



Universidad
Politécnica
de Cartagena



Centro
Universitario
de la Defensa

Guía docente de la asignatura:

Informática

Titulación:

Grado en Ingeniería de Organización Industrial

Curso: 2016-2017



1. Datos de la asignatura

Nombre		Informática (<i>Computer Science</i>)				
Materia*		Informática				
Módulo*		Materias básicas				
Código		511101005				
Titulación		Grado en Ingeniería de Organización Industrial				
Plan de estudios		2009 (Decreto 269/2009 de 31 de julio)				
Centro		Centro Universitario de la Defensa en la Academia General del Aire				
Tipo		Obligatoria				
Periodo lectivo		Cuatrimstral	Cuatrimestre	2º	Curso	1º
Idioma		Castellano				
ECTS	6	Horas / ECTS	25	Carga total de trabajo (horas)		150

* Todos los términos marcados con un asterisco están definidos en *Referencias para la actividad docente en la UPCT y Glosario de términos*:

<http://repositorio.bib.upct.es/dspace/bitstream/10317/3330/1/isbn8469531360.pdf>

2. Datos del profesorado

Profesor Responsable	Nina Skorin-Kapov		
Departamento	Departamento de Ingeniería y Tecnología Aplicada		
Área de conocimiento	Telecomunicaciones		
Ubicación del despacho	Despacho 7 - Edificio Administrativo del CUD		
Teléfono	968189911 (ext 2911)	Fax	968188780
Correo electrónico	nina.skorinkapov@ cud.upct.es		
URL / WEB	Aula Virtual UPCT: http://moodle.upct.es/course/view.php?id=511101005 Web Personal NSK: https://sites.google.com/site/ninaskorinkapov/		
Horario de atención / Tutorías	Martes y Jueves 12:40-14:40 Como criterio general, el alumno que desee realizar una tutoría deberá previamente (al menos con un día de antelación) enviar un e-mail al profesor solicitando una cita previa con el fin de poder organizar debidamente la atención de todo el alumnado.		
Ubicación durante las tutorías	Despacho 7 - Edificio Administrativo del CUD		

Perfil Docente e investigador	Doctorado en Ingeniería Eléctrica, Área Telecomunicaciones. Acreditada a Profesor Titular en Telecomunicaciones.
Experiencia docente	12 años. Asignaturas impartidas: Sistemas de Exploración Electromagnética, Informática, Metaheurística , Algoritmos y Métodos de Optimización, Tecnologías de Comunicación Fotónica, Redes de Telecomunicación Fotónica, Teoría de Información, Métodos Heurísticos de Optimización, Informática, Sistemas de Transmisión, Álgebra Lógica, Redes de Información, Fiabilidad de Sistemas de Telecomunicación, Teoría del Teletráfico, Lógica Digital, Sistemas de Telecomunicación y Redes
Líneas de Investigación	Optimización y planificación en telecomunicaciones, Redes ópticas, Algoritmos heurísticos, Seguridad de redes
Experiencia profesional	Miembro del Consejo de Administración del Instituto de Telemedicina (2009-13) Editora de Revista CIT - Journal of Computing and Information Technology



Profesor 2	José Santa Lozano		
Departamento	Departamento de Ciencias e Informática		
Área de conocimiento	Ingeniería Telemática		
Ubicación del despacho	Despacho 37 - Edificio Administrativo del CUD		
Teléfono	968 189943 (ext 2943)	Fax	968 189970
Correo electrónico	jose.santa@ cud.upct.es		
URL / WEB	Aula Virtual de la UPCT		
Horario de atención / Tutorías	Prevía cita por correo. Martes y Jueves (12:35-14:35)		
Ubicación durante las tutorías	Despacho 37 - Edificio Administrativo del CUD		

Perfil Docente e investigador	Doctor Ingeniero Informático. Acreditado a Profesor Titular de Universidad, Área Ingeniería Telemática
Experiencia docente	<p>3 años en Ingeniería Técnica en Informática de Sistemas en Universidad de Murcia; 4 años en Ingeniería Informática en la Universidad de Murcia; 5 años en Máster Universitario en Nuevas Tecnologías en Informática; 1 año en Escuela de Turismo de UPCT; 3 años en CUD de San Javier.</p> <p>Asignaturas impartidas: Sistemas Embebidos, Diseño Basado en Microprocesadores, Distribución Electrónica en el Sector Turístico, Fundamentos Físicos de la Informática, Redes Inalámbricas, Servicios Aplicados a la Movilidad, Tecnología Eléctrica, Automatización e Instrumentación Electrónica, Informática, Redes y Servicios de Comunicación.</p>
Líneas de investigación	Redes Móviles, Servicios Ubicuos, Edificios Inteligentes y Domótica, Redes Vehiculares, Sistemas de Navegación GNSS, Sistemas Inteligentes de Transporte, Telemática en el Transporte.
Experiencia profesional	<p>5 años.</p> <p>Gestión de Proyectos I+D+i e Ingeniero de Redes en el Departamento de Ingeniería de la Información y las Comunicaciones, en la Universidad de Murcia.</p>
Otros temas de interés	Tele-monitorización de infraestructuras y automatización.

Profesor 3	María Teresa Martínez Inglés		
Departamento	Departamento de Ingeniería y Tecnología Aplicada		
Área de conocimiento	Telecomunicaciones		
Ubicación del despacho	Despacho 2 - Edificio Administrativo del CUD		
Teléfono	968189916 (ext 2916)	Fax	968188780
Correo electrónico	mteresa.martinez@ cud.upct.es		
URL / WEB	Aula Virtual de la UPCT		
Horario de atención / Tutorías	<p>Martes y Jueves 12:50-14:35</p> <p>Como criterio general, el alumno que desee realizar una tutoría deberá previamente (al menos con un día de antelación) enviar un e-mail al profesor solicitando una cita previa con el fin de poder organizar debidamente la atención de todo el alumnado.</p>		
Ubicación durante las tutorías	Despacho 2- Edificio Administrativo del CUD		

Perfil Docente e investigador	Doctor Ingeniero de Telecomunicación. Área de Tecnologías de la Información y Comunicaciones. Acreditación Profesor Contratado Doctor.
Experiencia docente	<p>3 años</p> <p>Asignaturas impartidas: Comunicaciones Ópticas, Sistemas y Servicios de Telecomunicación, Radiocomunicaciones Terrestres y Espaciales, Informática y Sistemas de Exploración Electromagnética.</p>
Líneas de Investigación	<p>Medidas Experimentales del canal de radiocomunicaciones. Caracterización de modelos de propagación banda ancha basados en trazados de rayos para canales MIMO en entornos microcelulares. Aplicación Teoría Uniforme de la Difracción al desarrollo de modelos de propagación</p>
Experiencia profesional	<p>3 años</p> <p>AQUILINE: Ingeniero en desarrollo de software en la aplicación SELENE para Siemens Health Services y aplicaciones enmarcadas en la Consejería de Educación. UPCT: Ingeniero. Estudio de viabilidad del despliegue de un sistema cognitivo y desarrollo de simuladores IMS (IP Multimedia Subsystems).</p>

3. Descripción de la asignatura

3.1. Descripción general de la asignatura

La asignatura “*Informática*” se presenta como una *materia básica* en la formación de un ingeniero, y en este caso particular, un *futuro oficial del Ejército del Aire* con la titulación de *Grado en Ingeniería de Organización Industrial*. Esta asignatura aborda los fundamentos de los sistemas informáticos. Desde sus comienzos a mediados del siglo XX, los sistemas y las aplicaciones informáticas se han desarrollado como solución a las necesidades de la industria y la sociedad. Actualmente, la informática (incluyendo *hardware* y *software*) está presente en infinidad de contextos civiles y militares: los teléfonos móviles, los cajeros automáticos, los sistemas de navegación y de ayuda a la conducción, los sistemas de control y automatización de plantas industriales y de generación de energía, las aplicaciones de contabilidad y gestión económica, los sistemas de control aéreo, los sistemas de combate... Aunque estos sistemas difieren mucho entre sí, todos ellos comparten unos fundamentos comunes cuyo aprendizaje forma parte de los planes de estudio de todas las ingenierías. Así mismo, todos ellos son, entre muchos otros, sistemas intensivos en software, es decir sistemas en los que el software es una de sus partes fundamentales.

Por todo ello, el núcleo más importante de la asignatura lo constituye la introducción al desarrollo de software, describiendo para ello los elementos fundamentales que lo integran y las posibilidades existentes en el contexto de la ingeniería. El objetivo principal es que los alumnos aprendan los fundamentos teóricos básicos de la programación y desarrollen las habilidades necesarias para aplicarlos. Para ello, y desde un enfoque de programación estructurada, el alumno aprenderá a programar en lenguaje C.

La asignatura se desarrolla con un carácter eminentemente práctico, potenciándose para ello las prácticas en el aula de informática. Se pretende también el desarrollo de habilidades y competencias genéricas como el trabajo en equipo, el aprendizaje autónomo, la capacidad de organización y planificación y la preocupación por la calidad y el trabajo bien hecho.

3.2. Aportación de la asignatura al ejercicio profesional

Hoy en día, conocer los fundamentos de la Informática es parte integral de la formación de todo profesional, ya que los sistemas y las aplicaciones informáticas serán parte consustancial de los entornos de trabajo en los que se desarrollará su carrera. Así, un profesional debe ser capaz de hacer uso de los recursos propios de un sistema informático en el ámbito de su especialidad.

Con la asignatura *Informática*, el alumno será capaz de utilizar lenguajes de programación con los que desarrollar aplicaciones de propósito general. También se proporciona al alumno conocimientos básicos sobre las partes que constituyen un sistema informático y sobre el funcionamiento de los sistemas operativos. De manera complementaria, se le dan los fundamentos de las principales aplicaciones software en el ámbito de la ingeniería de organización, así como los principios básicos de las bases de datos. La formación aportada al alumno le capacita para poder extrapolar los conocimientos adquiridos a otras infraestructuras, de manera que las competencias adquiridas sean útiles en su desarrollo profesional a medio y largo plazo.

Es importante destacar que, dado el perfil de egreso del alumnado, en esta asignatura se incentiva un uso eficiente, responsable y seguro de los sistemas informáticos en las Fuerzas Armadas.



3.3. Relación con otras asignaturas del plan de estudios

La asignatura es básica y no requiere de otras asignaturas previas del plan de estudios. De esta forma, no existe ningún prerrequisito ni recomendación asociada con esta asignatura.

Tampoco mantiene una relación unívoca/directa con asignaturas específicas de la titulación de grado. Sí, de forma indirecta, con asignaturas tecnológicas que requieren - fundamentalmente para las sesiones prácticas- la utilización de aplicaciones software y la programación e implementación de algoritmos, modelos o sistemas. Algunos ejemplos de estas asignaturas tecnológicas pueden ser “Automatización e Instrumentación Electrónica” y “Mecánica de Fluidos” correspondientes a segundo curso, “Tecnologías de Seguridad y Defensa” y “Redes y Servicios de Comunicación” de tercer curso o “Sistemas de Exploración Electromagnética” de cuarto curso.

3.4. Incompatibilidades de la asignatura definidas en el plan de estudios

No existen incompatibilidades.

3.5. Recomendaciones para cursar la asignatura

No existen requisitos previos para cursar la asignatura.

3.6. Medidas especiales previstas

Se adoptarán medidas especiales que permitan simultanear los estudios de la asignatura con las actividades de formación militar. En concreto, y en caso de que surjan esas necesidades, se formarán grupos de trabajo/aprendizaje cooperativo de alumnos con disponibilidad limitada, fomentándose el seguimiento del aprendizaje mediante la programación de tutorías de grupo y planificación y entrega de actividades a través del Aula Virtual.



4. Competencias y resultados del aprendizaje

4.1. Competencias básicas* del plan de estudios asociadas a la asignatura

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.

4.2. Competencias generales del plan de estudios asociadas a la asignatura

Capacidad de organización y dirección de empresas e instituciones, evaluando los aspectos propios del comportamiento organizacional y de gestión de recursos, todo ello en un entorno legal seguro, proporcionado por sus conocimientos en materias jurídicas.

4.3. Competencias específicas* del plan de estudios asociadas a la asignatura

E1.1.c: Conocimientos fundamentales sobre el uso de la programación de ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

4.4. Competencias transversales del plan de estudios asociadas a la asignatura

- T1.1 Capacidad de análisis y síntesis
- T1.2 Capacidad de organización y planificación
- T1.3 Comunicación oral y escrita en lengua propia
- T1.5 Habilidades básicas computacionales
- T1.6 Capacidad de gestión de la información
- T1.7 Resolución de problemas
- T2.2 Trabajo en equipo
- T2.3 Habilidades en las relaciones interpersonales
- T3.1 Capacidad para aplicar los conocimientos a la práctica
- T3.2 Capacidad de aprender
- T3.3 Adaptación a nuevas situaciones
- T3.4 Capacidad de generar nuevas ideas (creatividad)
- T3.7 Habilidad de realizar trabajo autónomo



4.5. Resultados** del aprendizaje de la asignatura

Al finalizar con éxito esta asignatura, el alumno deberá ser capaz de:

1. Conocer los principios básicos de arquitectura de computadores, sistemas operativos, y los tipos de lenguajes de programación.
2. Aplicar los mecanismos básicos de construcción de algoritmos en el paradigma de la programación estructurada para el diseño de programas.
3. Conocer los mecanismos básicos del lenguaje de programación C para representar tipos de datos (tanto primitivos como compuestos).
4. Desarrollar programas utilizando estructuras de control de la programación estructurada, y transformar a dicho lenguaje los diseños realizados.
5. Desarrollar programas utilizando funciones y los mecanismos de paso de parámetros.
6. Conocer las características básicas de una base de datos y los programas de computador más comunes en el ámbito de la ingeniería de organización.

Las actividades de enseñanza/aprendizaje diseñadas permitirán al alumno desarrollar su capacidad de: trabajo en equipo, análisis y síntesis de información, expresión escrita y comunicación oral mediante el desarrollo de un programa de ordenador y su defensa oral al finalizar el cuatrimestre.

**** Véase también la *Guía de apoyo para la redacción, puesta en práctica y evaluación de los resultados del aprendizaje*, de ANECA:**

http://www.aneca.es/content/download/12765/158329/file/learningoutcomes_v02.pdf



5. Contenidos

5.1. Contenidos del plan de estudios asociados a la asignatura

Programación estructurada de aplicaciones informáticas. Lenguajes de programación. Edición y compilación de programas. Estructura y funciones de un sistema operativo. Tipos de sistemas operativos. Administración básica de sistemas operativos. Bases de Datos relacionales. Modelos de Datos. Herramientas de gestión de bases de datos. Componentes de un sistema informático. Categorías de aplicaciones informáticas. Recursos utilizados en un sistema informático. Aplicaciones informáticas habituales en ámbito ingenieril.

5.2. Programa de teoría (unidades didácticas y temas)

UD 1. Sistemas Informáticos y Arquitectura de Computadores

Tema 1. Introducción a los sistemas informáticos: estructura básica de un computador y sistemas operativos.

Tema 2. Representación de la información.

UD 2. La Programación Estructurada en C

Tema 3. Introducción a la programación estructurada y la algoritmia.

Tema 4. Tipos de datos primitivos.

Tema 5. Operaciones de entrada/salida.

Tema 6. Estructuras de control: selección y repetición

Tema 7. Abstracción Funcional. Paso de parámetros por valor.

Tema 8. Tipos Estructurados de Datos: Vectores numéricos

Tema 9. Tipos Estructurados de Datos: Caracteres y cadenas de caracteres.

Tema 10. Direcciones y punteros. Paso de parámetros por referencia.

UD 3. Bases de Datos y Aplicaciones Informáticas

Tema 11. Introducción a las bases de datos.

Tema 12. Introducción a las aplicaciones informáticas en la ingeniería de organización.

5.3. Programa de prácticas (nombre y descripción de cada práctica)

Se desarrollan diez sesiones de aula de informática (de 2 horas de duración cada una) donde los alumnos además de familiarizarse con el uso de un computador y conocer las herramientas habituales para el desarrollo de programas, serán capaces de resolver y probar con la asistencia del profesor de prácticas pequeños ejemplos guiados. Estos ejemplos tienen la complejidad suficiente para poder ser seguidos sin dificultad al tiempo que refuerzan los conocimientos adquiridos en las clases de teoría/problemas.

Práctica 1. Introducción al entorno de programación: Compiladores e intérpretes.

Práctica 2. Conceptos básicos para entrada / salida por consola.

Práctica 3. Estructuras de control condicionales.

Práctica 4. Estructuras de repetición (I).

Práctica 5. Estructuras de repetición (II).

Práctica 6. Funciones. Paso de parámetros por valor.

Práctica 7. Arrays numéricos (I).

Práctica 8. Arrays numéricos (II).

Práctica 9. Cadenas de caracteres.

Práctica 10. Funciones. Paso de parámetros por referencia.



5.4. Programa de teoría en inglés (unidades didácticas y temas)

I. Computer Systems & Architecture.

1. Introduction to computer systems: Basic structure of a computer and Operating systems.
2. Information representation.

II. Structured Programming in C.

3. Introduction to structured programming and algorithms.
4. Primitive data types.
5. Input/output operations
6. Control structures: conditional and loops.
7. Data Structures (I): Arrays/Numeric Vectors/Matrices.
8. Data Structures (II): Characters and strings.
9. Functional Abstraction. Passing parameters by value.
10. Memory Addresses and Pointers. Passing parameters by reference.

III. Databases & Software applications

11. Introduction to databases.
12. Introduction to software applications in industrial management engineering.

5.5. Objetivos del aprendizaje detallados por unidades didácticas

UD 1. Sistemas Informáticos y Arquitectura de Computadores.

- Conocer la evolución tecnológica de los sistemas informáticos.
- Ser consciente de la importancia actual de los sistemas informáticos -y su interconexión en redes- para la sociedad, la industria y la defensa, así como el uso eficiente y seguro de los mismos.
- Describir los niveles de abstracción de todo sistema informático.
- Exponer la arquitectura básica de von Neumann.
- Describir las características principales y la secuencia de funcionamiento de la unidad central de procesos.
- Construir programas simples que utilicen la unidad central de procesos mediante un simulador de la misma.
- Describir el funcionamiento de la memoria de un sistema informático y los tipos de memoria existentes (magnética, óptica y de estado sólido).
- Conocer las diferentes unidades de entrada/salida de un sistema informático.
- Enumerar los buses de un sistema informático y principales tipos de periféricos.
- Definir los programas encargados de controlar el funcionamiento de un sistema informático: sistemas operativos.
- Conocer la historia de los sistemas operativos y los existentes en la actualidad.
- Conocer las funciones principales de todo sistema operativo y su estructura.
- Conocer los distintos mecanismos de representación de la información en un computador digital.
- Enumerar las principales bases numéricas que se utilizan en informática
- Realizar la conversión entre distintas bases.
- Ser capaz de codificar números enteros empleando diferentes formatos.
- Describir las formas de codificar números reales y caracteres.



UD 2. La Programación Estructurada en C.

- Explicar los principios básicos y las fases del desarrollo de programas.
- Enumerar los principales paradigmas de programación existentes.
- Distinguir entre lenguajes de programación de alto y bajo nivel de abstracción y entre lenguajes compilados e interpretados.
- Conocer las herramientas necesarias para el desarrollo de programas.
- Explicar los principios de la programación imperativa y estructurada.
- Explicar los mecanismos básicos para la construcción de algoritmos en el paradigma de la programación estructurada.
- Construir algoritmos a partir de enunciados de problemas.
- Definir las características de los tipos de datos.
- Conocer los tipos de datos primitivos que ofrece el lenguaje C.
- Seleccionar el tipo de datos más adecuado a las necesidades de un problema.
- Ser capaz de declarar y usar variables y constantes.
- Aplicar los operadores aritméticos y lógicos y las reglas de precedencia de los operadores.
- Explicar las conversiones de tipos explícitas e implícitas.
- Resolver distintas expresiones aritméticas y lógicas en C.
- Enumerar las siete estructuras de control disponibles en C.
- Distinguir entre la selección única, la selección con alternativa y la selección múltiple.
- Describir el funcionamiento de las estructuras de selección if, if else, y switch y solucionar con ellas problemas sencillos.
- Distinguir las diferentes formas de repetición que aparecen en los algoritmos estudiados con anterioridad.
- Explicar el funcionamiento de las estructuras de repetición while y do-while, describir las diferencias entre ellas y resolver algoritmos sencillos.
- Explicar el funcionamiento de la estructura de repetición for, ser capaces de configurar las expresiones que usa y resolver con ella algoritmos sencillos.
- Interpretar la necesidad de alterar en ocasiones la repetición en curso y explicar el funcionamiento de las instrucciones break y continue.
- Modificar la programación de las estructuras de repetición para evitar el uso de ambas instrucciones.
- Utilizar las estructuras de control combinándolas en apilamiento y anidamiento, para resolver algoritmos complejos.
- Explicar las características de una estructura estática de datos y las posibilidades que ofrece.
- Conocer los mecanismos que ofrece el lenguaje C para definir estructuras estáticas de datos (*arrays*).
- Aplicar la sintaxis del lenguaje C para crear *arrays*.
- Explicar el procedimiento básico para recorrer *arrays* y acceder o modificar su contenido.
- Ser capaces de crear de *arrays* multidimensionales en el lenguaje C.
- Explicar los mecanismos básicos de manipulación de datos en *arrays*.
- Explicar los algoritmos más conocidos para la ordenación de datos en *arrays*.
- Exponer la necesidad de modularizar los programas de tamaño medio.
- Explicar el concepto de módulo y su correspondencia con la definición de una función en la programación estructurada.
- Explicar las partes que constituyen una función y la declaración de funciones en C.
- Clasificar los distintos métodos de pasos de parámetros (por valor y por referencia).



- Dar ejemplos demostrativos de declaración, invocación e implementación de funciones.
- Explicar los mecanismos de definición de funciones recursivas.
- Explicar los mecanismos básicos de referenciado de memoria en el lenguaje C.
- Distinguir entre la declaración de variables y la declaración de punteros a variables.
- Explicar la manera en la que se gestionan las direcciones de memoria y la aritmética básica de punteros.
- Explicar la manera en la que los punteros participan en el paso de parámetros por referencia.
- Explicar la reserva de memoria y su relación con el manejo de estructuras de datos estáticas y con cadenas de caracteres.
- Dar ejemplos del paso de parámetros a funciones utilizando punteros

UD 3. Bases de Datos y Aplicaciones Informáticas.

- Explicar las características básicas de una base de datos.
- Exponer las ventajas del uso de un enfoque de bases de datos.
- Conocer los modelos de datos más empleados.
- Conocer los fundamentos del lenguaje de manipulación de bases de datos SQL.
- Familiarizarse con las principales herramientas informáticas en el ámbito de la ingeniería de organización industrial.



6. Metodología docente

6.1. Metodología docente*

Actividad*	Técnicas docentes	Trabajo del estudiante	Horas
Clase de Teoría	Clase expositiva, incorporando técnicas de aprendizaje cooperativo de corta duración. Resolución de dudas planteadas por los estudiantes. Orientación a los alumnos de los recursos documentales y multimedia disponibles de soporte al aprendizaje.	<u>Presencial</u> : Asistencia a clase.	25
		<u>No presencial</u> : Estudio personal	25
Clase de Problemas	Clase expositiva, incorporando técnicas de aprendizaje cooperativo de corta duración. Se resolverán problemas tipo y se analizarán casos prácticos. Los problemas se plantearán con dificultad creciente. Se plantearán problemas y/o casos prácticos similares para que los alumnos lo vayan resolviendo	<u>Presencial</u> : Asistencia a clase.	10
		<u>No presencial</u> : Resolución de ejercicios en trabajo personal.	25
Clase de Prácticas. Sesiones de aula de informática	Las sesiones prácticas en el laboratorio de informática son fundamentales para enlazar los contenidos teóricos y prácticos de forma directa. A lo largo de las distintas sesiones, los alumnos deberán resolver diferentes colecciones de problemas guiados paso a paso por el profesor.	<u>Presencial</u> : Trabajo personal delante del ordenador supervisado por el profesor	20
Trabajos individuales	Trabajos a realizar individualmente y a entregar en Aula virtual (de carácter opcional).	<u>No Presencial</u> : Realización opcional de una serie de trabajos planteados por el profesor.	12
Tutorías (individual/grupal)	Las tutorías serán individuales o grupales, donde se atenderán las dudas/cuestiones del alumnado.	<u>Presencial</u> : Tutorías de consulta de dudas de teoría y problemas.	2
		<u>No presencial</u> : Planteamiento de dudas por correo electrónico.	
Seminario	Se realizará un seminario donde experto(s) del sector de la Informática y las TICs puedan mostrar al alumno los fundamentos y distintos casos prácticos sobre aplicaciones informáticas y bases de datos en el ámbito de la ingeniería de organización.	<u>Presencial</u> : Asistencia al seminario	2
Actividades de evaluación sumativa	Además del examen oficial final del cuatrimestre, se realizarán dos pruebas escritas de tipo individual.	<u>Presencial</u> : Realización de prueba escrita.	2
Realización de exámenes oficiales	Prueba escrita individual. Acorde a la normativa vigente.	<u>Presencial</u> : Realización de prueba escrita.	2
Trabajo final de asignatura.	Resolución de un problema complejo programado en C. (De carácter opcional)	<u>Presencial</u> : defensa del trabajo en una entrevista personal	~0
		<u>No presencial</u> : Implementación de la práctica final.	25
			150



6.2. Resultados (4.5) / actividades formativas (6.1) (opcional)

Resultados del aprendizaje (4.5)						
Actividades formativas (6.1)	1	2	3	4	5	6
Clase de Teoría	x	x	x	x	x	x
Clase de Problemas		x	x	x	x	
Clase de Prácticas. Sesiones de aula de informática		x	x	x	x	
Trabajos individuales		x	x	x	x	
Tutorías (individuales/grupales)	x	x	x	x	x	x
Seminario						x
Trabajo final de asignatura		x	x	x	x	

7. Metodología de evaluación

7.1. Metodología de evaluación*

Actividad	Tipo		Sistema y criterios de evaluación*	Peso (%)	Resultados (4.5) evaluados
	Sumativa*	Formativa*			
Prueba escrita individual en convocatoria oficial	x		Entre 5 y 10 problemas. Se evalúa la capacidad de aplicar los conocimientos adquiridos a la asignatura. En dicha prueba, se incluirán cuestiones relacionadas con las prácticas de laboratorio. <i>Es necesaria una calificación mínima de 4 sobre 10 en esta prueba final para superar la asignatura y considerar el resto de evaluaciones de la asignatura.</i>	65%	3-6
Prueba de evaluación sumativa	x		Problemas y ejercicios de complejidad similar a los resueltos en la bibliografía y en clase.	25%	1-4
Trabajo final de asignatura	x	x	Se propondrá un trabajo final, donde cada alumno deberá aplicar los distintos conocimientos adquiridos de programación en C.. <i>La evaluación del trabajo final se realizará mediante entrevista personal después del examen final y utilizando una rúbrica.</i>	10% y hasta 0,5 puntos adicionales	2-5

IMPORTANTE: Para superar la asignatura deberá conseguir una calificación igual o superior a 5, teniendo en cuenta los todos los criterios y pruebas de evaluación de la asignatura.

7.2. Mecanismos de control y seguimiento (opcional)

El seguimiento del aprendizaje se realizará mediante las siguientes actividades:

- Seguimiento de las entregas de trabajo individual optativo.
- Resultados de la prueba intermedia.
- Seguimiento del trabajo del alumno en prácticas.
- Estadísticas del uso del material documental colocado en el Aula Virtual.
- Tutorías individuales y grupales.



8 Bibliografía y recursos

8.1. Bibliografía básica*

- *Informática Aplicada. Programación en C*. Pedro M. Alcover Garau. Edita la Universidad Politécnica de Cartagena, 2012. (<http://hdl.handle.net/10317/1361>)
- *Informática Aplicada. Prácticas para aprender a programar en Lenguaje C*. Pedro J. García Laencina, Pedro M. Alcover Garau. Edita el Centro Universitario de la Defensa de San Javier, 2012. (<http://hdl.handle.net/10317/2740>)

8.2. Bibliografía complementaria*

Muy recomendados:

- *Introducción a la Informática*. Alberto Prieto y otros. Mc. Graw Hill, 4ª edición, 2006.
- *El lenguaje de programación C*, B. Kernighan y D. Ritchie, Editorial Prentice Hall, 1991.
- *Programación estructurada en C*, J. Antonakos y K. Mansfield, Editorial Prentice Hall, 2004.
- *Bases de datos relacionales*, Matilde Celma Giménez, Editorial Prentice Hall, 2003.
- *Enciclopedia de Microsoft. Visual C#*, Fco. J. Ceballos, Editorial RAMA, 3ª Ed., 2010.
- *Matlab: A Practical Introduction to Programming and Problem Solving*, S. Attaway, Editorial Butterworth-Heinemann 2nd Edition, 2011

Para ampliar conocimientos:

- *Organización de computadoras: un enfoque estructurado*, A. Tanenbaum, Editorial Pearson Education, 2000.
- *Organización y arquitectura de computadores*, W. Stallings, Editorial Prentice-Hall, 2006.
- *Sistemas operativos: aspectos internos y principios de diseño*, W. Stallings, Editorial Pearson Prentice Hall, 2005.

8.3. Recursos en red y otros recursos

Todo el material de la asignatura está disponible en Aula Virtual:

<https://aulavirtual.upct.es/course/view.php?id=599>

