



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 1º (1C)	Idioma: Español
Asignatura: 901950 - Fundamentos de computadores I		Abrev: FC1	6 ECTS
Asignatura en Inglés: Introduction to Computers I		Carácter: Formación básica	
Materia: Informática		24 ECTS	
Otras asignaturas en la misma materia:			
Fundamentos de computadores II		6 ECTS	
Fundamentos de la programación I		6 ECTS	
Fundamentos de la programación II		6 ECTS	
Módulo: Materias básicas			
Departamento: Arquitectura de Computadores y Automática		Coordinador: 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:

Tema 1. 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.
- Álgebra 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: conmutatividad, 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:

- Nota de clase: problemas, participación, pruebas de clase o test (NPro) = 10%
- Nota de Prácticas (NPra) = 25%
- Examen (NExa) = 65%

Es la mayor de las dos puntuaciones siguientes:

- $NExa * 0,65 + NPra * 0,25 + NPro * 0,10$

- $NExa * 0,75 + NPra * 0,25$

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 16/06/2021 11:02:00 por el usuario: David Soria Jimenez

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 1º (2C)	Idioma: Español
Asignatura: 901951 - Fundamentos de computadores II		Abrev: FC2	6 ECTS
Asignatura en Inglés: Introduction to Computers II		Carácter: Formación básica	
Materia: Informática		24 ECTS	
Otras asignaturas en la misma materia:			
Fundamentos de computadores I		6 ECTS	
Fundamentos de la programación I		6 ECTS	
Fundamentos de la programación II		6 ECTS	
Módulo: Materias básicas			
Departamento: Arquitectura de Computadores y Automática		Coordinador: 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:

- Diseño del procesador.
 - ¿Qué es un computador?
 - Simplificación de la ejecución de un programa
 - Procesadores ARM
 - Arquitectura básica ARM: Registros y Memoria
 - Lenguaje ensamblador: definiciones
 - ARM instrucciones básicas de ensamblador y directivas
 - Llamada a subrutinas
 - Representación de datos
- Lenguaje máquina y ensamblador.
 - Introducción al diseño de un procesador sencillo
 - Procesador multi-ciclo
 - Diseño de la unidad de control
 - Rendimiento
- Sistema de memoria de un computador.
 - Organización de la memoria
 - Jerarquía de memoria
 - Introducción a la memoria cache

Programa detallado en inglés:

- Module 1. Computers: Machine and assembly language
- What is a computer?
 - A simplistic view of program execution
 - ARM processors
 - ARM basic architecture: Registers and Memory
 - Assembly language: definitions
 - ARM basic assembly instructions and directives
 - Calling subroutines
 - Data representation
- Module 2. Designing the Processor
- Introduction to the design of a simple processor
 - A multicycle processor
 - Design of the control unit
 - Performance

Module 3: The memory subsystem

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- Memory organization
- Memory hierarchy
- Introduction to the 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:

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) = 10%
- Nota de Prácticas (NPra) = 25%
- Examen (NExa) = 65%

Es la mayor de las dos puntuaciones siguientes:

- $NExa * 0,65 + NPra * 0,25 + NPro * 0,10$

- $NExa * 0,75 + NPra * 0,25$

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 ARM Edition. Sarah Harris, David Harris. Morgan Kaufmann. 2015

Computer Organization and Design, 4th Edition, ARM Edition. D. A. Patterson and J. L. Hennessy. Morgan Kaufmann. 2009

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

Fecha: ____ de _____ de ____

Firma del Director del Departamento:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 1º (1C)	Idioma: Español
Asignatura: 901952 - Fundamentos de la programación I		Abrev: FP1	6 ECTS
Asignatura en Inglés: Fundamentals of Programming I		Carácter: Formación básica	
Materia: Informática		24 ECTS	
Otras asignaturas en la misma materia:			
Fundamentos de computadores I		6 ECTS	
Fundamentos de computadores II		6 ECTS	
Fundamentos de la programación II		6 ECTS	
Módulo: Materias básicas			
Departamento: Sistemas Informáticos y Computación		Coordinador: 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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 1º (2C)	Idioma: Español
Asignatura: 901953 - Fundamentos de la programación II		Abrev: FP2	6 ECTS
Asignatura en Inglés: Fundamentals of Programming II		Carácter: Formación básica	
Materia: Informática		24 ECTS	
Otras asignaturas en la misma materia:			
Fundamentos de computadores I		6 ECTS	
Fundamentos de computadores II		6 ECTS	
Fundamentos de la programación I		6 ECTS	
Módulo: Materias básicas			
Departamento: Sistemas Informáticos y Computación		Coordinador: 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:

- 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): 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: | Otras actividades: |
| Teoría: 3,00 | Clases teóricas: 2 horas de clases teóricas a la semana. |
| Problemas: 1,50 | Clases prácticas: 2 horas de clases de problemas/prácticas a la semana. |
| Laboratorios: 1,50 | 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:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 1º (1C)	Idioma: Español
Asignatura: 901954 - Matemática Discreta y Lógica Matemática I		Abrev: MDL1	6 ECTS
Asignatura en Inglés: Discrete Mathematics and Mathematical Logic		Carácter: Formación básica	
Materia: Matemáticas		12 ECTS	
Otras asignaturas en la misma materia: Matemática Discreta y Lógica Matemática II		6 ECTS	
Módulo: Materias básicas			
Departamento: Sistemas Informáticos y Computación		Coordinador: Bradley Delso, Margarita	

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 y orden. Relaciones de equivalencia, clases de equivalencia. Órdenes, conjuntos ordenados, retículos.

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. Numeric sets, 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 and order relations. Equivalence relations, equivalence classes. Orders, ordered sets, lattices.

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:



- 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
- 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).

En el caso de reducirse la presencialidad de la docencia en más de un 50% de las horas inicialmente previstas, los porcentajes se mantendrían: 90% examen final y 10% otras actividades.

En el caso de que la docencia pasase a ser completamente virtual se realizarían pruebas online de carácter teórico-práctico sobre los temas del programa de la asignatura para evaluar el 90% correspondiente al examen final, manteniéndose el 10% restante del apartado "Otras actividades".

Actividades docentes:

Reparto de créditos:	Otras actividades:
Teoría: 4,50	Actividades presenciales: Clases teóricas y clases prácticas 40%
Problemas: 1,50	Actividades dirigidas: Realización individual de ejercicios y problemas tutorizados 10%
Laboratorios: 0,00	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:



UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 1º (2C)	Idioma: Español
Asignatura: 901955 - Matemática Discreta y Lógica Matemática II		Abrev: MDL2	6 ECTS
Asignatura en Inglés: Discrete Mathematics and Mathematical Logic II		Carácter: Formación básica	
Materia: Matemáticas		12 ECTS	
Otras asignaturas en la misma materia: Matemática Discreta y Lógica Matemática I		6 ECTS	
Módulo: Materias básicas			
Departamento: Sistemas Informáticos y Computación		Coordinador: 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:

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

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- 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:

- 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 puntos, y la nota obtenida en otras actividades (valor máximo: 2 puntos).
En caso de que el examen final no pueda ser presencial, se dividirá en dos exámenes en línea más breves.

Actividades docentes:

Reparto de créditos:	Otras actividades:
Teoría: 5,00	Actividades presenciales: Clases teóricas y clases prácticas 40%
Problemas: 1,00	Actividades dirigidas: Realización individual de ejercicios y problemas tutorizados 10%
Laboratorios: 0,00	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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 2º (1C)	Idioma: Español
Asignatura: 900210 - Gestión empresarial		Abrev: GE	6 ECTS
Asignatura en Inglés: Business Management		Carácter: Formación básica	
Materia: Empresa		6 ECTS	
Otras asignaturas en la misma materia: No hay			
Módulo: Materias básicas			
Departamento: Administración Financiera y Contabilidad		Coordinador: 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
3. Variación de existencias

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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

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

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- 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:



Bibliografía:

BIBLIOGRAFÍA:

- Plan General de Contabilidad, 2007.
- Supuestos prácticos. Material elaborado por los profesores de la asignatura, 2010.
- Camacho, M^a del Mar y Rivero, M^a 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
- CNMV, COMISIÓN NACIONAL DEL MERCADO DE VALORES: www.cnmv.es
- FASB, FINANCIAL ACCOUNTING STANDARDS BOARD / US GAAP: www.fasb.org
- ICAC, INSTITUTO DE CONTABILIDAD Y AUDITORÍA DE CUENTAS: www.icac.mineco.es
- www.noticiasjuridicas.com
- 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^a del Mar y Rivero, M^a 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 23:00:00 por el usuario: Vic. Estudios

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 2º (2C)	Idioma: Español
Asignatura: 900211 - Fundamentos de electricidad y electrónica	Abrev: FEE	6 ECTS	
Asignatura en Inglés: Introduction to the concepts of electricity and electronics	Carácter: Formación básica		
Materia: Física		6 ECTS	
Otras asignaturas en la misma materia: No hay			
Módulo: Materias básicas			
Departamento: Interdepartamental EMFTEL/FM		Coordinador: 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.
- Transistor bipolar de unión.
- 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 Kirchhoff. 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. Ecuaciones características y regiones de operación. Circuitos prácticos con transistores MOS. Introducción a la lógica CMOS. 2. Transistor bipolar de unión (BJT): Estructura y funcionamiento básico.

Programa detallado en inglés:

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

Topic I. Electric field. Electric current

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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.

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

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

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:

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:

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)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMATICA

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:

$$CFINAL = 0.8 CEX + 0.2 CACT \text{ o } CFINAL = 0.9 CEX + 0.1 CACT$$

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%)

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. Etxebarria. Análisis Básico de Circuitos Eléctricos y Electrónicos. Pearson Prentice Hall.

Ficha docente guardada por última vez el 17/07/2020 17:07:00 por el departamento: Interdepartamental EMFTEL/FM

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 INFORMÁTICA**

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 2º (1C)	Idioma: Español
Asignatura: 901956 - Ingeniería del software I		Abrev: IS1	4,5 ECTS
Asignatura en Inglés: Software Engineering I		Carácter: Obligatoria	
Materia: Desarrollo del software fundamental		15 ECTS	
Otras asignaturas en la misma materia:			
Bases de datos		6 ECTS	
Ingeniería del software II		4,5 ECTS	
Módulo: Materias comunes a la rama de la informática			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Gervás Gómez-Navarro, Pablo	

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.
- 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:



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 24/06/2021 14:04:00 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 INFORMÁTICA**

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 2º (2C)	Idioma: Español
Asignatura: 901957 - Ingeniería del software II		Abrev: IS2	4,5 ECTS
Asignatura en Inglés: Software Engineering II		Carácter: Obligatoria	
Materia: Desarrollo del software fundamental		15 ECTS	
Otras asignaturas en la misma materia:			
Bases de datos		6 ECTS	
Ingeniería del software I		4,5 ECTS	
Módulo: Materias comunes a la rama de la informática			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Gervás Gómez-Navarro, Pablo	

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.
 - 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: | Otras actividades: |
| Teoría: 1,25 | Clases teóricas |
| Problemas: 0,25 | Sí, de exposición de los principales conceptos teóricos de la asignatura y resolución de casos prácticos. |
| Laboratorios: 3,00 | Seminarios |
| | Sí, sobre temas de especial relevancia e impartidos por expertos del área. |
| | Clases prácticas |
| | Sí |
| | Laboratorios |

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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 24/06/2021 14:07:00 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:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 2º (1C)	Idioma: Español
Asignatura: 901958 - Fundamentos de algoritmia		Abrev: FAL	4,5 ECTS
Asignatura en Inglés: Fundamentals of Algorithms		Carácter: Obligatoria	
Materia: Programación fundamental		20 ECTS	
Otras asignaturas en la misma materia:			
Estructuras de datos		4,5 ECTS	
Tecnología de la programación I		6 ECTS	
Tecnología de la programación II		6 ECTS	
Módulo: Materias comunes a la rama de la informática			
Departamento: Sistemas Informáticos y Computación		Coordinador: Segura Díaz, Clara M ^a	

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.

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)
- 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)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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:

R. Peña.

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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 2º (2C)	Idioma: Español
Asignatura: 901959 - Estructuras de datos		Abrev: ED	4,5 ECTS
Asignatura en Inglés: Data structures		Carácter: Obligatoria	
Materia: Programación fundamental		20 ECTS	
Otras asignaturas en la misma materia:			
Fundamentos de algoritmia		4,5 ECTS	
Tecnología de la programación I		6 ECTS	
Tecnología de la programación II		6 ECTS	
Módulo: Materias comunes a la rama de la informática			
Departamento: Sistemas Informáticos y Computación		Coordinador: Montenegro Montes, Manuel	

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:**

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:

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)

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.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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. Si en alguna de las convocatorias el examen no pudiese realizarse presencialmente, este podrá dividirse en dos controles y un examen final cuyos respectivos pesos son 20%, 20% y 30% del total de la asignatura. En ese caso, para aprobar será necesario obtener una puntuación de 3.5 sobre 7 en las pruebas realizadas.

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 15/07/2020 9:16:00 por el departamento: **Sistemas Informáticos y Computación**

Fecha: ____ de _____ de ____

Firma del Director del Departamento:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 2º (1C)	Idioma: Español
Asignatura: 901960 - Tecnología de la programación I		Abrev: TP1	6 ECTS
Asignatura en Inglés: Programming Technology I		Carácter: Obligatoria	
Materia: Programación fundamental		20 ECTS	
Otras asignaturas en la misma materia:			
Estructuras de datos		4,5 ECTS	
Fundamentos de algoritmia		4,5 ECTS	
Tecnología de la programación II		6 ECTS	
Módulo: Materias comunes a la rama de la informática			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Valero Espada, Miguel Ángel	

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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 2º (2C)	Idioma: Español
Asignatura: 901961 - Tecnología de la programación II		Abrev: TP2	6 ECTS
Asignatura en Inglés: Programming Technology II		Carácter: Obligatoria	
Materia: Programación fundamental		20 ECTS	
Otras asignaturas en la misma materia:			
Estructuras de datos		4,5 ECTS	
Fundamentos de algoritmia		4,5 ECTS	
Tecnología de la programación I		6 ECTS	
Módulo: Materias comunes a la rama de la informática			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: 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:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 3º (1C)	Idioma: Español
Asignatura: 900221 - Bases de datos		Abrev: BD	6 ECTS
Asignatura en Inglés: Databases		Carácter: Obligatoria	
Materia: Desarrollo del software fundamental			15 ECTS
Otras asignaturas en la misma materia: Ingeniería del software			9 ECTS
Módulo: Materias comunes a la rama de la informática			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: 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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: 3º (2C)	Idioma: Español
Asignatura: 900222 - Estructura de computadores	Abrev: EC	6 ECTS
Asignatura en Inglés: Computer Organization	Carácter: Obligatoria	
Materia: Estructura de computadores		6 ECTS
Otras asignaturas en la misma materia: No hay		
Módulo: Materias comunes a la rama de la informática		
Departamento: Arquitectura de Computadores y Automática		
Coordinador: Molina Prego, M ^a 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:

Módulo 1. Entrada/salida

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

Módulo 2. Arquitectura del procesador

Repertorio de instrucciones ARM.

Módulo 3. Diseño del procesador

Segmentación. Riesgos: Estructurales, de datos y de control. Segmentación del procesador. Operaciones multi-ciclo. Gestión de excepciones. Rendimiento del procesador segmentado.

Módulo 4. Jerarquía de memoria

Jerarquía de memoria. Memoria cache. Rendimiento de la memoria cache. Optimización de la memoria cache. La memoria principal. Memoria virtual. Gestión de memoria virtual. Protección.

Programa detallado en inglés:

Module 1. Input / Output

I/O System: Structure and function. Interconnection System. Interrupts. DMA

Module 2. Processor Architecture

ARM architecture.

Module 3. Processor Design

Pipelining. Hazards: Structural, data and control. Pipelining implementation. Multi-cycle operations. Exceptions. Performance.

Module 4. Memory Hierarchy

Memory Hierarchy. Cache memory. Performance. Cache optimization. Main memory. Virtual memory. Virtual Memory Management. Protection.

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.

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.

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)

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:

1. Exámenes

Examen final en aula

2. Método de evaluación:

Para la evaluación se tienen en cuenta los siguientes elementos:

a) Nota de laboratorio: Es la media ponderada de las calificaciones de las prácticas.

b) Pruebas de clase (problemas, controles, tests,...)

c) Exámenes: Examen tanto en la convocatoria ordinaria como en la extraordinaria. Todos los exámenes son escritos y están formados por cuestiones teóricas y problemas. Mismo examen, en todos los grupos de la asignatura, criterios detallados de puntuación comunes.

3. Calificación

- Nota del examen * 0,70 + Nota de Prácticas * 0,30

La nota de pruebas de clase y la nota de las prácticas 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:

W. Stallings; Organización y Arquitectura de Computadores; Prentice Hall, 2006;

D.A. Patterson y J.L. Hennessy; Estructura y diseño de computadores. La interfaz hardware/software; Reverté, 2011;

A. Cuesta, J.I. Hidalgo, J., J.L. Risco; Problemas de fundamentos y estructura de computadoras; Pearson, 2009;

S. Furber; ARM System-on-Chip architecture; Addison-Wesley, 2000.

Sarah Harris y David Harris."Digital Design and Computer Architecture. ARM Edition", Elsevier 2015

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 3º (1C)	Idioma: Español
Asignatura: 900223 - Tecnología y organización de computadores		Abrev: TOC	6 ECTS
Asignatura en Inglés: Technology and Organization of Computer Systems		Carácter: Obligatoria	
Materia: Tecnología y Arquitectura de Computadores			12 ECTS
Otras asignaturas en la misma materia: Arquitectura de Computadores			6 ECTS
Módulo: Complementario			
Departamento: Arquitectura de Computadores y Automática		Coordinador: Garnica Alcazar, Oscar	

Descripción de contenidos mínimos:

- Circuitos aritméticos.
- Diseño multimódulo.
- Sistemas algorítmicos.
- Organización de la memoria.
- Lenguajes de descripción de HW.
- Prácticas de diseño de circuitos digitales.

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
- 2.4. Análisis del área
- 2.5. 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
- 5.6. Códigos de detección de errores

Tema 6. Aritmética

- 6.1. Sumadores rápidos (anticipación y puenteo de arrastres)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- 6.2. Multiplicadores sin/con signo (secuencial y combinacional)
- 6.3. Divisor secuencial
- 6.4. Representación IEEE 754
- 6.5. Suma, multiplicación en punto flotante
- 6.6. Precisión y redondeo IEEE 754

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
 - 6.4. IEEE 754 Representation
 - 6.5. Floating Point Addition and Multiplication
 - 6.6. IEEE 754 Accuracy and Rounding

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.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMATICA

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 y diseñar la estructura de un sistema electrónico digital de complejidad media-alta. (CG4)
- Aplicar técnicas para la depuración de sistemas electrónicos digitales mediante simulación. (CG4)
- Combinar componentes hardware para el diseño de un sistemas electrónico digital. (CG4)
- 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)
- Diseñar los componentes básicos de un computador usando metodologías y herramientas de diseño de circuitos electrónicos digitales. (CG4)
- 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)
- Utilizar una plataforma basada en dispositivos programables para implementar sistemas electrónicos digitales. (CG4)
- 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 metodología de diseño RTL para generar la descripción de un sistema electrónico digital. (CG4)
- Aplicar la teoría de circuitos combinatoriales y secuenciales para diseñar y evaluar distintas opciones de diseño de una especificación dada. (CG14)
- Comprender la importancia de los sistemas digitales síncronos. (CG4)
- 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 las ventajas e inconvenientes de las alternativas tecnológicas en el diseño o fabricación de los sistemas electrónicos digitales. (CG4)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- Evaluar una especificación y justificar modificaciones basadas en la tecnología. (CT2)
- Evaluar y aplicar metodologías, estrategias, técnicas y herramientas CAD profesionales para el diseño de sistemas o subsistemas electrónicos digitales. (CG4)
- Experimentar el desarrollo de sistemas digitales de complejidad media-alta. (CT3)
- Planear las necesidades en recursos humanos de la ejecución de un proyecto de desarrollo hardware, así como su ejecución. (CT4)
- Usar los medios audiovisuales para expresar la información de un proyecto de sistema electrónico digital de forma apropiada para su comprensión por el cliente y el equipo de desarrollo. (CT1)
- Valorar el impacto y cambio social que puede involucrar la puesta en marcha de un sistema electrónico digital. (CT5)
- Valorar y seleccionar alternativas de diseño. (CT2)
- Valorar y seleccionar la tecnología de memoria adecuada para la aplicación objetivo. (CG14)

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 una o varias pruebas individuales. Dichas pruebas individuales supondrán entre 60-70% de la nota del laboratorio y servirán para evaluar que se han adquirido todas las destrezas relacionadas con la programación en VHDL. La nota de las prácticas de laboratorio no es recuperable en la convocatoria extraordinaria. Es decir, la nota de las prácticas será la obtenida en la convocatoria ordinaria y no habrá posibilidad de recuperar o mejorar dicha calificación en la convocatoria extraordinaria.

-Otras actividades en el aula: entrega de problemas y test.

-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 mayor de las dos puntuaciones siguientes:
 $0,65 * \text{Nota del examen} + 0,20 * \text{Nota del laboratorio} + 0,15 * \text{Nota otras actividades en el aula}$
 $0,65 * \text{Nota del examen} + 0,35 * \text{Nota del laboratorio}$

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.

Bibliografía:

- Gajski, D.D., "Principios de diseño digital", Prentice Hall, 1997
- 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:



UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 3º (2C)	Idioma: Español
Asignatura: 900224 - Fundamentos de los lenguajes informáticos		Abrev: FLI	6 ECTS
Asignatura en Inglés: Foundations of computer languages		Carácter: Obligatoria	
Materia: Lenguajes informáticos y procesadores de lenguaje		12 ECTS	
Otras asignaturas en la misma materia: Procesadores de Lenguajes		6 ECTS	
Módulo: Tecnología específica: Computación			
Departamento: Sistemas Informáticos y Computación		Coordinador: Ullán Hernández, Eva	

Descripción de contenidos mínimos:

- Introducción a los lenguajes formales.
- Lenguajes regulares: expresiones, reconocedores y propiedades.
- Estructura léxica de los lenguajes de programación.
- Lenguajes incontextuales: gramáticas, reconocedores y propiedades.
- Estructura sintáctica de los lenguajes de programación.
- Lenguajes recursivos y recursivamente enumerables: gramáticas y reconocedores.
- Introducción a la teoría de la computabilidad.

Programa detallado:

1. Introducción a los autómatas y los lenguajes formales
2. Lenguajes regulares: autómatas finitos y expresiones regulares
3. Lenguajes independientes del contexto: autómatas con pila y gramáticas independientes del contexto
4. Lenguajes recursivos y recursivamente enumerables: máquinas de Turing

Programa detallado en inglés:

1. Introduction to automata and formal languages
2. Regular languages: finite automata and regular expressions
3. Context-free languages: pushdown automata and context-free grammars
4. Recursive and recursively enumerable languages: Turing machines

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.

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.

CE_C3-Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.

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:

Aplicar algoritmos de determinización y minimización de autómatas finitos (CE_C1, CE_C2, CE_C3)

Aplicar el lema de iteración para demostrar que un lenguaje no es regular o no es independiente del contexto (CG2, CE_C1, CE_C2, CE_C3)

Calcular la expresión regular asociada a un autómata finito (CE_C1, CE_C2, CE_C3)

Clasificar lenguajes según las categorías principales de la jerarquía de Chomsky (CT2, CE_C1, CE_C2, CE_C3)

Fecha: ____ de _____ de _____

Firma del Director del Departamento:



- Construir un autómata con pila para reconocer un lenguaje independiente del contexto (CE_C1, CE_C2, CE_C3)
- Construir un autómata finito para reconocer un lenguaje regular (CE_C1, CE_C2, CE_C3)
- Contrastar las principales propiedades de clausura y decisión de tales categorías (CT1, CE_C1, CE_C2, CE_C3)
- Diseñar un gramática independiente del contexto para representar un lenguaje independiente del contexto (CE_C1, CE_C2, CE_C3)
- Diseñar una máquina de Turing para reconocer un lenguaje o calcular una función (CT3, CE_C1, CE_C2, CE_C3)

Evaluación detallada:

Convocatoria ordinaria:

- 15% participación activa durante el cuatrimestre.
- 85% en base a un conjunto de exámenes, parte de los cuales se repartirán a lo largo del cuatrimestre

Convocatoria extraordinaria:

- 15% participación activa durante el cuatrimestre
- 85% examen final

Actividades docentes:

Reparto de créditos:

Teoría: 4,50

Problemas: 1,50

Laboratorios: 0,00

Otras actividades:

- Actividades dirigidas: realización y entrega de ejercicios siguiendo un calendario establecido.
- Trabajo personal no dirigido: estudio, preparación de exámenes, realización de ejercicios.
- Realización de exámenes (parcial y finales).

Bibliografía:

- John E. Hopcroft, Rajeev Motwani & Jeffrey D. Ullman. Introducción a la Teoría de Autómatas, Lenguajes y Computación. Tercera edición. Pearson Addison-Wesley, 2010. ([M519.713HOP])
- Peter Linz. An Introduction to Formal Languages and Automata. Fifth Edition. Jones & Bartlett, 2012. ([M519.76LIN]. Disponible 6ª edición en libro electrónico)
- John C. Martin. Introduction to Languages and the Theory of Computation. Fourth Edition. McGraw-Hill, 2011. ([M519.6/.7MAR])
- Dexter C. Kozen. Automata and Computability. Springer, 1997. ([M519.713KOZ])
- Dean Kelley. Teoría de Autómatas y Lenguajes Formales. Pearson Prentice Hall, 1995. ([M519.76KEL])
- Susan H. Rodger & Thomas W. Finley. JFLAP: An Interactive Formal Languages and Automata Package. Jones & Bartlett, 2006. ([M519.76JFL])

Ficha docente guardada por última vez el 22/07/2020 15:38:00 por el departamento: Interdepartamental ISIA / SIC

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 3º (1C)	Idioma: Español
Asignatura: 901962 - Métodos algorítmicos en resolución de problemas I		Abrev: MAR1	ECTS: 4,5 ECTS
Asignatura en Inglés: Algorithmic methods in problem solving I		Carácter: Obligatoria	
Materia: Programación avanzada		20 ECTS	
Otras asignaturas en la misma materia: Métodos algorítmicos en resolución de problemas II Programación Concurrente Programación Declarativa		4,5 ECTS 6 ECTS 6 ECTS	
Módulo: Tecnología específica: Computación			
Departamento: Sistemas Informáticos y Computación		Coordinador: Martí Oliet, Narciso	

Descripción de contenidos mínimos:

Estructuras arbóreas avanzadas.
Colas de prioridad y montículos.
Grafos.
Métodos voraces.

Programa detallado:

1. Complejidad media de algoritmos. Análisis amortizado
2. Árboles de búsqueda avanzados
3. Colas con prioridad y montículos
4. Grafos
5. Estructuras de partición
6. Algoritmos voraces

Programa detallado en inglés:

1. Average complexity and amortized analysis of algorithms
2. Advanced search trees
3. Priority queues and heaps
4. Graphs
5. Data structures for disjoint sets
6. Greedy 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.
- 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.
- 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:

- 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_C3-Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.

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.

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 y justificar el coste medio y amortizado de algoritmos (CG2, CG11, CT1, CT5, CE_C3)

Comparar el comportamiento de distintas estructuras arbóreas para implementar conjuntos, diccionarios y colas con prioridad (CG2, CG3, CG11, CG12, CT1)

Conocer la complejidad intrínseca de problemas clásicos (CG2, CG11, CT1, CT5, CE_C1, CE_C3)

Discernir qué métodos generales son apropiados para tratar distintas clases de problemas (CG3, CG11, CG13, CT1, CT2, CT3, CE_C1, CE_C3)

Utilizar algoritmos de recorrido, camino mínimo y árboles de expansión mínimos sobre grafos (CG2, CG3, CG11, CG12, CT1)

Utilizar los patrones generales de implementación de los métodos de diseño de algoritmos adecuándolos a cada caso concreto (CG3, CG11, CG13, CT1, CT2, CT3, CE_C3)

Evaluación detallada:

Un 30% de la nota se obtendrá a lo largo del curso mediante la evaluación de actividades prácticas (entrega de problemas, programas o trabajos, discusiones en clases prácticas, posibles tutorías obligatorias, presentaciones etc.). No existirá nuevo periodo de entrega de prácticas o trabajos para la convocatoria extraordinaria.

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 4 sobre 10 en dicho examen.

Actividades docentes:

Reparto de créditos:

Teoría: 3,00

Problemas: 1,50

Laboratorios: 0,00

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:

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.

S. Dasgupta, C. H. Papadimitriou, U. V. Vazirani, Algorithms, McGraw-Hill Education, 2006.

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.

J. Kleinberg, É. Tardos. Algorithm Design. Pearson/Addison Wesley, 2006.

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:



Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 3º (2C)	Idioma: Español
Asignatura: 901963 - Métodos algorítmicos en resolución de problemas II		Abrev: MAR2	4,5 ECTS
Asignatura en Inglés: Algorithmic methods in problem solving II		Carácter: Obligatoria	
Materia: Programación avanzada		20 ECTS	
Otras asignaturas en la misma materia:			
Métodos algorítmicos en resolución de problemas I		4,5 ECTS	
Programación Concurrente		6 ECTS	
Programación Declarativa		6 ECTS	
Módulo: Tecnología específica: Computación			
Departamento: Sistemas Informáticos y Computación		Coordinador: Segura Díaz, Clara Mª	

Descripción de contenidos mínimos:

Programación dinámica.
Métodos de búsqueda en espacios de estados y en árboles de juegos.
Algoritmos probabilísticos.
Complejidad de problemas.

Programa detallado:

1. Programación dinámica
2. Precondicionamiento
3. Ramificación y acotación
4. Árboles de juego
5. Algoritmos probabilistas
6. Complejidad de problemas
7. Algoritmos aproximados

Programa detallado en inglés:

1. Dynamic programming
2. Preconditioning and precomputation
3. Branch and bound
4. Game trees
5. Probabilistic algorithms
6. Computational complexity
7. Approximate 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.
- 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.
- 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:

- 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_C3-Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.

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.

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 y justificar el coste medio y amortizado de algoritmos (CG2, CG11, CT1, CT5, CE_C3)

Conocer la complejidad intrínseca de problemas clásicos (CG2, CG11, CT1, CT5, CE_C1, CE_C3)

Desarrollar soluciones aproximadas de problemas, sea cuantitativamente o en términos de probabilidad (CG3, CG11, CT1, CT2, CT3)

Discernir qué métodos generales son apropiados para tratar distintas clases de problemas (CG3, CG11, CG13, CT1, CT2, CT3, CE_C1, CE_C3)

Implementar estrategias ganadoras en juegos bipersonales o aproximar las mismas (CG2, CT1, CT2, CT3)

Utilizar los patrones generales de implementación de los métodos de diseño de algoritmos adecuándolos a cada caso concreto (CG3, CG11, CG13, CT1, CT2, CT3, CE_C3)

Evaluación detallada:

Un 30% de la nota se obtendrá a lo largo del curso mediante la evaluación de actividades prácticas (entrega de problemas, programas o trabajos, discusiones en clases prácticas, posibles tutorías obligatorias, presentaciones etc.). No existirá nuevo periodo de entrega de prácticas o trabajos para la convocatoria extraordinaria.

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 4 sobre 10 en dicho examen.

Actividades docentes:

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 1,50

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:

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.

S. Dasgupta, C. H. Papadimitriou, U. V. Vazirani, Algorithms, McGraw-Hill Education, 2006.

E. Horowitz, S. Sahni, S. Rajasekaran. Computer Algorithms, Computer Science Press, 1998. Second Edition, Silicon Press, 2007.

J. Kleinberg, É. Tardos. Algorithm Design. Pearson/Addison Wesley, 2006.

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.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 4º (1C)	Idioma: Español
Asignatura: 900226 - Programación Declarativa		Abrev: PD	6 ECTS
Asignatura en Inglés: Declarative Programming		Carácter: Obligatoria	
Materia: Programación avanzada		21 ECTS	
Otras asignaturas en la misma materia:			
Métodos algorítmicos en resolución de problemas		9 ECTS	
Programación Concurrente		6 ECTS	
Módulo: Tecnología específica: Computación			
Departamento: Sistemas Informáticos y Computación		Coordinador: Nieva Soto, Susana	

Descripción de contenidos mínimos:

Programación imperativa vs programación declarativa.
Paradigma funcional: funciones y evaluación de expresiones.
Tratamiento funcional de estructuras de datos y algoritmos.
Paradigma lógico: predicados y resolución de objetivos.
Tratamiento lógico de estructuras de datos y algoritmos.

Programa detallado:

- Elementos básicos de la programación funcional: funciones y expresiones, tipos, orden superior, lambda abstracciones.
- Ejecución de programas funcionales: evaluación impaciente y perezosa, ajuste de patrones.
- Tipos de datos: tipos definidos, polimórficos, inferencia de tipos, clases de tipos.
- Técnicas básicas de programación funcional.
- Elementos básicos de la programación lógica: relaciones, términos, hechos, cláusulas, variables lógicas.
- Ejecución de programas lógicos: unificación, resolución, espacio de búsqueda.
- Programación lógica con datos estructurados
- Programación en lenguaje Prolog: control, predicados metalógicos.

Programa detallado en inglés:

- Basic notions of functional programming: functions and expressions, types, higher order functions, lambda-abstractions.
- Execution of functional programs: eager and lazy evaluation, pattern matching.
- Data types: user-defined types, polymorphism, type inference, type classes.
- Basic techniques of functional programming.
- Basic notions of logic programming: relations, terms, clauses, logical variables.
- Execution of logic programs: unification, resolution, search space.
- Logic programming with structured data.
- Programming in Prolog: control and metalogic predicates.

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.
- 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:

- 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.
- CE_C3-Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.

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 la aportación de los recursos propios de los lenguajes funcionales (ajuste de patrones, orden superior, lambda expresiones, evaluación perezosa, polimorfismo, ...) a la construcción de programas concisos, abstractos y correctos (CG11, CG12, CG13, CE_C3)

Calcular la complejidad de programas funcionales sencillos (CG11, CE_C3)

Comparar críticamente la resolución de un mismo problema en el paradigma funcional y el paradigma lógico (CG13, CT1, CE_C1, CE_C2)

Conocer el mecanismo concreto de recorrido del árbol de resolución utilizado por el lenguaje Prolog (CE_C1, CE_C2)

Conocer la diferencia entre polimorfismo paramétrico y polimorfismo de clases (CG12, CG13, CE_C2)

Conocer y manipular con fluidez las particularidades sintácticas específicas de los lenguajes funcionales modernos (CG3, CE_C1, CE_C2)

Construir árboles de resolución como representación abstracta adecuada de los cómputos lógicos (CG13, CE_C1, CE_C2)

Desarrollar programas lógicos para resolver problemas que involucran búsqueda (CG11, CG12, CE_C3)

Diseñar programas que realizan interacciones mediante el modelo de la entrada/salida monádica (CG13, CE_C2)

Diseñar tipos de datos algebraicos para representar adecuadamente la información en lenguajes funcionales (CG12, CT3, CE_C2)

Diseñar y definir clases de tipos e instancias de ellas (CG12, CG13, CE_C2)

Distinguir el comportamiento de los cómputos funcionales dependiendo del régimen de evaluación utilizado (CG11, CE_C2, CE_C3)

Inferir sistemáticamente tipos de expresiones y funciones en el sistema de Hindley-Milner (CG12, CE_C1, CE_C2)

Interpretar un cómputo lógico como una deducción lógica que determina un espacio de búsqueda (CE_C1, CE_C2)

Interpretar un programa lógico como una teoría lógica en la lógica de Horn (CE_C1, CE_C2)

Razonar con precisión propiedades de programas funcionales (CG11, CT2, CE_C1, CE_C2)

Reconocer la diferencia entre programas lógicos puros y programas Prolog que usan recursos adicionales (CE_C1, CE_C2)

Resolver problemas de unificación sintáctica (CE_C1, CE_C2)

Valorar el contenido computacional de las variables lógicas y el proceso de extracción de respuestas (CG13, CE_C1, CE_C2)

Valorar la facilidad aportada por Prolog a la metaprogramación (CG11, CG13, CE_C1, CE_C2)

Evaluación detallada:

La calificación tanto en la convocatoria ordinaria como en la extraordinaria tiene en cuenta las siguientes actividades y pruebas a realizar por el estudiante:

A. Un examen a mitad del cuatrimestre, resolución de ejercicios prácticos y cuestionarios realizados en clase, participación activa en las clases teóricas y de laboratorio: 20%

B. Realización de un trabajo práctico asignado: 10%

Para la evaluación de este trabajo el profesor podrá convocar al alumno.

C. Examen final: 70%

La nota final será la suma de las calificaciones de estos tres apartados.

La calificación obtenida en los puntos A y B en la convocatoria ordinaria se mantendrá para la convocatoria extraordinaria, no siendo recuperable en esta segunda convocatoria.

Actividades docentes:

Reparto de créditos:

Teoría: 4,50

Problemas: 0,00

Laboratorios: 1,50

Otras actividades:

Actividad presencial (40%): clases teóricas y de resolución de ejercicios; clases prácticas en laboratorio, en sesiones de dos horas en semanas alternas.

Actividades dirigidas (10%): trabajos dirigidos.

Trabajo personal (50%).

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMÁTICA

Bibliografía:

Libros de programación funcional

- * R. Bird; Introducción a la Programación Funcional con Haskell; Segunda edición, Prentice Hall, 2000;
- * B.C. Ruiz, F. Gutiérrez, P. Guerrero, J.E. Gallardo; Razonando con Haskell: un curso sobre programación funcional; Thomson, 2004;
- * G. Hutton; Programming in Haskell; Cambridge University Press, 2007;
- * M. Lipovaca, Learn You a Haskell for Great Good; No Starch Press, 2011. (Existe versión on-line);

Libros de programación lógica

- * L.Sterling, E.Shapiro; The Art of Prolog. Advanced Programming Techniques; The MIT Press, 2ª Edición, 1994;
- * P. Blackburn, J. Bos, K. Striegnitz; Learn Prolog Now!; Colledge Publications, 2006. (Existe versión on-line);
- * P. Julián, M. Alpuente; Programación Lógica, Teoría y Práctica; Pearson, 2007;
- * W.F. Clocksin, C.S. Mellish; Programming in Prolog Using the ISO Standard; Springer Verlag, 5ª edición, 2003;

Ficha docente guardada por última vez el 15/06/2021 18:18:00 por el departamento: **Sistemas Informáticos y Computación**

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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 4º (1C)	Idioma: Español
Asignatura: 900232 - Redes		Abrev: RED	6 ECTS
Asignatura en Inglés: Computer Networks		Carácter: Obligatoria	
Materia: Sistemas operativos y redes fundamentales		12 ECTS	
Otras asignaturas en la misma materia: Sistemas Operativos		6 ECTS	
Módulo: Materias comunes a la rama de la informática			
Departamento: Arquitectura de Computadores y Automática		Coordinador: 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 se realizará un examen final que incluye una parte de problemas y cuestiones teóricas (85%). En ambas convocatorias para calificar la parte práctica del laboratorio se deberán responder una serie de cuestiones sobre el laboratorio (15%). La calificación conseguida en la parte del laboratorio no se guarda entre convocatorias. Estos exámenes se realizarán preferiblemente de forma presencial en el aula, pero si fuese necesario, se podrían realizar de forma no presencial (on-line).

Para aprobar la asignatura es necesario obtener una nota total mínima de 5 puntos sobre 10 en el examen final.

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 17/06/2021 11:50:00 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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 4º (1C)	Idioma: Español
Asignatura: 900233 - Sistemas Operativos		Abrev: SO	6 ECTS
Asignatura en Inglés: Operating Systems		Carácter: Obligatoria	
Materia: Sistemas operativos y redes fundamentales		12 ECTS	
Otras asignaturas en la misma materia: Redes		6 ECTS	
Módulo: Materias comunes a la rama de la informática			
Departamento: Arquitectura de Computadores y Automática		Coordinador: Olcoz Herrero, Katzalin	

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

=====

1. Introducción a la programación de sistemas.

2. Práctica de sistemas de ficheros

3. Práctica de procesos/hilos y sincronización.

4. Práctica de 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

Fecha: ____ de _____ de _____

Firma del Director del Departamento:



- 3.2.1 Introduction to the OS scheduler
- 3.2.2 Scheduling algorithms: FCFS, priority, round-robin
- 3.3 Threads
 - 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

====

1. Introduction to system programming.
2. Lab assignment on file systems
3. Lab assignment on processes threads and synchronization.
4. Lab assignment on 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)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- 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)
- 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 prácticas de la asignatura: Práctica especial de laboratorio (se realizarán diferentes ejercicios teórico-prácticos el día asignado por el profesor).
2. Nota del examen. Habrá examen final teórico en ambas convocatorias. El examen será común para todos los grupos de la asignatura, será escrito y estará formado por cuestiones teóricas y problemas.
3. Nota de 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 * 0,60 + Nota de Prácticas * 0,20 + Nota Pruebas de clase * 0,20
- Nota del examen * 0,70 + Nota de Prácticas * 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 el examen teórico. En caso de haber obtenido una nota inferior a 4 en el examen teórico, la nota final de la asignatura será la obtenida en el examen teórico.

La nota de pruebas de clase es la obtenida en la convocatoria ordinaria, es decir, no hay posibilidad de recuperar/mejorar esa calificación.

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
- 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. 2001.
- Machtelt Garrels. Bash Guide for Beginners. <http://www.tldp.org/LDP/Bash-Beginners-Guide/Bash-Beginners-Guide.pdf>. 2008

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 4º (2C)	Idioma: Español
Asignatura: 900235 - Procesadores de Lenguajes		Abrev: PL	6 ECTS
Asignatura en Inglés: Language Processors		Carácter: Obligatoria	
Materia: Lenguajes informáticos y procesadores de lenguaje			12 ECTS
Otras asignaturas en la misma materia: Fundamentos de los lenguajes informáticos			6 ECTS
Módulo: Tecnología específica: Computación			
Departamento: Interdepartamental ISIA / SIC		Coordinador: Sierra Rodríguez, José Luis	

Descripción de contenidos mínimos:

- Analizadores léxicos.
- Analizadores sintácticos.
- Comprobación de tipos, semántica estática y restricciones contextuales.
- Traducción y generación de código.
- Máquinas virtuales.
- Optimización de código.
- Herramientas de desarrollo de procesadores de lenguaje

Programa detallado:

1. Introducción a los Procesadores de Lenguaje
2. Análisis Léxico
3. Análisis Sintáctico
4. Procesamiento Dirigido por la Sintaxis
5. Análisis de la Semántica Estática
6. Máquinas Virtuales y Generación de Código

Programa detallado en inglés:

1. Introduction to Language Processors
2. Scanning
3. Parsing
4. Syntax-directed Processing
5. Static Semantic Analysis
6. Virtual Machines and Code Generation

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.
- 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:

- 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.

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:

- Saber desarrollar sistemáticamente un procesador para un lenguaje informático a partir de su especificación (CG2, CG3, CG13, CE_C2)
- Saber especificar formalmente los diferentes aspectos de un procesador para un lenguaje informático (CG2, CT2, CT3, CT5, CE_C2)
- Saber traducir lenguajes informáticos de alto nivel a lenguajes de bajo nivel (CG2, CG3, CG13, CT2, CE_C2)
- Saber utilizar herramientas específicas de generación automática durante la construcción de procesadores de lenguajes informáticos (CG2, CG3, CG13, CT1, CT2, CT3, CT4, CT5, CE_C2)
- Ser capaz de extrapolar los métodos, técnicas y herramientas propios del diseño y la implementación de lenguajes informáticos a otros campos del desarrollo de software (CG2, CG3, CG13, CT2, CT3, CE_C2)
- Ser capaz de planificar, gestionar y desarrollar proyectos de construcción de procesadores de lenguajes informáticos (CG2, CG3, CG13, CT1, CT2, CT3, CT4, CT5, CE_C2)

Evaluación detallada:

La evaluación se llevará a cabo mediante un examen final y mediante la realización de actividades prácticas.

Las actividades prácticas cubrirán aspectos prácticos de la asignatura, tales como el desarrollo de un procesador para un lenguaje informático, resolución de problemas, exposición en público de soluciones, y cualquier otro tipo de actividad práctica que el profesor considere adecuada para la correcta adquisición de las competencias planteadas en dicha asignatura.

Las actividades prácticas deben realizarse durante el curso, en los plazos establecidos para ello por el profesor. Las actividades prácticas no realizadas dentro de los plazos establecidos, no serán recuperables (puntuarán 0 en la nota final de la parte práctica, tanto en la convocatoria ordinaria como en la extraordinaria).

Para aprobar la asignatura será necesario aprobar el examen (puntuación mínima de 5.0). En dicho caso:

- El 30% de la nota se obtendrá mediante la evaluación de las actividades prácticas.
- El 70% de la nota se alcanzará mediante el examen final.

En caso de no aprobar el examen, la nota corresponderá a la obtenida en dicho examen.

Actividades docentes:

Reparto de créditos:	Otras actividades:
Teoría: 3,00	Clases teóricas magistrales.
Problemas: 0,00	Estudio
Laboratorios: 3,00	Realización individual de ejercicios
	Tutorías
	Clases de problemas.
	Realización de exámenes.

Bibliografía:

- M.L. Scott. Programming Language Pragmatics. Third Edition. Elsevier 2009.
- R. Wilhelm; D. Maurer. Compiler Design. Addison-Wesley, 1995.
- A. W. Appel; Modern Compiler Implementation in Java. Cambridge University Press, 1997.
- A.V. Aho, R. Sethi, J.D. Ullman; Compilers. Principles, Techniques and Tools. Addison-Wesley, 1988.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 4º (2C)	Idioma: Español
Asignatura: 900236 - Programación Concurrente		Abrev: PC	6 ECTS
Asignatura en Inglés: Concurrent Programming		Carácter: Obligatoria	
Materia: Programación avanzada		21 ECTS	
Otras asignaturas en la misma materia:			
Métodos algorítmicos en resolución de problemas		9 ECTS	
Programación Declarativa		6 ECTS	
Módulo: Tecnología específica: Computación			
Departamento: Sistemas Informáticos y Computación		Coordinador: Albert Albiol, Elvira María	

Descripción de contenidos mínimos:

Programación con memoria compartida.
Monitores.
Programación distribuida.
Paso de mensajes síncronos y asíncronos.
Protocolos de comunicación.
Especificación de sistemas concurrentes.
Tecnologías para el desarrollo de sistemas concurrentes y distribuidos.

Programa detallado:

1. Introducción a la programación concurrente (conceptos y terminología básicos)
Procesos e hilos; planificación; exclusión mutua; concurrencia y paralelismo, no determinismo, atomicidad, trazas de ejecución y semántica por entrelazamiento; propiedades de seguridad, viveza, justicia e inanición; deadlock y livelock;

2. Programación con memoria compartida
Interferencia y sincronización; esquemas de sincronización: espera activa, semáforos, cerrojos, mutex, variables de condición, monitores, sincronización no-bloqueante; programación concurrente y paralela en Java.

3. Programación con paso de mensajes
Canales y enlaces, sincronía/asincronía, panorama de sistemas de paso de mensajes.

Programa detallado en inglés:

1. Introduction to Concurrent Programming.
Processes and threads; scheduling; mutual exclusion; concurrency and parallelism; non-determinism, atomicity, execution traces and interleaving semantics; safety, liveness, fairness and starvation; deadlock and livelock;

2. Programming with Shared Memory
Interference and synchronization; synchronization schemes: busy wait, semaphores, locks, mutex, condition variables, monitors, non-blocking synchronization; concurrent and parallel programming in Java.

3. Programming with Message Passing
Channels and links, synchrony and asynchrony, message passing systems

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.

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:

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.

CE_C3-Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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. (CG11, CT1, CE_C2, CE_C3)
- Desarrollar las prácticas y ejercicios tomando decisiones sobre su diseño. (CT3, CE_C3)
- Desarrollar y validar programas expresados en diferentes modelos concurrentes. (CG3, CE_C1)
- Evaluar el grado de paralelismo y concurrencia de los algoritmos para elegir el más adecuado. (CT2, CE_C1, CE_C2)
- Manejar en los programas desarrollados mecanismos de sincronización y acceso en exclusión mutua a secciones críticas de código. (CG3, CG12, CG13)
- Realizar en equipo las prácticas de programación concurrente. (CT1)
- Resolver ejercicios de programación concurrente analizando el problema y diseñando la solución. (CT2, CT3, CE_C1, CE_C2)
- Utilizar herramientas informáticas sobre sistemas operativos concretos para desarrollar programas concurrentes. (CG3, CG13, CE_C1, CE_C2, CE_C3)

Evaluación detallada:

Convocatoria ordinaria

Prácticas: 20% de la nota (se conserva hasta la convocatoria extraordinaria)
Examen final: 80% de la nota

Convocatoria extraordinaria

Prácticas: 20% de la nota
Examen final: 80% de la nota

Actividades docentes:

Reparto de créditos:
Teoría: 3,00
Problemas: 1,50
Laboratorios: 1,50

Otras actividades:
Clases magistrales y clases participativas (2 horas a la semana en aula de teoría).
Clases de problemas (2 horas cada dos semanas en aula de teoría)
Clases de problemas/prácticas (2 horas cada dos semanas en aula de informática).
Tutorización personalizada en los horarios establecidos.

Bibliografía:

Bibliografía básica

Gregory R. Andrews. Foundations of Multithreaded, Parallel and Distributed Programming, Addison Wesley.
D. Lea, "Programación concurrente en Java. Principios y patrones de diseño". 2ª edición, Addison Wesley, 2001.

Bibliografía complementaria

M. Ben-Ari, "Principles of Concurrent and Distributed Programming". 2ª edición, Addison - Wesley, 2006.
J. Magee y J. Kramer, "Concurrency. State Models and Java Programming". Wiley 2006.
M. Herlihy y N. Shavit, "The Art of Multiprocessor Programming". Elsevier, 2008.
T. Rauber y G. Rünger, "Parallel Programming: for Multicore and Cluster Systems". Springer 2010.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 4º (1C)	Idioma: Español
Asignatura: 901964 - Inteligencia Artificial I		Abrev: IA1	4,5 ECTS
Asignatura en Inglés: Artificial Intelligence I		Carácter: Obligatoria	
Materia: Inteligencia artificial		9 ECTS	
Otras asignaturas en la misma materia: Inteligencia Artificial II		4,5 ECTS	
Módulo: Tecnología específica: Computación			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Díaz Agudo, Mª Belén	

Descripción de contenidos mínimos:

Introducción a la inteligencia artificial.
Inteligencia artificial distribuida y sistemas multiagente.
Introducción a la visión artificial y la robótica.
Estrategias de búsqueda.
Algoritmos genéticos.
Representación básica del conocimiento y razonamiento.

Programa detallado:

1. Introducción a la IA. Principios fundamentales. Evolución histórica. Aplicaciones. Retos y características de los problemas de la IA. Agentes, visión artificial y robótica. Tipos de IA según el conocimiento y métodos de razonamiento.
2. Resolución de problemas con búsqueda. Representación de problemas. Algoritmos de búsqueda heurística. Búsqueda local. Optimización. Algoritmos genéticos. Aprendizaje por refuerzo.
3. Representación básica del conocimiento y razonamiento. Propiedades de la representación del conocimiento. Tipos de razonamiento (genera y prueba, análisis de medios y fines, reducción de problemas). Sistemas de producción. Sistemas basados en casos. Sistemas recomendadores.

Programa detallado en inglés:

1. Introduction to AI. Fundamental aspects. Historical evolution. AI Applications. AI Conundrums. Agents, computer vision and robotics. Cognitive AI, other types of AI and introductory review of different reasoning methods.
2. Problem solving and search Problem representation and state space Heuristic search algorithms. Local optimization algorithms. Genetic algorithms. Reinforcement Learning.
3. Basic representation of knowledge and reasoning. Properties of knowledge representation Rule-based systems. Case based reasoning. Recommender systems.

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.
CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

Específicas:

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.

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:

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMÁTICA

- Analizar las características de un problema dado y determinar si es susceptible de ser resuelto mediante técnicas de búsqueda. (CG11, CT1, CT2)
- Decidir en base a criterios racionales la técnica de IA más apropiada para resolver un problema y saber aplicarla. (CG11, CT2)
- Comprender las técnicas de representación de problemas en IA y la búsqueda en el espacio de estados. (CG23, CT2)
- Construir sistemas capaces de resolver problemas mediante técnicas de IA (CG23, CT3, CE_C4)
- Conocer qué es la Inteligencia Artificial, sus fundamentos como disciplina científico-técnica y su historia. (CE_C4)
- Entender la IA como conjunto de técnicas para el desarrollo de sistemas informáticos que exhiben comportamientos reactivos, deliberativos y/o adaptativos. (CE_C4)
- Conocer distintas aplicaciones reales de la IA. Explorar y analizar soluciones actuales basadas en técnicas de IA. (CG23)
- Conocer las técnicas más representativas de búsqueda en un espacio de estados y analizar su eficiencia en tiempo y espacio. (CE_C4)
- Entender la utilidad de la representación del conocimiento basada en reglas y aplicarla a la construcción de sistemas de producción. (CG23, CT4, CE_C5)
- Analizar las técnicas de representación del conocimiento y seleccionar la más apropiada para desarrollar un sistema inteligente. (CT2, CE_C5)

Evaluación detallada:

Calificación final = $0,7 \cdot \text{NFE} + 0,3 \cdot \text{NFP}$, siendo $\text{NFE} \geq 4$

NFE: nota final de exámenes.

NFP: nota final de prácticas. Se obtiene calculando la media ponderada de las prácticas y otros ejercicios propuestos durante el curso.

No habrá posibilidad de entregar las prácticas en la convocatoria extraordinaria.

Actividades docentes:

Reparto de créditos:

Teoría: 2,00

Problemas: 1,00

Laboratorios: 1,50

Otras actividades:

Clases teóricas, clases de problemas y laboratorios.

Bibliografía:

Russell, S., Norvig, P. Artificial Intelligence: A Modern Approach, third revised edition. Pearson, 2016.

Russell, S., Norvig, P. Inteligencia Artificial: Un enfoque moderno, segunda edición. Pearson Educación, 2004.

Pajares, G., Santos, M. Inteligencia Artificial e Ingeniería del Conocimiento. RA-MA, 2005.

Winston, P. (1993). Artificial Intelligence (3rd ed.). Addison-Wesley.

Ficha docente guardada por última vez el 17/07/2020 12:31:00 por el departamento: Ingeniería del Software e Inteligencia Artificial

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 4º (2C)	Idioma: Español
Asignatura: 901965 - Inteligencia Artificial II		Abrev: IA2	4,5 ECTS
Asignatura en Inglés: Artificial Intelligence II		Carácter: Obligatoria	
Materia: Inteligencia artificial		9 ECTS	
Otras asignaturas en la misma materia: Inteligencia Artificial I		4,5 ECTS	
Módulo: Tecnología específica: Computación			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Sánchez Ruiz-Granados, Antonio Alejandro	

Descripción de contenidos mínimos:

Representación avanzada del conocimiento y razonamiento.
Aprendizaje automático.
Procesamiento del lenguaje natural.

Programa detallado:

1. Representación avanzada del conocimiento y razonamiento
 - Sistemas basados en lógica
 - Representaciones estructuradas: redes semánticas y ontologías
2. Aprendizaje automático
 - Aprendizaje supervisado
 - Aprendizaje no supervisado
3. Procesamiento de lenguaje natural
 - Modelos de lenguaje
 - Clasificación de documentos y recuperación de información
 - Métodos basados en gramáticas

Programa detallado en inglés:

1. Advanced knowledge representation and reasoning
 - Logic-based systems
 - Structured representations: semantic networks and ontologies
2. Machine learning
 - Supervised learning
 - Unsupervised learning
3. Natural language processing
 - Language models
 - Document classification and information retrieval
 - Grammar-based methods

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.

CG23-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

Específicas:

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.

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.

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:

Decidir en base a criterios racionales la técnica de IA más apropiada para resolver un problema y saber aplicarla. (CG11, CT2)

Construir sistemas capaces de resolver problemas mediante técnicas de IA (CG23, CT3, CE_C4)

Entender la IA como conjunto de técnicas para el desarrollo de sistemas informáticos que exhiben comportamientos reactivos, deliberativos y/o adaptativos. (CT5, CE_C4)

Conocer distintas aplicaciones reales de la IA. Explorar y analizar soluciones actuales basadas en técnicas de IA. (CG23)

Aplicar los aspectos de representación basada en la lógica y mecanismos de inferencia, mediante técnicas y herramientas de programación lógica. (CG23, CE_C5)

Analizar las técnicas de representación del conocimiento y seleccionar la más apropiada para desarrollar un sistema inteligente. (CT2, CE_C5)

Entender las técnicas básicas de aprendizaje automático y sus posibilidades de aplicación. (CE_C7)

Resolver problemas en los que sea necesario el aprendizaje automático. Seleccionar la técnica más adecuada y analizar resultados. (CT2, CT3, CE_C7)

Evaluación detallada:

Calificación final = $0,7 * NFE + 0,3 * NFP$, siendo $NFE \geq 5$

NFE: nota final de exámenes.

NFP: nota final de prácticas. Se obtiene calculando la media ponderada de las prácticas y otros ejercicios propuestos durante el curso.

No habrá posibilidad de entregar las prácticas en la convocatoria extraordinaria.

Actividades docentes:

Reparto de créditos:

Teoría: 2,00

Problemas: 1,00

Laboratorios: 1,50

Otras actividades:

Clases teóricas, clases de problemas y laboratorios.

