizará en el laboratorio, el día fijado para su finalización. La asistencia a los laboratorios en los días de entrega de prácticas es obligatoria (obteniendo una calificación de 0 puntos aquellos alumnos ausentes y que no puedan justificar documentalmente dicha ausencia).

Se valorará la participación en las clases teóricas, respondiendo a preguntas o ejercicios planteados por el profesor.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



# UNIVERSIDAD COMPLUTENSE DE MADRID

## FACULTAD DE INFORMATICA

Convocatoria ordinaria: Prácticas y otras actividades evaluables 90%; Participación en clase 10%.

Convocatoria extraordinaria: Examen 50%; Prácticas y otras actividades evaluables 40%; Participación en clase 10%.

El peso de cada práctica en la nota final de la convocatoria ordinaria, que será una media ponderada de las notas obtenidas en cada práctica, dependerá de su complejidad y se anunciará con la debida antelación. Las prácticas no presentadas en la fecha fijada tendrán una puntuación de 0 tanto en convocatoria ordinaria como extraordinaria. Las actividades evaluables se anunciarán al inicio del curso. En el caso de la convocatoria extraordinaria, se mantiene la nota que el alumno hubiera obtenido en la evaluación de las prácticas y actividades evaluables durante el periodo de impartición de la asignatura.

### Actividades docentes:

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

### Bibliografía:

Básica

P. Ammann and J. Offutt. Introduction to Software Testing (2nd edition). Cambridge University Press, 2016.

Complementaria

R. Biering, S. Brown, E. Galván, J. Timoney. Essentials of Software Testing. Cambridge University Press, 2022.

R. M. Hierons, J. P. Bowen and M. Harman. Formal Methods and Testing. Springer, 2008.

G. J. Myers, C. Sandler, T. Badgett. The Art of Software Testing (3rd edition). John Wiley & Sons, 2011.

M. Utting and B. Legeard. Practical Model-Based Testing: A Tools Approach. Morgan-Kaufmann, 2007.

Ficha docente guardada por última vez el 22/06/2022 20:37:00 por el departamento: Sistemas Informáticos y Computación

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Ficha del curso: 2023-2024

<b>Grado:</b> GRADO EN INGENIERÍA DE COMPUTADORES	<b>Curso:</b> Optativas 3º y 4º ( 2C )	<b>Idioma:</b> Español
<b>Asignatura:</b> 805761 - Informática Musical <b>Asignatura en Inglés:</b> Computer music	<b>Abrev:</b> MUS <b>Carácter:</b> Optativa	6 ECTS
<b>Materia:</b> Complementos de desarrollo de videojuegos	6 ECTS	
<b>Otras asignaturas en la misma materia:</b> No hay		
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Sistemas Informáticos y Computación		<b>Coordinador:</b> Gómez-Zamalloa Gil, Miguel

**Descripción de contenidos mínimos:**

- Procesamiento y programación básica de audio digital.
- Modelos y algoritmos básicos de síntesis de audio.
- Producción musical básica: Midi, DAWs, secuenciación, mezcla, efectos e instrumentos virtuales, y principios de composición.

**Programa detallado:**

1. Introducción.
2. Audio digital: procesamiento y programación básica.
3. Modelos y algoritmos básicos de síntesis.
4. Producción musical: Midi, DAWs, secuenciación, mezcla estéreo, efectos e instrumentos virtuales.
5. Teoría musical básica y principios de composición.

**Programa detallado en inglés:**

1. Introduction.
2. Digital audio: sample processing and basic programming.
3. Sound synthesis models and basic algorithms.
4. Music production: Midi, DAWs, sequencing, stereo mixing, effects and virtual instruments.
5. Basics of music theory and composition.

**Competencias de la asignatura:****Generales:**

- CG3-Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- CG11-Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

**Específicas:**

No tiene

**Básicas y Transversales:**

- CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

**Resultados de aprendizaje:**

- Programar algoritmos y aplicaciones que manipulan y/o generan audio digital. (CG3, CG11)
- Programar algoritmos y aplicaciones que manipulan y/o generan información musical simbólica en forma de MIDI. (CG3, CG11)
- Comprender y saber aplicar las distintas técnicas y algoritmos de síntesis de audio digital. (CG3, CG11)
- Mezclar, producir y componer música haciendo uso de herramientas informáticas. (CG3, CG11)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Realizar proyectos relacionados con el audio digital y/o la producción musical. (CT1)

**Evaluación detallada:**

Convocatoria ordinaria:

- Prácticas/ejercicios de laboratorio: 40%
- Proyecto final (obligatorio): 60%

Convocatoria extraordinaria:

- Prácticas/ejercicios de laboratorio: 40%
- Proyecto final (obligatorio): 60%

En ambas convocatorias, para poder aprobar será necesario tener al menos un 5 sobre 10 tanto en la nota del proyecto final como en la nota de las prácticas/ejercicios de laboratorio.

La nota obtenida en la convocatoria ordinaria en las prácticas/ejercicios de laboratorio se guarda para la convocatoria extraordinaria. En caso de no llegar al mínimo exigido, podrán (re)entregarse los ejercicios que el profesor establezca para llegar a este mínimo. La nota obtenida en la convocatoria ordinaria en el proyecto final puede guardarse para la convocatoria extraordinaria o incrementarse en caso de (re)entregar el proyecto en la convocatoria extraordinaria.

**Actividades docentes:**

Reparto de créditos:

- Teoría: 3,00
- Problemas: 0,00
- Laboratorios: 3,00

Otras actividades:

No tiene

**Bibliografía:**

- The Computer Music Tutorial. Curtis Roads. MIT Press, 1996.
- Computer Music Instruments Foundations, Design and Development. Victor Lazzarini. Springer International Publishing, 2017.
- Music: A Mathematical Offering. Dave Benson, 2008. Disponible en [http://logosfoundation.org/kursus/music\\_math.pdf](http://logosfoundation.org/kursus/music_math.pdf)
- Real Sound Synthesis for Interactive Applications. Perry R. Cook. A. K. Peters, Ltd., 2003.
- The Audio Programming Book. Edited by Richard Boulanger and Victor Lazzarini. The MIT Press, 2011
- Python Tutorial Release 3.7.0. Guido van Rossum. Guido van Rossum and the Python development team. 2018. Disponible en: [https://bugs.python.org/file47781/Tutorial\\_EDIT.pdf](https://bugs.python.org/file47781/Tutorial_EDIT.pdf)
- Hack audio: an introduction to computer programming and digital signal processing in MATLAB. Eric Tarr, Routledge (Audio Engineering Society Presents), 2019.
- Reaper user guide. Disponible en <http://reaper.fm/userguide.php>
- JUCE tutorials. Disponibles en <https://juce.com/learn/tutorials>
- Music Theory: The TL;DR Version. Reginal Young. Disponible en <https://gumroad.com/l/tldrmusic>
- Home Recording for Beginners. Geoffrey Francis. Cengage Learning PTR, 2009.
- Designing Audio Effect Plug-Ins in C++: With Digital Audio Signal Processing Theory. Will Pirkle, Focal Press, 2012.
- The Sounds of Music: Perception and Notation. Gerald Eskelin. Bertrams, 1998.

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento: