

Plan 394 GRADO DE MATEMATICAS

Asignatura 40004 INFORMÁTICA

Grupo 1

### Tipo de asignatura (básica, obligatoria u optativa)

Básica, Rama de Ingeniería y Arquitectura

### Créditos ECTS

6

### Competencias que contribuye a desarrollar

#### Generales

- Conocer y utilizar recursos informáticos de carácter general y tecnologías de la información y las comunicaciones como medios de comunicación, organización, aprendizaje e investigación.
- Leer y comprender textos científicos tanto en lengua propia como en otras de relevancia en el ámbito científico, especialmente la inglesa.
- Gestionar de forma óptima, tanto en el trabajo individual como en equipo, el tiempo de trabajo y organizar los recursos disponibles, estableciendo prioridades, caminos alternativos e identificando errores lógicos en la toma de decisiones.
- Tener la capacidad de trabajar en equipo, aportando orden, abstracción y razonamiento lógico; comprobando o refutando razonadamente los argumentos de otras personas y contribuyendo con profesionalidad al buen funcionamiento y organización del grupo.

#### Específicas

- Desarrollar programas que resuelvan problemas matemáticos utilizando para cada caso el entorno computacional adecuado.
- Utilizar aplicaciones informáticas de análisis estadístico, cálculo numérico y simbólico, visualización gráfica, optimización u otras para experimentar en Matemáticas y resolver problemas.
- Saber abstraer las propiedades estructurales (de objetos matemáticos, de la realidad observada, y de otros ámbitos) distinguiéndolas de aquellas puramente ocasionales y poder comprobarlas con demostraciones o refutarlas con contraejemplos, así como identificar errores en razonamientos incorrectos.
- Capacitar para el aprendizaje autónomo de nuevos conocimientos y técnicas.
- Proponer, analizar, validar e interpretar modelos de situaciones reales sencillas, utilizando las herramientas matemáticas más adecuadas a los fines que se persigan.
- Planificar la resolución de un problema en función de las herramientas de que se disponga y de las restricciones de tiempo y recursos.

### Objetivos/Resultados de aprendizaje

1. Comprender los principios de funcionamiento de las computadoras y el modo en que se representa la información en ellas. Manejar una plataforma de sistema operativo a un nivel suficiente que permita desenvolverse con soltura en sus necesidades cotidianas que requieran el uso de computadoras.
2. Asimilar el funcionamiento interno de un programa de computador. Entender cómo representar la información mediante tipos de datos básicos. Integrar el comportamiento interactivo o de comunicación con el usuario en el proceso secuencial de un programa. Entender y manejar las estructuras fundamentales de control. Aprender a manejar las estructuras fundamentales de datos así como escoger en cada caso la más adecuada y los algoritmos de manejo más eficientes.
3. Comprender el proceso general de la programación. Comprender y analizar el concepto de eficiencia o complejidad en algoritmos básicos. Tener la capacidad de elección de la estructura de datos adecuada para cada tipo de problema.

## Contenidos

### Bloque 1. Conceptos básicos de Informática (0,9 ECTS)

#### Tema 1. Conceptos Fundamentales.

- Algunas definiciones.
- Un poco de historia.
- Representación de la Información en un ordenador.
- Funcionamiento de un ordenador.
- Qué es un Sistema Operativo. S.O. Windows.
- Redes de Ordenadores e Internet.
- Herramientas software a utilizar en la asignatura: Explorador, Web-Mail, Moodle.

### Bloque 2. Fundamentos de Programación (5,1 ECTS)

#### Tema 1. Introducción a la Programación: lenguajes y paradigmas.

Introducción a los lenguajes de programación. Historia. Paradigmas de programación. Fases de creación de un programa. Traducción: compilación e interpretación.

#### Tema 2. Nociones de entorno, procesador y acción.

Noción de entorno, acción y procesador. Acciones primitivas. Acciones compuestas. Noción de análisis descendente. Noción de algoritmo.

#### Tema 3. Tipos y Acciones elementales.

Objetos, constantes y variables. Tipos elementales. Expresiones. Acción de asignación. Composición de acciones. Descripción de un algoritmo. Los arrays.

Presentación del lenguaje C. Estructura de un programa en C. Tipos de datos escalares. Operadores básicos.

Definición de un tipo array.

#### Tema 4. Control de flujo del programa I: Esquemas Condicionales.

Esquemas condicionales (sintaxis, semántica y ejemplos): condicional simple, condicional compuesto y esquema condicional generalizado.

Condicionales en C: if (), if ()+ else, switch ().

#### Tema 5. Control de flujo del programa II: Esquemas Repetitivos.

Tipos de esquemas (sintaxis, semántica, ejemplos): mientras, repetir, para. Equivalencia entre esquemas.

Esquemas repetitivos en C: while(), do ... while(), for ().

#### Tema 6. Algoritmos con nombre: funciones.

Parametrización de un algoritmo con nombre. Funciones. Los punteros.

Paso de parámetros en C: paso por valor y paso por referencia.

#### Tema 7. Estructuras de datos: Arrays y Registros.

Tipos estructurados. Tipos vectoriales. Tipos compuestos. El tipo string.