Bibliografía:

- Berthold, M. R., Borgett, C., Höppner, F., Klawonn, F. Guide to Intelligent Data Analysis. Springer, 2010.
- Russell, S., Norvig, P. Artificial Intelligence: A Modern Approach, 4ª edición. Pearson, 2020
- Russell, S., Norvig, P. Inteligencia Artificial: Un enfoque moderno, segunda edición. Pearson Educación, 2004.
- James, G., Witten, D., Hastie, T., Tibshirani, R. An introduction to statistical learning with applications in R. Springer, 2017.
- Jurafsky, D., Martin, J.H. Speech and Language Processing, 2ª ed. Prentice Hall, 2008.
- Allemang, D., Hendler, J.A. Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL, 2ª edición Elsevier Science & Technology, 2011.
- Garreta, R., Moncecchi, G. Learning scikit-learn: Machine Learning in Python . Packt Publishing, 2013.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 5º (1C)	Idioma: Español
Asignatura: 900265 - Ética, legislación y profesión		Abrev: ELP	6 ECTS
Asignatura en Inglés: Ethics, Legislation and Profession		Carácter: Obligatoria	
Materia: Ética, legislación y profesión		6 ECTS	
Otras asignaturas en la misma materia: No hay			
Módulo: Materias comunes a la rama de la informática			
Departamento: Interdepartamental ACYA / ISIA		Coordinador: Román Navarro, Sara	

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

Tema 2. Privacidad:

- Vigilancia
- Redes sociales.
- GDPR
- Criptografía.
- Filtraciones

Tema 3. Derechos digitales:

- Comunidades online.
- Libertad de expresión en internet.
- Anonimato. Censura. Transparencia. Neutralidad de red.

Tema 4. Brecha digital y Privilegios:

- Privilegios y desigualdad
- Brecha digital (por edad, género, raza, nivel económico...)
- Cyber-bullying, trolls, acoso en redes sociales
- Sesgos en el software

Tema 5. Derechos de autor:

- Licencias.
- Software libre.
- Hardware libre

Tema 6. Cultura libre:

- Procomún.
- Copia privada.
- P2P.
- Patentes

Tema 7. Delitos y responsabilidad informática:

- Ética hacker.
- Uso responsable de la tecnología. Responsabilidad, seguridad y control.
- Sistemas distribuidos.
- Blockchain

Tema 8. Profesión:

- Qué es ser informático/a.
- Opciones profesionales.
- Tipos de empleadores.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- Investigación. Emprendimiento.
- Búsqueda de trabajo y procesos de selección.
- Códigos éticos profesionales.

Programa detallado en inglés:

Lesson 1: Introduction to Ethics and Law

Lesson 2: Privacy. Surveillance. Social networks.GDPR. Cryptography. Leaks.

Lesson 3: Digital rights. Online communities. Freedom of expression in internet. Anonymity. Censorship. Transparency. Net Neutrality.

Lesson 4: Privileges and Inequality, Digital divide (age, gender etc.), Cyberbullying, Software design bias.

Lesson 5: Copyright. Licenses. Free software. Free Hardware.

Lesson 6: Free culture. Commons. Right to private copy ("copia privada"). P2P. Patents.

Lesson 6: Computer responsibility and crimes. Hacker ethics. Responsible use of technology. Responsibility, security and control..Distributed systems. Blockchain.

Lesson 7: Profession. What is a computer scientist. Professional options. Types of employers. Research. Entrepreneurship. Job search and selection processes. Professional ethical codes.

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)

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)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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: parte teórica (35%), que consiste en una evaluación sobre los contenidos desarrollados a lo largo de la asignatura, parte de conferencias (10%) y una parte liberatoria (25%) sobre Impacto Social de las TICs que es convalidable a través de la superación de pruebas realizadas a lo largo del cuatrimestre (realización de un Trabajo de Impacto Social en grupo). El 30% de otras actividades reflejará la evaluación de la participación y el trabajo del/la alumno/a en las diferentes actividades diarias de clase, tales como debates, foros, ejercicios, entre otras.

La asignatura ha tenido una serie de conferencias de expertas/os invitadas/os en los últimos cursos que siempre que ha sido posible se han grabado. Durante este curso haremos uso de esas conferencias grabadas para la calificación correspondiente al 10% de conferencias.

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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 5º (1C)	Idioma: Español
Asignatura: 900266 - Ampliación de Sistemas Operativos y Redes		Abrev: ASOR	6 ECTS
Asignatura en Inglés: Advanced Operating Systems and Networks		Carácter: Obligatoria	
Materia: Sistemas Operativos y Redes Avanzados		6 ECTS	
Otras asignaturas en la misma materia: No hay			
Módulo: Complementario			
Departamento: Arquitectura de Computadores y Automática		Coordinador: Santiago Montero, Rubén Manuel	

Descripción de contenidos mínimos:

- Administración con lenguajes de script.
- Diseño e implementación de aplicaciones basadas en servicios del SO.
- Utilidades de monitorización.
- Introducción a sistemas distribuidos.
- Internet de nueva generación (Ipv6).
- Protocolos de encaminamiento.
- Protocolos y servicios de red avanzados.
- Programación con sockets.

Programa detallado:

1. AMPLIACIÓN DE REDES

1.1. Revisión del protocolo IPv4. DHCP

- Revisión de IPv4, ARP e ICMP
- Configuración dinámica (DHCP)

1.2. Conceptos avanzados del protocolo TCP

- Revisión de TCP
- Control de errores y temporizadores de retransmisión
- Control de flujo y congestión
- Ajuste de parámetros

1.3. Servicios de red

- Traducción de direcciones de red (NAT)
- Filtrado de paquetes
- Sistema de nombres de dominio (DNS)

1.4. Internet de nueva generación: IPv6

- Comparación con IPv4
- Direccionamiento
- Formato del datagrama
- ICMPv6

1.5. Encaminamiento en Internet

- Sistemas autónomos
- RIP, OSPF y BGP

2. AMPLIACIÓN DE SISTEMAS OPERATIVOS

2.1. Introducción

- Estructura y recursos del sistema
- Arquitectura del sistema operativo Linux
- Interfaz de llamadas al sistema
- Códigos de error y gestión de errores
- Llamadas al sistema y funciones de biblioteca

2.2. Gestión del sistemas de ficheros

- Arquitectura del sistema de ficheros
- Manejo de ficheros ordinarios
- Manejo de directorios

2.3. Gestión de procesos y memoria

- Estructura e información de procesos

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- Ejecución de programas
- Control de procesos
- Gestión de señales y temporizadores
- Comunicación mediante tuberías (pipes)

2.4. Programación con sockets

- Modelo cliente-servidor
- Tipos de sockets
- Gestión y uso de sockets

PRÁCTICAS

1.1. DHCP

1.2. TCP

1.3. DNS

1.4. IPv6

1.5. RIP

2.1. Programación shell

2.2. Entorno de desarrollo

2.3. Sistema de ficheros

2.4. Procesos

2.5. Tuberías

2.6. Sockets

Programa detallado en inglés:

1. ADVANCED NETWORKS

1.1. IPv4 Review. DHCP

- Review of IPv4, ARP and ICMP
- Dynamic configuration (DHCP)

1.2. TCP Advanced Concepts

- TCP review
- Error control and retransmission timers
- Flow and congestion control
- Parameter tuning

1.3. Network Services

- Network Address Translation (NAT)
- Packet filtering
- Domain Name System (DNS)

1.4. Next Generation Internet: IPv6

- Comparison with IPv4
- Datagram format
- Addressing
- ICMPv6

1.5. Internet Routing

- Autonomous Systems
- RIP, OSPF and BGP

2. ADVANCED OPERATING SYSTEMS

2.1. Introduction

- Structure and system resources
- Linux operating system architecture
- System call interface
- Error codes and error management
- System calls and library functions

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



2.2. File System Management

- File system architecture
- Regular file management
- Directory management

2.3. Process and Memory Management

- Process structure and information
- Program execution
- Process control
- Signal and timer management
- Communication with pipes

2.4. Socket Programming

- Client-server model
- Socket types
- Socket management and use

LABORATORY

1.1. DHCP

1.2. TCP

1.3. DNS

1.4. IPv6

1.5. RIP

2.1. Shell programming

2.2. Development environment

2.3. File systems

2.4. Processes

2.5. Pipes

2.6. Sockets

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.
- 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.
- 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:

- 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 los requisitos de funcionamiento de una red de computadores y diseñar la estructura y servicios adecuados (CG10, CG16, CT1, CT3)
- Comprensión del funcionamiento de los algoritmos de encaminamiento y configuración de los mismos (CG10, CG16, CT2, CT3)
- Desarrollo de aplicaciones distribuidas (CG3, CG10, CG15, CG16, CG19, CT2, CT3, CT5)
- Desarrollo de aplicaciones específicas para la gestión de sistemas de ficheros, gestión de memoria y sincronización y planificación de procesos (CG3, CG10, CG15, CT2, CT3)
- Desarrollo de prácticas en equipo (CT1, CT4)

Evaluación detallada:

Asistencia al laboratorio y realización de prácticas durante el cuatrimestre = 10%
Examen final teórico-práctico (laboratorio y aula) = 90% (50% teoría y 40% práctica). Tanto para la convocatoria ordinaria como para la extraordinaria. Es necesario obtener al menos un 40% de la nota en cada parte para aprobar.
La nota obtenida durante el curso en las prácticas se conserva para la convocatoria extraordinaria. En la convocatoria extraordinaria no existirá la opción de entregar nuevas prácticas

Actividades docentes:

Reparto de créditos:	Otras actividades:
Teoría: 3,00	No tiene
Problemas: 0,00	
Laboratorios: 3,00	

Bibliografía:

- A.S. Tanenbaum and A.S. Woodhull.; Operating Systems Design and Implementation. Prentice Hall. 3rd Edition; Enero 2006
- Robert Love; Linux Kernel Development. Addison-Wesley. 3rd Edition. Julio 2010
- F. Márquez García. "UNIX. Programación Avanzada". 3ª Edición. Editorial RA-MA, 2004
- L. Parziale. "TCP/IP Tutorial and Technical Overview". 8th edition. IBM RedBooks. 2006.
- Q. Li. "IPv6 Core Protocols Implementation". 1st edition. Morgan Kaufmann Publishers. 2005.
- F. Halsall. "Redes de Computadores e Internet". 5ª edición. Addison-Wesley. 2006.

Ficha docente guardada por última vez el 06/07/2021 15:03:00 por el usuario: Vic. Estudios

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: 5º (1C)	Idioma: Español
Asignatura: 900267 - Arquitectura de Computadores		Abrev: AC	6 ECTS
Asignatura en Inglés: Computer Architecture		Carácter: Obligatoria	
Materia: Tecnología y Arquitectura de Computadores		12 ECTS	
Otras asignaturas en la misma materia: Tecnología y organización de computadores		6 ECTS	
Módulo: Complementario			
Departamento: Arquitectura de Computadores y Automática		Coordinador: Tirado Fernández, Francisco	

Descripción de contenidos mínimos:

Paralelismo a nivel de instrucción y a nivel thread.
Introducción a los multiprocesadores: Problemas de sincronización.
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.
- Ejemplos: Evolución de arquitecturas Intel
- Multithreading: concepto y tipos
- Ejemplos de arquitecturas multithread

Módulo 3. Paralelismo a nivel de datos.

- Concepto de arquitectura vectorial
- Instrucciones SIMD para procesamiento multimedia
- Unidades para procesamiento gráfico (GPUs)
- Paralelismo a nivel bucle: vectorización

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
- o Module 2. Instruction-level parallelism (ILP) and multithreading.
Basic compilation techniques
Dynamic instruction scheduling: dependences and register renaming
Branch prediction

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Speculative execution
Multiple issue techniques
Limits of ILP
Architecture of superscalar out-of-order processors
Examples: Evolutions of Intel architectures
Multithreading: concept and types
Examples of multithread architectures

- o Module 3. Data-level parallelism
Vector architecture
SIMD instruction set extensions for multimedia
Graphics processing units (GPUs)
Loop-level parallelism: vectorization
- 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:

No tiene

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 la terminología propia de la arquitectura de computadores en lengua inglesa y manejar fuentes bibliográficas en dicha lengua (CT1)
- Resolver problemas de Arquitectura de Computadores seleccionando la solución más adecuada entre las posibles. (CT2, CT3)

Evaluación detallada:

A mitad del cuatrimestre habrá un examen parcial opcional y no liberatorio, cuyo peso en la nota de la asignatura será del 20% para los alumnos que deseen realizarlo.

Convocatoria ordinaria: Examen final obligatorio y escrito.

La nota de esta convocatoria será la mayor de la dos siguientes:

-Nota del examen parcial x 0,2 + Nota examen final x 0,7 + Nota entrega ejercicios x 0,1

-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:

Otras actividades:

Fecha: ____ de _____ de _____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMATICA

Teoría: 4,50
Problemas: 1,50
Laboratorios: 0,00

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", 5ª Ed., Morgan-Kaufmann, 2012.
- Culler, D., Singh, J., "Parallel Computer Architecture: A Hardware/Software Approach", Morgan Kaufmann, 1999

Complementaria:

- Dubois, M., Annavaram, M., Stenström, P. "Paraller Computer Organization and Design", Cambridge University Press, 2012
- González, A., Latorre, F., Magklis, G., "Processor Microarchitecture - An Implementation Perspective", Morgan & Claypool Publishers, 2011.

Ficha docente guardada por última vez el 07/07/2021 11:18:00 por el usuario: Vic. Estudios

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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: 5º (1C)	Idioma: Español
Asignatura: 900268 - Desarrollo de sistemas interactivos	Abrev: DSI	6 ECTS
Asignatura en Inglés: Interactive Systems Development	Carácter: Optativa	
Materia: Interacción persona-computador		6 ECTS
Otras asignaturas en la misma materia: No hay		
Módulo: Tecnología específica: Computación		
Departamento: Ingeniería del Software e Inteligencia Artificial		
Coordinador: Hervás Ballesteros, Raquel		

Descripción de contenidos mínimos:

Fundamentos de la Interacción persona-computador.
Modelos y metáforas de interacción.
Diseño de implementación de aplicaciones interactivas.
Evaluación de sistemas interactivos.
Interfaces a Bases de Datos y Sistemas de Información.
Interfaces inteligentes.
Accesibilidad e interfaces para usuarios con necesidades especiales.

Programa detallado:

- 1.- Introducción a la Interacción Persona-Ordenador (IPO). Usabilidad.
- 2.- Modelos y metáforas de interacción.
- 3.- Diseño e implementación de aplicaciones interactivas. Diseño centrado en el usuario.
- 4.- Evaluación de sistemas interactivos.
- 5.- Conceptos avanzados.

Programa detallado en inglés:

- 1.- Introduction to Human-Computer Interaction (HCI). Usability.
- 2.- Interaction models and metaphors.
- 3.- Design and development of interactive applications: User-centered design.
- 4.- Assessment of interactive systems.
- 5.- Advanced concepts

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:

CE_C6-Capacidad para desarrollar y evaluar sistemas interactivos y de presentación de información compleja y su aplicación a la resolución de problemas de diseño de interacción persona computadora.

CE_TI3-Capacidad para emplear metodologías centradas en el usuario y la organización para el desarrollo, evaluación y gestión de aplicaciones y sistemas basados en tecnologías de la información que aseguren la accesibilidad, ergonomía y usabilidad de los sistemas.

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:

Aplicar el Diseño Centrado en Usuario trabajando en equipo para diseñar un gran sistema interactivo con un interfaz innovadora. (CG21, CT1, CT2, CT3, CT4, CT5, CE_C6, CE_TI3)

Argumentar decisiones de diseño en el desarrollo de interfaces. (CG21, CT1, CT2)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- Comprender el impacto del diseño de interfaces en la Experiencia de Usuario. (CG21, CT2, CT3, CT5)
- Conocer el campo de la Usabilidad y su repercusión en la aceptabilidad de los sistemas informáticos. (CG21, CT5)
- Conocer los distintos procesos de evaluación de usabilidad y aceptabilidad de un sistema. (CG21, CT2, CT3, CT5, CE_C6, CE_TI3)
- Conocer los procesos de Diseño Centrado en Usuario (CG21, CT2, CE_C6, CE_TI3)
- Conocer y combinar los distintos patrones de diseño de interfaces. (CG21, CE_C6, CE_TI3)
- Distinguir entre los conceptos de Funcionalidad, Usabilidad y Experiencia de Usuario (CG21, CT3, CT5, CE_TI3)
- Dominar el vocabulario de modelos y metáforas de interacción. (CG21, CE_C6)
- Entender los principios de la disciplina de la Interacción Persona-Ordenador (CG21, CT5)

Evaluación detallada:

Para aprobar la asignatura, el estudiante deberá superar las pruebas de evaluación continua (ejercicios y prácticas) y una prueba final (examen o proyecto).

La nota final de la asignatura se calculará en base a la siguiente fórmula: $0.8*NF + 0.2*NP$

Siendo:

* NF: nota del examen o proyecto final.

* NP: nota de las prácticas y ejercicios propuestos durante el curso.

Para aprobar la asignatura es requisito necesario obtener al menos un 5 en ambos apartados. Para aprobar mediante proyecto final, es necesario asistir al 80% de las clases.

Las calificaciones obtenidas en cualquiera de las pruebas se mantendrán para la convocatoria extraordinaria. Los ejercicios y prácticas suspensos podrán entregarse de nuevo antes de dicha convocatoria. En caso de suspender o no presentar el proyecto en la convocatoria ordinaria, la prueba final en la convocatoria extraordinaria será un examen.

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:

* About face 3 : the essentials of interaction design. Alan Cooper, Robert Reimann, and Dave Cronin. Wiley, cop. 2007

* Usability Engineering. Jakob Nielsen. AP Professional, 1993.

* Handbook of Usability Testing. Jeff Rubin, Dana Chisnell. Wiley Publishing. 2008.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5º (1C)	Idioma: Español
Asignatura: 900240 - Programación paralela para móviles y multicores Asignatura en Inglés:	Abrev: PP Carácter: Optativa	6 ECTS
Materia: Complementos de computadores	12 ECTS	
Otras asignaturas en la misma materia: Seguridad en redes	6 ECTS	
Módulo: Optativo		
Departamento: Arquitectura de Computadores y Automática		Coordinador: Prieto Matias, Manuel

Descripción de contenidos mínimos:

No tiene

Programa detallado:

Tema 1: Introducción a la Programación paralela

- 1.1 Motivación. ¿por qué la computación paralela?. Prestaciones
- 1.2 Memoria Compartida vs Memoria Distribuida. Paradigmas de programación.
- 1.3 Sistemas multicore y dispositivos móviles. Arquitecturas.

Tema 2: Programación paralela en dispositivos móviles

- 2.1 Introducción a la plataforma Android.
- 2.2 Hilos, controladores.
- 2.3 Render Script
- 2.4 MARE (Multicore Asynchronous Runtime Environment)
- 2.5 OpenCL para Android
- 2.6 Casos Prácticos

(Prácticas tema 2)

Tema 3- Programación paralela con OpenMP.

- 3.1 Introducción a OpenMP. Evolución de las diferentes versiones.
- 3.2 Directivas OpenMP
- 3.3 Clausulas y Funciones de OpenMP
- 3.4 Ejecución múltiples hebras.
- 3.5 Variables de Entorno
- 3.6 Dependencia de datos, Interferencia. Sincronización.
- 3.7 Planificación y Transformación de bucles.
- 3.8 Rendimiento en OpenMP
- 3.9 Casos prácticos

(Prácticas Tema 3)

Tema 4 Nuevos paradigmas y herramientas de programación para Multicores

- 4.1 Entorno Intel Parallel Composer
- 4.2 Intel TBB/ ArBB
- 4.3 Cilk, Cilk+ .
- 4.4 Comparación de estos paradigmas con OpenMP

(Prácticas Tema 4)

Programa detallado en inglés:

No tiene

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.
- 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:

- 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.

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:

Aprender y practicar los conceptos complementarios de Programación paralela para móviles y multicores

Evaluación detallada:

Dos modalidades

Evaluación Continua: Prácticas (50%) + Exposición y entrega de trabajos (50%) o bien

Calificación Examen final: 100% examen final.

Actividades docentes:

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

Bibliografía:

- <http://developer.android.com/guide/topics/renderscript/index.html>
- Pro Android Apps Performance Optimization, Herv Guihot, Apress. ISBN-10: 1430239999
- <https://code.google.com/p/aopencl/>
- <https://developer.qualcomm.com/mobile-development/maximize-hardware/parallel-computing-mare>
- Introduction to Parallel Programming, Peter Pacheco (Autor)
- Using OpenMP: Portable Shared Memory Parallel Programming (Scientific and Engineering Computation) Barbara Chapman (Autor), Gabriele Jost (Autor), Ruud van der Pas (Autor), MIT press 208.
- Structured Parallel Programming: Patterns for Efficient Computation. Michael McCool (Author), James Reinders (Author), Arch RobisoMK, 2012.
- Intel Threading Building Blocks : outfitting C++ for multi-core processor parallelism. James Reindes (Author)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 900241 - Seguridad en redes Asignatura en Inglés: Network security	Abrev: SER Carácter: Optativa	6 ECTS
Materia: Complementos de computadores	6 ECTS	
Otras asignaturas en la misma materia: No hay		
Módulo: Optativo		
Departamento: Arquitectura de Computadores y Automática		Coordinador: Moreno Vozmediano, Rafael

Descripción de contenidos mínimos:

Redes avanzadas
Comunicaciones seguras
Protección redes y sistemas en red

Programa detallado:

TEORÍA

Módulo 1. Introducción a la seguridad

- 1.1. Introducción
- 1.2. Vulnerabilidades y amenazas
- 1.3. Anatomía de un ataque
- 1.4. Servicios y mecanismos de seguridad
- 1.5. Aspectos legales y éticos

Módulo 2. Seguridad en las comunicaciones

- 2.1. Introducción a la criptografía
- 2.2. Criptografía de clave secreta
- 2.3. Funciones resumen
- 2.4. Criptografía de clave pública
- 2.5. Certificados digitales y modelos de confianza
- 2.6. Aplicaciones para comunicaciones seguras

Módulo 3. Seguridad de servidores de Internet

- 3.1. Seguridad Web
- 3.2. Seguridad del correo electrónico
- 3.3. Seguridad DNS

Módulo 4. Ataques y mecanismos de seguridad en redes

- 4.1. Vulnerabilidades en protocolos de red y ataques
- 4.2. Cortafuegos
- 4.3. Detección de intrusos
- 4.4. Conexiones de red seguras
- 4.5. Seguridad en redes inalámbricas

PRÁCTICAS

Módulo 2. Seguridad en las comunicaciones

- 2.1. Criptografía de clave secreta y funciones resumen (OpenSSL y GnuPG)
- 2.2. Criptografía de clave pública (OpenSSL y GnuPG)
- 2.3. Certificados digitales (OpenSSL y GnuPG)

Módulo 3. Seguridad de servidores de Internet

- 3.1. Configuración de un servidor web seguro
- 3.2. Seguridad de e-mail (S/MIME)
- 3.3. Seguridad en DNS (DNSSEC)

Módulo 4. Ataques y mecanismos de seguridad en redes

- 4.1. Ataques a protocolos Ethernet y ARP
- 4.2. Ataques a protocolos de red y transporte
- 4.3. Cortafuegos (iptables)
- 4.4. Detección de intrusos (snort)
- 4.5. Conexiones de red seguras (IPsec y OpenSSH)

Programa detallado en inglés:

THEORY

Module 1. Introduction to security

- 1.1. Introduction

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- 1.2. Vulnerabilities and threats
- 1.3. Anatomy of an attack
- 1.4. Security services and mechanisms
- 1.5. Ethical and legal aspects
- Module 2. Communication security
- 2.1. Introduction to cryptography
- 2.2. Secret key cryptography
- 2.3. Hash functions
- 2.4. Public key cryptography
- 2.5. Digital certificates and trust models
- 2.6. Applications for secure communications

Module 3. Internet server security

- 3.1. Web security
- 3.2. E-mail security
- 3.3. DNS security

Module 4. Network attacks and security mechanisms

- 4.1. Network protocol vulnerabilities and attacks
- 4.2. Firewalls
- 4.3. Intrusion detection
- 4.4. Secure network connections
- 4.5. Wireless network security

LABORATORY

Module 2. Communication security

- 2.1. Secret key cryptography and hash functions (OpenSSL and GnuPG)
- 2.2. Public key cryptography (OpenSSL and GnuPG)
- 2.3. Digital certificates (OpenSSL and GnuPG)

Module 3. Internet server security

- 3.1. Secure web server configuration (Apache)
- 3.2. Secure E-mail (S/MIME)
- 3.3. Secure DNS (DNSSEC)

Module 4. Network attacks and security mechanisms

- 4.1. Ethernet and ARP protocol attacks (hping3 and nmap)
- 4.2. Network and transport protocol attacks (hping3 and nmap)
- 4.3. Firewalls (iptables)
- 4.4. Intrusion detection (snort)
- 4.5. Secure network connections (IPsec and OpenSSH)

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.
- 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)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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:

La evaluación consta de dos partes:

- Examen final (60% de la nota). Se realizará un examen final en la convocatoria ordinaria y otro en la convocatoria extraordinaria. Estos exámenes se realizarán preferiblemente de forma presencial en el aula, pero si fuese necesario, se podrían realizar de forma no presencial (online). Ambos exámenes podrán incluir cuestiones teóricas, ejercicios, supuestos prácticos, o cuestiones relacionadas con las prácticas de laboratorio.
- Prácticas de laboratorio (40% de la nota). Se valorará la asistencia al laboratorio y la realización de las prácticas con sus correspondientes entregas. En caso de que la docencia fuese no presencial, las prácticas se realizarán en casa y no se tendrá en cuenta la asistencia al laboratorio. Esta nota se consigue a lo largo del cuatrimestre y por lo tanto no se podrá recuperar en ninguna convocatoria.

Para hacer media entre ambas partes es necesario obtener al menos 4 puntos sobre 10 en la nota del examen final. De lo contrario la convocatoria se consideraría suspensa. La nota final se calcula de la siguiente forma:

Si Nota_Examen \geq 4

$$\text{Nota_Final} = 0,6 \times \text{Nota_Examen} + 0,4 \times \text{Nota_Prácticas}$$

Si Nota_Examen $<$ 4

$$\text{Nota_Final} = \text{Nota_Examen}$$

Actividades docentes:

Reparto de créditos:

Teoría: 3,60

Problemas: 0,00

Laboratorios: 2,40

Otras actividades:

No tiene

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

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 900243 - Aprendizaje automático y big data Asignatura en Inglés: Machine learning and big data		Abrev: AA Carácter: Optativa	6 ECTS
Materia: Complementos de ingeniería de software e inteligencia artificial		42 ECTS	
Otras asignaturas en la misma materia: Ingeniería de Comportamientos Inteligentes Ingeniería de sistemas basados en el conocimiento Ingeniería web Programación Competitiva Programación evolutiva Testing de Software		6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Díaz Esteban, Alberto	

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

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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 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:

En ambas convocatorias (ordinaria y extraordinaria) la realización de las prácticas es obligatoria. Además, es necesaria la realización en grupo de un proyecto y su defensa individual.

La nota final se calculará de acuerdo a los siguientes porcentajes:

- Defensa del proyecto: 20% de la nota.
- Trabajo escrito sobre el proyecto: 80% de la nota.

Existe la posibilidad de entregar las prácticas en la convocatoria extraordinaria, manteniéndose para la convocatoria extraordinaria las calificaciones de las prácticas aprobadas durante el curso.

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:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5º (A)	Idioma: Español
Asignatura: 900245 - Ingeniería de sistemas basados en el conocimiento Asignatura en Inglés: Knowledge Based Systems	Abrev: ISBC Carácter: Optativa	6 ECTS
Materia: Complementos de ingeniería de software e inteligencia artificial	24 ECTS	
Otras asignaturas en la misma materia: Aprendizaje automático y big data Ingeniería web Programación evolutiva	6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo		
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Díaz Agudo, Mª Belén

Descripción de contenidos mínimos:

No tiene

Programa detallado:

- Introducción a los SBC. Estructura y Ciclo de Vida de los SBC. Desarrollo de SBC. Ingeniería del Conocimiento. Modelo de desarrollo de sistemas basados en conocimiento. Herramientas. Arquitectura de un SBC. Análisis de Viabilidad. Especificación de requisitos. Diseño preliminar. Paradigmas de razonamiento.
- Paradigmas de representación e inferencia de conocimiento:
 - Razonamiento basado en casos. Introducción al razonamiento basado en casos. Dominios de aplicación de los sistemas CBR. Recuperación de casos. Ejemplos de sistemas CBR. Aplicaciones. Herramientas. jCOLIBRI. Sistemas de Recomendación de productos.
 - Razonamiento Textual. CBR Textual usando técnicas de IR (Information Retrieval). Sistemas de Acceso Inteligente a la Información. Sistemas de clasificación automática de documentos.
 - Razonamiento con ontologías. Introducción a las Ontologías (desarrollo y aplicaciones) . Lenguajes de representación de ontologías. Web Semántica. Protege - OWL
- IA en entornos de simulación Introducción a los agentes software. Definición. Clasificación. Agentes y entorno. Racionalidad. Sistemas multiagentes. Diseño de agentes. Arquitecturas de agentes. Entornos de simulación. Batallas de tanques: RoboCode. Partidos de fútbol (SoccerBots)
- Adquisición de conocimiento Técnicas y metodologías de adquisición de conocimiento. Técnicas de extracción automática de conocimiento. Aprendizaje máquina, Data Mining.