Recorrido secuencial de un array. Búsqueda secuencial. Arrays ordenados. Búsqueda dicotómica. Arrays de más de una dimensión: matrices. Operaciones elementales con matrices.

Definición de arrays en C. Recorrido de arrays en C. Paso de Arrays como argumentos. Arrays y punteros.

#### Tema 8. Ficheros.

Noción de fichero. Tipos de acceso a ficheros. Asociación de un fichero a un programa. Operaciones primitivas de acceso a ficheros. Operaciones de creación, recorrido, búsqueda y actualización de un fichero secuencial.

Ficheros en C. Operaciones básicas y recorrido de ficheros en C.

#### Tema 9. Memoria Dinámica.

Memoria estática vs memoria dinámica. Operaciones básicas de gestión de memoria dinámica. Memoria dinámica y punteros.

## Principios Metodológicos/Métodos Docentes

La docencia estará enfocada al aprendizaje del estudiante.

Para ello se utilizarán distintos métodos docentes:

- Clases Teóricas.
- Seminarios.
- Clases prácticas en el laboratorio de ordenadores de la Facultad.

En el bloque 2, además se complementarán estos métodos con Sesiones de Problemas, bien intercaladas con clases teóricas o como parte de una sesión de aula.

## Criterios y sistemas de evaluación

La evaluación de la asignatura se divide en dos partes.

Durante el curso se realizará evaluación continua, con tres pruebas con distinto peso (10%, 25% y 40%) de la nota final. Además, se recogerán dos de los trabajos realizados en dos sesiones de prácticas, que tendrán un peso entre las dos del 25% de la nota final. La nota final de la convocatoria ordinaria se obtendrá con la suma ponderada de estas calificaciones.

Si no se supera la asignatura durante el curso, se realizará un examen extraordinario en julio donde entrará toda la materia de la asignatura y su superación supondrá el 100% de la nota en la convocatoria extraordinaria

A) La evaluación del bloque 1 consistirá en un ejercicio de preguntas cortas (0,5 h), y supondrá el 10% de la nota final

de la asignatura

B) La evaluación del bloque 2 consistirá en:

- Un ejercicio de evaluación con preguntas teórico prácticas, que supondrá el 25% de la nota final de la asignatura, que se realizará al terminar el tema 6 del bloque II.
- La entrega de dos prácticas por determinar, dentro de las sesiones de laboratorio, que supondrán en total el 25% de la nota final.
- Habrá una prueba final de cuestiones teórico-prácticas que supondrá el 40% de la nota final y que se realizará en la fecha oficial de examen de la asignatura. En este examen entrarán toda la materia relacionada con la programación

## Recursos de aprendizaje y apoyo tutorial

Al comenzar el curso, el o la estudiante dispondrá en el Campus Virtual de la UVA de todo el material de la asignatura.

Se incluirán las diapositivas presentadas en clase, como guión (nunca como único contenido a evaluar) de la asignatura, para facilitar su seguimiento.

Se proporcionarán artículos relevantes cuando se discutan aspectos específicos de las asignaturas.

Se pondrán las hojas de ejercicios, así como todos los avisos relativos a la asignatura.

Los enunciados de las prácticas también estarán disponibles cada semana, así como hojas de ejercicios para trabajo personal de los estudiantes.

## Calendario y horario

Las clases serán durante el primer cuatrimestre. Habrá tres horas semanales de clases en aula: teoría y clases de problemas. Además, habrá siete sesiones de seminario y 15 horas de laboratorio. Las clases de laboratorio están organizadas en sesiones de dos horas cada dos semanas, aproximadamente (se confirmará el horario final en función de los días festivos del cuatrimestre).

Se recomienda visitar la página <http://www.cie.uva.es> para confirmar los horarios y aula definitivas.

## Tabla de Dedicación del Estudiante a la Asignatura/Plan de Trabajo

ACTIVIDADES PRESENCIALES

HORAS

ACTIVIDADES NO PRESENCIALES

HORAS

Clases teóricas

20

Estudio y trabajo autónomo individual

70

Clases prácticas

15

Estudio y trabajo autónomo grupal

20

Laboratorios

15

Prácticas externas, clínicas o de campo

Seminarios

6

Otras actividades

4

Total presencial

60

Total no presencial

90

Responsable de la docencia (recomendable que se incluya información de contacto y breve CV en el que aparezcan sus líneas de investigación y alguna publicación relevante)

---

Responsable de la asignatura: Pulido Junquera, José Belarmino

Correo electrónico: belar@infor.uva.es

Universidad de Valladolid

Edificio de las Tecnologías de la Información y las Telecomunicaciones

Departamento de Informática

Campus Miguel Delibes

Camino del Cementerio s/n

47011. Valladolid

Teléfono (indicar prefijo, número y extensión): 983-185606

Fax: 983-423671

Doctorado: Febrero de 2001

Doctor por la Universidad de Valladolid. Premio Extraordinario de Doctorado ETSI Informática.

Líneas de investigación

- Diagnóstico Automático de Dispositivos,
- Diagnóstico basado en Consistencia,
- Razonamiento basado en Modelos,
- Supervisión y Diagnosis,
- Inteligencia Artificial,
- Sistemas Basados en Conocimiento

---

## Idioma en que se imparte

Español

---