Programa detallado en inglés:

- Knowledge Based Systems (KBS). Introduction. KBS Life Cycle. KBS Development. Knowledge Engineering. KBS Development model. Tools. KBS Architecture. Viability analysis, Requirements specification. KBS Design. Reasoning Paradigms.
- Paradigms on Knowledge Representation and inference:
 - Case Based Reasoning (CBR): introduction, application domains, case retrieval and reuse. CBR system examples and tools. jCOLIBRI. Recommender systems.
 - Textual reasoning. Textual CBR using IR (Information Retrieval) techniques. Intelligent access to information systems. Automatic document classification systems.
 - Ontologies. Introduction. Reasoning. Development. Applications. Ontology languages. Semantic Web. Protege-OWL.
- AI in simulation environments. Software agents. Introduction. Definition. Classification. Agents and environment. Rationality. Design. Agent Architectures. Multiagent systems. Simulation environments: tanks (RoboCode). SoccerBots.
- Knowledge acquisition. Technologies and methodologies. Automatic knowledge extraction from data. Machine learning and data mining basics.

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.
- 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.
- 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:



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: Aprender y practicar los conceptos complementarios de Ingeniería de sistemas basados en el conocimiento
Evaluación detallada: Evaluación continua mediante la realización y defensa de prácticas realizadas en grupos de 2/3 personas. No se descarta la realización de examen para situaciones particulares: entrega de trabajos no originales, no asistencia a clase.
Actividades docentes: Reparto de créditos: Otras actividades: Teoría: 3,00 No tiene Problemas: 0,00 Laboratorios: 3,00
Bibliografía: Material de clase publicado en el campus virtual. Robert J. Schalkoff: Intelligent Systems: Principles, Paradigms, And Pragmatics Hardcover – November 9, 2009 A. Tiwana; The Knowledge Management Toolkit: Practical Techniques for Building a Knowledge ; Prentice Hall, 1999; Rajendra Akerkar : Knowledge-Based Systems – August 25, 2009 John Davies, Rudi Studer, Paul Warren (Eds).Semantic Web Technologies: Trends and Research in Ontology-based Systems. Wiley (July 11, 2006) Ian Witten, Eibe Frank, Mark Hall. Data Mining: Practical Machine Learning Tools and Techniques. January 2011. Morgan Kaufmann Publishers (ISBN: 978-0-12-374856-0)

Ficha docente guardada por última vez el 31/08/2015 10:41:00 por el usuario: Vic. Estudios

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 900246 - Ingeniería web Asignatura en Inglés: Web engineering		Abrev: IW Carácter: Optativa	6 ECTS
Materia: Complementos de ingeniería de software e inteligencia artificial		24 ECTS	
Otras asignaturas en la misma materia: Aprendizaje automático y big data Ingeniería de sistemas basados en el conocimiento Programación evolutiva		6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: 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 y pruebas.
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 and testing
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.

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 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 : unleash the power of the latest Spring MVC 4.x to develop a complete application : beginner's guide, by Amuthan Ganeshan - disponible como e-book en biblioteca FdI
- Effective Java, 2nd Edition; Joshua Bloch; Addison-Wesley, 2012; ISBN 978-0321356680; disponible en la biblioteca de la FdI.
- Eloquent JavaScript: A Modern Introduction to Programming. Marijn Haverbeke. 2011. ISBN 978-1593272821; disponible en <http://eloquentjavascript.net>.
- Van Lancker, Luc. "HTML5 y CSS3. Domine los estándares de las aplicaciones web". Ediciones ENI, 2011. ISBN: 978-2746068162. Disponible en la biblioteca de la FdI.
- Van Lancker, Luc. "jQuery. El framework JavaScript de la Web 2.0". Ediciones ENI, 2012. ISBN: 978-2746072589. Disponible en la biblioteca de la FdI.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 900247 - Programación evolutiva Asignatura en Inglés: Evolutionary Computation		Abrev: PEV Carácter: Optativa	6 ECTS
Materia: Complementos de ingeniería de software e inteligencia artificial		24 ECTS	
Otras asignaturas en la misma materia: Aprendizaje automático y big data Ingeniería de sistemas basados en el conocimiento Ingeniería web		6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Cervigon Rückauer, Carlos	

Descripción de contenidos mínimos:

Algoritmos genéticos
Algoritmos evolutivos
Programación genética
Gramáticas Evolutivas
Extensiones de algoritmos evolutivos

Programa detallado:

1. Introducción a la Programación Evolutiva.
2. Estructura y componentes básicos del algoritmo genético simple. Operadores básicos.
3. Implementación del Algoritmo genético simple.
4. Mejoras al esquema básico del algoritmo genético simple (representación, operadores).
5. Fundamentos matemáticos.
6. Algoritmos evolutivos: otras representaciones.
7. Programación genética
8. Gramáticas evolutivas.
9. Extensiones de los algoritmos evolutivos: ACO, PSO, Multiobjetivo, Meméticos...

Programa detallado en inglés:

1. Introduction to Evolutionary Computation.
2. Structure and components of simple genetic algorithm.
3. Implementation of simple genetic algorithm.
4. Improvements to the basic outline of simple genetic algorithm.
5. Mathematical Foundations.
6. Evolutionary algorithms: other representations.
7. Genetic programming
8. Gramatical evolution.
9. Extension of evolutionary algorithms: ACO, PSO, Multiobjective, Memetics...

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%)

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 requiere nivel avanzado de lenguaje Java) que se deben presentar y defender en las fechas establecidas durante el curso: Tres prácticas obligatorias realizadas en parejas, pero defendidas individualmente: 75%.

B. Actividades adicionales: Presentación de trabajos, exposición de temas relacionados con la asignatura, presentaciones de prácticas, participación y ejercicios en el aula: 25%.

La convocatoria extraordinaria dispondrá de un nuevo plazo para la defensa de las prácticas suspendas o no entregadas en plazo establecido.

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:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5° (1C)	Idioma: Español
Asignatura: 900248 - Criptografía y teoría de códigos Asignatura en Inglés: Cryptography and code theory	Abrev: CTC Carácter: Optativa	6 ECTS
Materia: Complementos de matemáticas e investigación operativa	6 ECTS	
Otras asignaturas en la misma materia: No hay		
Módulo: Optativo		
Departamento: Álgebra, Geometría y Topología		Coordinador: Luengo Velasco, Ignacio

Descripción de contenidos mínimos:

Criptografía. 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:



UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (1C)	Idioma: Español
Asignatura: 900250 - Especificación, validación y testing Asignatura en Inglés: Specification, validation, and testing		Abrev: EVT Carácter: Optativa	6 ECTS
Materia: Complementos de programación y lenguajes		18 ECTS	
Otras asignaturas en la misma materia: Informática gráfica Programación con restricciones		6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Sistemas Informáticos y Computación		Coordinador: García Merayo, M ^a . de las Mercedes	

Descripción de contenidos mínimos: No tiene
Programa detallado: <ol style="list-style-type: none">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: <ol style="list-style-type: none">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: No tiene
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: No tiene
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 INFORMÁTICA

Convocatoria de febrero: Prácticas 90%; Participación en clase 10%.

Convocatoria de septiembre: Examen 50%; Prácticas 40%; Participación en clase 10%.

El peso de cada práctica en la nota final, que será una media ponderada de las notas obtenidas en cada práctica, dependerá de su complejidad y se anunciará con la debida antelación. Estos pesos se moverán en un rango fijado entre 0,1 y 0,4. En el caso de la convocatoria de septiembre, se mantiene la nota que el alumno hubiera obtenido en la evaluación de las prácticas 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. 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 30/06/2021 13:19:00 por el usuario: Coordinador GII

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 900251 - Programación con restricciones Asignatura en Inglés: Constraint Programming	Abrev: PR Carácter: Optativa	6 ECTS
Materia: Complementos de programación y lenguajes	18 ECTS	
Otras asignaturas en la misma materia: Especificación, validación y testing Informática gráfica	6 ECTS 6 ECTS	
Módulo: Optativo		
Departamento: Sistemas Informáticos y Computación		Coordinador: Rubio Gimeno, Albert

Descripción de contenidos mínimos: Codificación. Resolutores. Optimizadores. Dominios de aplicación.
Programa detallado: <ul style="list-style-type: none">• 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: <ul style="list-style-type: none">- 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: No tiene
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) 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)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMATICA

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:

- 70%: realización de dos trabajos prácticos.
- 20%: realización de un examen.
- 10%: 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).

Ficha docente guardada por última vez el 30/06/2021 13:19:00 por el usuario: Coordinador GII

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 900271 - Desarrollo de videojuegos mediante tecnologías web Asignatura en Inglés: Web technologies for game development	Abrev: DVI Carácter: Optativa	6 ECTS
Materia: Complementos de software de sistemas	12 ECTS	
Otras asignaturas en la misma materia: Programación de aplicaciones para dispositivos móviles	6 ECTS	
Módulo: Optativo		
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: León Aznar, Carlos

Descripción de contenidos mínimos:

HTML5 y JavaScript
Arquitecturas de videojuegos
Tecnologías de videojuegos: sprites, audio, física.

Programa detallado:

1. Programación de aplicaciones y juegos en HTML5.
2. JavaScript.
3. Arquitectura de un motor de juegos en JavaScript: componentes y eventos.
4. Carga de recursos. Gestión de imágenes.
5. Gestión de entidades. Creación de recursos y exportación con editores de mapas.
6. Audio.
7. Motores de física.
8. Animaciones basadas en sprites.
9. Despliegue de proyectos en la web.

Programa detallado en inglés:

1. Application and game development in HTML5.
2. JavaScript.
3. Game engine architecture in JavaScript: components and events.
4. Resource management. Image management.
5. Entity management. Map editors: resource creation and export.
6. Audio.
7. Physics engines.
8. Sprite-based animation.
9. Web projects deployment.

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.

Evaluación detallada:**Convocatoria ordinaria:**

Es obligatoria la asistencia a clase y la realización en grupo de un proyecto (consistente en el desarrollo de un videojuego) y su defensa para poder aprobar la asignatura en la convocatoria ordinaria:

- Defensa del proyecto: 20% de la nota
- Calidad del proyecto: 80% de la nota. En este apartado se valorará el: diseño de la implementación, video de gameplay, declaración del trabajo individual y código fuente.

Convocatoria extraordinaria:

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMATICA

Existe la posibilidad de entregar el proyecto en la convocatoria extraordinaria, manteniéndose para esta convocatoria la nota del trabajo de evaluación continua durante el curso.

Actividades docentes:

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

Bibliografía:

- 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 30/06/2021 13:16:00 por el usuario: Coordinador GII

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 900272 - Programación de aplicaciones para dispositivos móviles	Abrev: PAD Carácter: Optativa		6 ECTS
Asignatura en Inglés: Application Programming for Mobile Devices			
Materia: Complementos de software de sistemas		12 ECTS	
Otras asignaturas en la misma materia: Desarrollo de videojuegos mediante tecnologías web		6 ECTS	
Módulo: Optativo			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: Valiente Blázquez, Mª Cruz	

Descripción de contenidos mínimos:

Desarrollo nativo para plataformas móviles
Aplicaciones web móviles
Desarrollo multiplataforma

Programa detallado:

- 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).

Programa detallado en inglés:

- 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).

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.

Resultados de aprendizaje:

- 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:

La asignatura tiene un objetivo eminentemente práctico y de evaluación continua, con un enfoque de aprendizaje basado en proyectos, por lo que la evaluación se encuentra dividida en las siguientes partes:

- Elaboración y exposición pública en clase de un proyecto Android (aplicación nativa) en grupo, que proporcionará el 60% de la nota final.
- Elaboración de prácticas en grupo a lo largo del cuatrimestre, que proporcionará el 30% de la nota final.
- Participación individual en clase y en los foros propuestos para la asignatura, 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.

Los alumnos que no superen la asignatura en convocatoria ordinaria serán evaluados de la misma manera en la convocatoria extraordinaria, debiendo consensuar con el profesor la realización de un nuevo proyecto y/o prácticas. Se conservarán hasta entonces las notas de las partes de la asignatura con nota mayor o igual que 4 (sobre 10).

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.

Ficha docente guardada por última vez el 30/06/2021 13:18:00 por el usuario: Coordinador GII

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 900273 - Programación de GPUs y aceleradores Asignatura en Inglés: GPUs and accelerators programming		Abrev: GPU Carácter: Optativa	6 ECTS
Materia: Complementos de arquitectura y tecnología de computadores		24 ECTS	
Otras asignaturas en la misma materia: Cloud y Big Data Diseño automático de sistemas Robótica		6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Arquitectura de Computadores y Automática		Coordinador: 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:

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:

Explotación eficiente de hardware específico para acelerar problemas. (CG1, CG2, CG4)

Evaluación de metodologías de programación paralela en problemas complejos. (CG11, CG14, CG23)

Búsqueda e interpretación de información técnica para desarrollar proyectos informáticos. (CT1, CT2, CT3)

Evaluación detallada:

La calificación consta de un Trabajo Final que constituye el 20% de la nota.

El 80% restante podrá ser obtenido de las siguientes formas:

a.- Sólo para la convocatoria ordinaria. Aquellos estudiantes que hayan asistido a un 80% de las clases prácticas su calificación se calculará como 40% de test + 40% de las prácticas

b.- En convocatoria extraordinaria o aquellos estudiantes que no cumplan el mínimo de asistencia en ambas convocatorias. El 60% examen final + 20% evaluación de la parte práctica de la asignatura, esta evaluación se realizará en el aula en la misma fecha y hora que el examen.

Actividades docentes:

Reparto de créditos:

Teoría: 2,50

Problemas: 0,50

Laboratorios: 3,00

Otras actividades:

Enseñanza presencial en aula y laboratorio

Bibliografía:

- The CUDA handbook : a comprehensive guide to GPU programming / Nicholas Wilt.
- Intel Xeon Phi Coprocessor High Performance Programming / James Jeffers, James Reinders
- Data Parallel C++. Mastering DPC++ for Programming of Heterogeneous Systems using C++ and SYCL, James Reinders, et al. Springer.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5º (1C)	Idioma: Español
Asignatura: 900281 - Testing de Software Asignatura en Inglés: Software Testing	Abrev: TSW Carácter: Optativa	6 ECTS
Materia: Complementos de ingeniería de software e inteligencia artificial	30 ECTS	
Otras asignaturas en la misma materia: Aprendizaje automático y big data Ingeniería de sistemas basados en el conocimiento Ingeniería web Programación evolutiva	6 ECTS 6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo		
Departamento: Sistemas Informáticos y Computación		Coordinador: García Merayo, M ^a . 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.

Convocatoria ordinaria: Prácticas 90%; Participación en clase 10%.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Convocatoria extraordinaria: Examen 50%; Prácticas 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. 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.

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. 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 30/06/2021 14:07:00 por el usuario: Coordinador GII

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (1C)	Idioma: Español
Asignatura: 901932 - Programación Competitiva Asignatura en Inglés: Competitive Programming		Abrev: PCOM Carácter: Optativa	6 ECTS
Materia: Complementos de ingeniería de software e inteligencia artificial		42 ECTS	
Otras asignaturas en la misma materia: Aprendizaje automático y big data Ingeniería de Comportamientos Inteligentes Ingeniería de sistemas basados en el conocimiento Ingeniería web Programación evolutiva Testing de Software		6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Interdepartamental ISIA / SIC		Coordinador: Gómez Martín, Marco Antonio	

Descripción de contenidos mínimos:

Jueces de programación.
Tipos de concursos.
Tipos de problemas.
Fundamentos matemáticos de soluciones.
Estructuras de datos en programación competitiva.
Técnicas algorítmicas en programación competitiva.

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
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

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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.

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:

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMATICA

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

Ficha docente guardada por última vez el 09/07/2021 18:42:00 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:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (1C)	Idioma: Español
Asignatura: 901933 - Ingeniería de Comportamientos Inteligentes Asignatura en Inglés: Intelligent Behaviours Engineering		Abrev: ICI Carácter: Optativa	6 ECTS
Materia: Complementos de ingeniería de software e inteligencia artificial		42 ECTS	
Otras asignaturas en la misma materia: Aprendizaje automático y big data Ingeniería de sistemas basados en el conocimiento Ingeniería web Programación Competitiva Programación evolutiva Testing de Software		6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Ingeniería del Software e Inteligencia Artificial		Coordinador: 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:**

No tiene

Específicas:

No tiene

Básicas y Transversales:

No tiene

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.

Evaluación detallada:**Convocatoria ordinaria:**

Evaluación continua mediante la realización y defensa de prácticas realizadas en grupo: 100%. La nota final se calculará como la media de las calificaciones obtenidas en las prácticas. Los alumnos que no aprueban las practicas podrán realizar un examen teórico.

Convocatoria extraordinaria:

Las notas de las prácticas entregadas y aprobadas durante el curso se guardan para la convocatoria extraordinaria. Para aprobar en convocatoria extraordinaria el alumno debe entregar las prácticas no entregadas o suspensas en la convocatoria ordinaria, o las modificaciones indicadas por el profesor en cada caso.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

En función de las situaciones particulares de cada alumno, el profesor puede proponer modificaciones en los enunciados de las prácticas para las entregas de la convocatoria extraordinaria. Estas modificaciones serán comunicadas a los alumnos con suficiente antelación y no más tarde de la publicación de actas de la convocatoria ordinaria.

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 Gamo 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 01/07/2021 20:37:00 por el usuario: Vic. Estudios

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 901934 - Inteligencia Artificial Aplicada al Control Asignatura en Inglés: Artificial Intelligences applied to control systems.	Abrev: IAAC Carácter: Optativa	6 ECTS
Materia: Complementos de sistemas inteligentes	6 ECTS	
Otras asignaturas en la misma materia: No hay		
Módulo: Optativo		
Departamento: Arquitectura de Computadores y Automática		Coordinador: 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
Aplicaciones

Programa detallado en inglés:

Automatic control
Intelligent control
Expert Systems applied to control
Neuro-control
Fuzzy-control
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)
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)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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 (30%)

- Se realizará la evaluación final mediante un examen escrito o mediante la presentación y memoria de un trabajo, según se solicite (60%).

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
- Cai, Si-Xing., Intelligent Control and Intelligent Automation. Principles, techniques and applications. World Scientific, 1997
- King, R.E., Computational Intelligence in Control Engineering. Control Engineering Series, Marcel Dekker, N.Y., 1999

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 901935 - Diseño automático de sistemas Asignatura en Inglés: Digital Systems Design		Abrev: DAS Carácter: Optativa	6 ECTS
Materia: Complementos de arquitectura y tecnología de computadores		24 ECTS	
Otras asignaturas en la misma materia: Cloud y Big Data Programación de GPUs y aceleradores Robótica		6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Arquitectura de Computadores y Automática		Coordinador: Román Navarro, Sara	

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.
Especificación a nivel lógico-RT de sistemas digitales.
Técnicas de diseño de nivel lógico-RT.
Análisis de sistemas digitales a nivel lógico-RT.
Síntesis de sistemas
Prácticas de laboratorio.

Programa detallado en inglés:

Introduction to design automation of digital systems.
Logic-RT level specification.
Logic-RT level design techniques.
Logic-RT level analysis of digital systems.
System Synthesis
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)
Diseñar utilizando lenguajes de descripción hardware de alto nivel. (CG4, CG14, CT2)
Comprender el funcionamiento del hardware reconfigurable. (CG4, CG14)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Evaluación detallada:

La evaluación es continua y la nota final en cualquiera de las convocatorias se obtendrá sumando ponderadamente la nota obtenida por las siguientes actividades:

- 1) Prácticas (50% de la nota)
- 2) Examen (50% de la nota)

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 30/06/2021 13:18:00 por el usuario: **Coordinador GII**

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 901936 - Minería de datos y el paradigma Big Data Asignatura en Inglés: Data mining and the Big Data paradigm		Abrev: MIN Carácter: Optativa	6 ECTS
Materia: Complementos de gestión y procesamiento de la información		6 ECTS	
Otras asignaturas en la misma materia: No hay			
Módulo: Optativo			
Departamento: Sistemas Informáticos y Computación		Coordinador: 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.

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.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- 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.

3.3.3.2.- Regression.

3.3.3.3.- Clustering.

3.3.3.4.- Association rules.

Fecha: ____ de _____ de _____

Firma del Director del Departamento:



- 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_T11-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_T11)
- 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_T11)
- 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)
- 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)

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (1C)	Idioma: Español
Asignatura: 901937 - Cloud y Big Data Asignatura en Inglés: Cloud and Big Data		Abrev: CLO Carácter: Optativa	6 ECTS
Materia: Complementos de arquitectura y tecnología de computadores		24 ECTS	
Otras asignaturas en la misma materia: Diseño automático de sistemas Programación de GPUs y aceleradores Robótica		6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Arquitectura de Computadores y Automática		Coordinador: 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. Data Science a gran escala
 - Computational Science
 - Data Science
 - Necesidad de procesamiento paralelo
2. Arquitecturas de procesamiento paralelo
 - Arquitecturas paralelas de memoria compartida
 - Sistemas GPU
 - Arquitecturas paralelas de memoria distribuida
3. Procesamiento a gran escala en el Cloud
 - Virtualización
 - ¿Que es Cloud Computing?
 - Tipos de servicios Cloud
 - Servicios Cloud para Big Data
4. Aspectos prácticos de Cloud Computing
 - Aspectos económicos
 - El estado del Cloud público
 - Necesidad de Cloud privados
 - Anatomía del Cloud
5. Fundamentos de procesamiento de datos
 - Descripción del problema
 - Big Data
 - Modelos de procesamiento de datos
6. Procesamiento de datos modo Batch
 - Modelo programación MapReduce
 - Plataforma Hadoop
 - Ecosistema Hadoop
7. Procesamiento de datos modo Dataflow
 - Limitaciones MapReduce
 - Sistema Spark
 - RDDs, transformaciones, acciones
 - Ecosistema Spark
8. Procesamiento de datos modo Stream
 - Big Streaming Data
 - Procesamiento Stream con Spark
 - Procesamiento Stream en el Edge

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Laboratorio:

- Public cloud: Amazon Wen Services
- Private cloud: OpenNebula and Docker
- Big Data: MapReduce/Hadoop and Spark

Programa detallado en inglés:

Programa detallado en inglés:

1. Large-scale Data Science

- Computational science
- Data science
- The need for parallel processing

2. Parallel Processing Architectures

- Shared-memory parallel architectures
- GPU systems
- Distributed-memory parallel architectures

3. Large-scale Processing on the Cloud

- Virtualization
- ¿What is cloud computing?
- Types of cloud services
- Cloud services for parallel processing

4. Practical Aspects of Cloud Computing

- Economic aspects
- The state of public cloud
- The need for private clouds
- The anatomy of the cloud

5. Foundations of Data Processing

- Problem description
- Big Data basics
- Data processing models

6. Batch Data Processing

- The MapReduce programming model
- The Hadoop processing framework
- The Hadoop ecosystem

7. Dataflow Processing

- MapReduce limitations
- The Spark execution engine
- RDDs, transformations, actions
- The Spark ecosystem

8. Stream Data Processing

- Big streaming data
- Stream Processing with Spark
- Stream Processing at the Edge

Hands-on

- Public cloud: Amazon Wen Services
- Private cloud: OpenNebula and Docker
- Big Data: MapReduce/Hadoop and Spark

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.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



<p>CG18-Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.</p> <p>CG19-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.</p>								
<p>Específicas: No tiene</p>								
<p>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.</p>								
<p>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) Aplicar big data y cloud para resolver un problema de procesamiento de datos a gran escala. (CG18, CT3)</p>								
<p>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%).</p>								
<p>Actividades docentes:</p> <table><tr><td>Reparto de créditos:</td><td>Otras actividades:</td></tr><tr><td>Teoría: 3,00</td><td>No tiene</td></tr><tr><td>Problemas: 0,00</td><td></td></tr><tr><td>Laboratorios: 3,00</td><td></td></tr></table>	Reparto de créditos:	Otras actividades:	Teoría: 3,00	No tiene	Problemas: 0,00		Laboratorios: 3,00	
Reparto de créditos:	Otras actividades:							
Teoría: 3,00	No tiene							
Problemas: 0,00								
Laboratorios: 3,00								
<p>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 - Cloudeconomics: 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</p>								

Ficha docente guardada por última vez el 30/06/2021 14:05:00 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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (1C)	Idioma: Español
Asignatura: 901938 - Robótica Asignatura en Inglés: Robotics		Abrev: ROB Carácter: Optativa	6 ECTS
Materia: Complementos de arquitectura y tecnología de computadores		24 ECTS	
Otras asignaturas en la misma materia: Cloud y Big Data Diseño automático de sistemas Programación de GPUs y aceleradores		6 ECTS 6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Arquitectura de Computadores y Automática		Coordinador: López Orozco, José Antonio	

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.

5.- Navigation and autonomous robots

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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.
- 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.
- 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:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMATICA

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.

Ficha docente guardada por última vez el 30/06/2021 13:24:00 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:

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

Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 901967 - Introducción a la Tecnología Blockchain y Smart Contracts Asignatura en Inglés: Introduction to Blockchain Technology and Smart Contracts		Abrev: TBC Carácter: Optativa	6 ECTS
Materia: Complementos de programación y lenguajes		18 ECTS	
Otras asignaturas en la misma materia: Especificación, validación y testing Programación con restricciones		6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Sistemas Informáticos y Computación		Coordinador: Correas Fernández, Jesús	

Descripción de contenidos mínimos:

- Fundamentos de la tecnología Blockchain
- Programación de smart contracts
- Aplicaciones

Programa detallado:

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.

Programa detallado en inglés:

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.

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.
- 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:

- 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:

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.

Ficha docente guardada por última vez el 30/06/2021 14:05:00 por el usuario: Coordinador GII

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



Ficha del curso: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA	Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 901968 - Computación Cuántica Asignatura en Inglés: Quantum Computing	Abrev: CC Carácter: Optativa	6 ECTS
Materia: Complementos de matemáticas e investigación operativa	12 ECTS	
Otras asignaturas en la misma materia: Criptografía y teoría de códigos	6 ECTS	
Módulo: Optativo		
Departamento: Análisis Matemático y Matemática Aplicada		Coordinador: Pérez García, David

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: Fundamentos.
 - 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 Códigos correctores de error clásicos
 - 4.3 Códigos de Shor y Steane.
 - 4.4 Códigos cuánticos derivados de estabilizadores.
 - 4.5 Computación cuántica robusta.
- 5.- Trabajo por grupos: análisis de algoritmos cuánticos y de paradigmas de computación.

Programa detallado en inglés:

Quantum computing is becoming a key technology for the neat 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.

OBJECTIVES:

The main goal is to acquire the necessary knowledge to design quantum algorithms.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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: basics
 - 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 Classical error correcting codes
 - 4.3 Shor code and Steane code
 - 4.4 Quantum codes from stabilizers
 - 4.5 Fault-tolerant quantum computation.
- 5.- Teamwork: analysis of quantum algorithms and quantum computing paradigms.

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:

La evaluación de los primeros cuatro temas será mediante la participación activa en la resolución de los problemas o una prueba parcial. Contará el 20-40% de la nota final.
El tema 5 consistirá en el trabajo por grupos pequeños supervisados por el profesor para analizar (o incluso mejorar o diseñar) un algoritmo cuántico que resuelva un problema concreto de interés, o un esquema concreto de computación. El desarrollo y presentación del trabajo supondrá el 60-80% de la nota final.

Actividades docentes:

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMATICA

Reparto de créditos:

Teoría: 3,00

Problemas: 1,00

Laboratorios: 2,00

Otras actividades:

La docencia de los primeros cuatro temas será mediante clases magistrales y clases de problemas, donde los alumnos resolverán en la pizarra los problemas que se les propongan asociados a cada tema.

El tema 5 consistirá en el trabajo por grupos pequeños supervisados por el profesor para analizar (o incluso mejorar o diseñar) un algoritmo cuántico que resuelva un problema concreto de interés, o un esquema concreto de computación. Las clases tendrán en esa parte del curso un carácter eminentemente práctico: discusión conjunta de ideas, planteamiento y solución de dudas, ...

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 30/06/2021 14:06:00 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: 2021-2022

Grado: DOBLE GRADO DE MATEMÁTICAS E INFORMÁTICA		Curso: Optativas 5º (2C)	Idioma: Español
Asignatura: 901969 - Arquitecturas y Programación de Computadores Cuánticos Asignatura en Inglés: Architecture and Programming of Quantum Computers		Abrev: APCC Carácter: Optativa	6 ECTS
Materia: Complementos de computadores		18 ECTS	
Otras asignaturas en la misma materia: Programación paralela para móviles y multicores Seguridad en redes		6 ECTS 6 ECTS	
Módulo: Optativo			
Departamento: Arquitectura de Computadores y Automática		Coordinador: 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
 - 1.2. Reversible computing
2. Introduction to quantum circuits

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



- 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
- 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

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



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.

Fecha: ____ de _____ de ____

Firma del Director del Departamento:



UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMATICA

Fecha: ____ de _____ de ____

Firma del Director del Departamento: