



Universidad  
Politécnica  
de Cartagena



Centro  
Universitario  
de la Defensa

Guía docente de la asignatura:

## **Informática**

**Titulación:**

**Grado en Ingeniería de Organización Industrial**

## 1. Datos de la asignatura

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

<b>Profesor responsable</b>	Nina Skorin-Kapov		
<b>Departamento</b>	Departamento de Ingeniería y Tecnología Aplicada		
<b>Área de conocimiento</b>	Ingeniería Telemática		
<b>Ubicación del despacho</b>	Despacho 7, Edificio Administrativo Del CUD		
<b>Teléfono</b>	968189911	<b>Fax</b>	968188780
<b>Correo electrónico</b>	<a href="mailto:nina.skorinkapov@ cud.upct.es">nina.skorinkapov@cud.upct.es</a>		
<b>URL / WEB</b>	Aula Virtual UPCT		
<b>Horario de atención / Tutorías</b>	<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>		
<b>Ubicación durante las tutorías</b>	Despacho 7, Edificio Administrativo del CUD		

<b>Titulación</b>	Licenciatura en telecomunicaciones por la Universidad de Zagreb, Croacia; Doctora por la Universidad de Zagreb, Croacia (homologado por la UPCT) Acreditación ANECA Profesor Titular de Universidad
<b>Vinculación con CUD-UPCT</b>	Profesor Contratado Doctor en centro adscrito
<b>Año de ingreso en CUD-UPCT</b>	2013
<b>Nº de quinquenios (si procede)</b>	2
<b>Líneas de investigación (si procede)</b>	Optimización y planificación de redes de comunicaciones, seguridad de redes ópticas, algoritmos (meta)-heurísticos
<b>Nº de sexenios (si procede)</b>	2 sexenios acreditado por CENAI (2004-2009, 2010-2015)
<b>Experiencia profesional (si procede)</b>	<p>Universidad de Zagreb, Croacia, 2003-2013 (2003-6 eqv. Ayudante, 2006-7 eqv. Ayudante Doctor, 2008-12 eqv. Contratado Doctor, 2012-13, eqv. Titular de Universidad)</p> <p>Telecom ParisTech (Ecole nationale supérieure des télécommunications), Paris, Francia, 2006-7 (Postdoc)</p>
<b>Otros temas de interés</b>	Ver: <a href="https://sites.google.com/site/ninaskorinkapov/">https://sites.google.com/site/ninaskorinkapov/</a>

<b>Profesor 2</b>	María Teresa Martínez Inglés		
<b>Departamento</b>	Departamento de Ingeniería y Tecnología Aplicada		
<b>Área de conocimiento</b>	Telecomunicaciones		
<b>Ubicación del despacho</b>	Despacho 2 - Edificio Administrativo del CUD		
<b>Teléfono</b>	968189916 (ext 2916)	<b>Fax</b>	968188780
<b>Correo electrónico</b>	<a href="mailto:mteresa.martinez@ cud.upct.es">mteresa.martinez@cud.upct.es</a>		
<b>URL / WEB</b>	Aula Virtual de la UPCT		
<b>Horario de atención / Tutorías</b>	<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>		
<b>Ubicación durante las tutorías</b>	Despacho 2- Edificio Administrativo del CUD		

<b>Titulación</b>	Doctor Ingeniero de Telecomunicación. Área de Tecnologías de la Información y Comunicaciones. Acreditación Profesor Contratado Doctor.
<b>Vinculación con CUD-UPCT</b>	Profesor Ayudante Doctor en centro adscrito.
<b>Año de ingreso en CUD-UPCT</b>	2016
<b>Nº de quinquenios (si procede)</b>	
<b>Líneas de investigación (si procede)</b>	<p>Medidas Experimentales del canal de radiocomunicaciones.</p> <p>Caracterización de modelos de propagación banda ancha basados en trazados de rayos para canales MIMO en entornos microcelulares.</p> <p>Aplicación Teoría Uniforme de la Difracción al desarrollo de modelos de propagación</p>
<b>Nº de sexenios (si procede)</b>	
<b>Experiencia profesional (si procede)</b>	<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.</p> <p>UPCT: Ingeniero. Estudio de viabilidad del despliegue de un sistema cognitivo y desarrollo de simuladores IMS (IP Multimedia Subsystems).</p>
<b>Otros temas de interés</b>	

<b>Profesor 3</b>	Fernando Pereñíguez García		
<b>Departamento</b>	Departamento de Ciencias e Informática		
<b>Área de conocimiento</b>	Ingeniería Telemática		
<b>Ubicación del despacho</b>	Despacho 38 – Edificio Administrativo del CUD		
<b>Teléfono</b>	968189946 (ext 2946)	<b>Fax</b>	968189970
<b>Correo electrónico</b>	<a href="mailto:fernando.pereniguez@ cud.upct.es">fernando.pereniguez@cud.upct.es</a>		
<b>URL / WEB</b>	Aula Virtual de la UPCT		
<b>Horario de atención / Tutorías</b>	Martes y Jueves 12:35-14:35 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.		
<b>Ubicación durante las tutorías</b>	Despacho 38 – Edificio Administrativo del CUD		

<b>Titulación</b>	Ingeniero en Informática; Máster en Tecnologías de la Información y Telemáticas Avanzadas; Doctor Ingeniero Informático. Acreditación ANECA Profesor Titular de Universidad, Área Ingeniería Telemática.
<b>Vinculación con CUD-UPCT</b>	Profesor Ayudante Doctor en centro adscrito.
<b>Año de ingreso en CUD-UPCT</b>	2017
<b>Nº de quinquenios (si procede)</b>	
<b>Líneas de investigación (si procede)</b>	Control de acceso a la red y modelos de distribución de claves en redes móviles. Handoffs transparentes en redes heterogéneas. Soluciones de federación para acceso Single-Sign-On (SSO) a servicios de aplicación. Seguridad en redes de comunicaciones.
<b>Nº de sexenios (si procede)</b>	Un sexenio acreditado por CENAI (2008-2013)
<b>Experiencia profesional (si procede)</b>	- Universidad de Murcia, 2011-2013, investigador contratado en proyectos I+D nacionales y europeos. - Universidad de Murcia, 2012-2013, profesor asociado. - Universidad Católica de Murcia, 2013-2017 (2013-14 Ayudante Doctor, 2014-15 Contratado Doctor, 2016-17 Profesor Titular de Universidad)
<b>Otros temas de interés</b>	Ciberseguridad y ciberdefensa.

### 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

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

CG2 - Aplicar las tecnologías generales y las materias fundamentales en el ámbito industrial para la resolución de problemas propios de la ingeniería.

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

CE3 - Usar y programar los 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

CT7 - Diseñar y emprender proyectos innovadores.

### 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.

**\*\* 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](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. Introducción a la programación estructurada y la algoritmia.

#### UD 2. La Programación Estructurada en C

Tema 3. Representación de la información.

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 9 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 refuercen los conocimientos adquiridos en las clases de teoría/problemas.

*Práctica 1. Introducción al entorno de programación. Conceptos básicos para entrada / salida por consola.* En esta práctica introductoria se familiariza al alumno con el entorno de programación y se explican las operaciones básicas de entrada/salida de datos.

*Práctica 2. Estructuras de control condicionales.* Esta práctica presenta las principales estructuras de control condicionales: **if**, **if-else**, **switch**.

*Práctica 3. Estructuras de repetición (I).* Esta práctica consiste de diversos ejercicios que pretenden afianzar el uso de las estructuras de repetición: **for**, **while**, **do-while**.

*Práctica 4. Estructuras de repetición (II).* En esta práctica de ampliación, además de proceder con ejercicios de mayor dificultad que la anterior práctica, se introducen las sentencias **break** y **continue**.

*Práctica 5. Funciones. Paso de parámetros por valor.* En esta práctica el alumno afianza los conceptos relacionados con el uso de funciones haciendo uso del método básico de paso de parámetros por valor.

*Práctica 6. Arrays numéricos.* En esta práctica se introduce al alumno con el manejo de arrays, que son un tipo básico de datos agregado que permiten almacenar colecciones de datos del mismo tipo.

*Práctica 7. Cadenas de caracteres.* Para complementar la formación en el uso de arrays, esta práctica versa sobre la manipulación de arrays que almacenan caracteres, también conocidos como cadenas de caracteres.

*Práctica 8. Funciones. Paso de parámetros por referencia.* En esta práctica, el alumno ampliará sus destrezas en el manejo de funciones poniendo en práctica el método avanzado de paso de parámetros por referencia, el cual que requiere el uso de punteros.

*Práctica 9. Apoyo a la resolución del Trabajo Final.* En esta última práctica se realizará una presentación del Trabajo Final de la asignatura y se proporcionarán orientaciones que sirvan de ayuda al alumno para su resolución.

## **Prevención de riesgos**

La Universidad Politécnica de Cartagena considera como uno de sus principios básicos y objetivos fundamentales la promoción de la mejora continua de las condiciones de trabajo y estudio de toda la Comunidad Universitaria.

Este compromiso con la prevención y las responsabilidades que se derivan atañe a todos los niveles que integran la Universidad: órganos de gobierno, equipo de dirección, personal docente e investigador, personal de administración y servicios y estudiantes.

El Servicio de Prevención de Riesgos Laborales de la UPCT ha elaborado un "Manual de acogida al estudiante en materia de prevención de riesgos" que puedes encontrar en el Aula Virtual, y en el que encontraras instrucciones y recomendaciones acerca de cómo actuar de forma correcta, desde el punto de vista de la prevención (seguridad, ergonomía, etc.), cuando desarrolles cualquier tipo de actividad en la Universidad. También encontrarás recomendaciones sobre cómo proceder en caso de emergencia o que se produzca algún incidente.

En especial, cuando realices prácticas docentes en laboratorios, talleres o trabajo de campo, debes seguir todas las instrucciones del profesorado, que es la persona responsable de tu seguridad y salud durante su realización. Consúltale todas las dudas que te surjan y no pongas en riesgo tu seguridad ni la de tus compañeros.

## **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. Introduction to structured programming and algorithms.

### **II. Structured Programming in C.**

3. Information representation.
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.
- 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.
- Conocer las diferentes unidades de entrada/salida de un sistema informático.
- Enumerar los buses de un sistema informático y principales tipos de periféricos.
- Conocer las funciones principales de sistema operativo y su estructura.
- Explicar los principios básicos y las fases del desarrollo de programas.
- Explicar los mecanismos básicos para la construcción de algoritmos en el paradigma de la programación estructurada.
- Construir algoritmos a partir de enunciados de problemas.

### **UD 2. La Programación Estructurada en C.**

- 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.
- Definir las características de los tipos de datos.
- Conocer los tipos de datos primitivos que ofrece el lenguaje C.
- Ser capaz de declarar y usar variables y constantes.
- Aplicar los operadores aritméticos y lógicos y las reglas de precedencia de los operadores.
- Resolver distintas expresiones aritméticas y lógicas en C.
- 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.
- Utilizar las estructuras de control combinándolas en apilamiento y anidamiento, para resolver algoritmos complejos.
- 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 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.
- Distinguir entre la declaración de variables y la declaración de punteros a variables.
- Explicar la manera en la que los punteros participan en el paso de parámetros por referencia.
- 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.
- 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
Clases teóricas en el aula	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 y participación a las clases presenciales. Toma de apuntes. Planteamiento de dudas.	<b>25</b>
Clases de problemas en el aula	Clase expositiva, incorporando técnicas de aprendizaje cooperativo de corta duración. Se resolverán problemas tipo y se analizarán casos prácticos. Se plantearán problemas y/o casos prácticos similares para que los alumnos lo vayan resolviendo	<u>Presencial</u> : Asistencia y participación a las clases presenciales. Toma de apuntes. Planteamiento de dudas.	<b>10</b>
Sesiones Prácticas en 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 con la ayuda del profesor.	<u>Presencial</u> : Trabajo personal delante del ordenador supervisado por el profesor	<b>18</b>
		<u>No presencial</u> : Preparación de ejercicios de programación	<b>2</b>
Tutorías	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.	<b>1</b>
		<u>No presencial</u> : Preparación de dudas para la tutoría presencial	<b>1</b>
Asistencia a Seminarios	Se realizará un seminario sobre aplicaciones informáticas y bases de datos en el ámbito de la ingeniería de organización.	<u>Presencial</u> : Asistencia al seminario	<b>2</b>
Trabajo/Estudio Individual	Se proporcionará al alumno apuntes y ejemplos de todos los temas de la asignatura para facilitar el estudio personal.	<u>No presencial</u> : Estudio personal; Resolución de ejercicios en trabajo personal.	<b>50</b>
Preparación Trabajos/Informes	Se proporcionará el enunciado del trabajo final de la asignatura.	<u>No presencial</u> : Desarrollo individual del trabajo final de la asignatura	<b>12</b>
Preparación Trabajos/Informes en grupo	Se proporcionará el enunciado del trabajo final de la asignatura. Se realizará una clase de presentación y apoyo a la resolución del trabajo final de la asignatura.	<u>Presencial</u> : Asistencia a la clase de presentación y apoyo al trabajo final; Entrevista personal	<b>6,2</b>
		<u>No presencial</u> : Desarrollo del trabajo final de la asignatura	<b>18,8</b>
Realización de actividades de evaluación formativas y sumativas	Preparación de las pruebas escritas de evaluación	<u>Presencial</u> : Realización de pruebas escritas.	<b>2</b>
Realización de exámenes oficiales	Preparación del examen final	<u>Presencial</u> : Realización de examen final	<b>2</b>
			<b>150</b>

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

### Resultados del aprendizaje (4.5)

Actividades formativas (6.1)	1	2	3	4	5	6
Clases teóricas en el aula	x	X	x	x	x	X
Clases de problemas en el aula		X	x	x	x	
Sesiones Prácticas en Aula de Informática		X	x	x	x	
Tutorías	x	X	x	x	x	x
Asistencia a Seminarios						x
Trabajo/Estudio Individual	x	X	x	x	x	x
Preparación Trabajos/Informes		X	x	x	x	
Preparación Trabajos/Informes en grupo		X	x	x	x	
Realización de actividades de evaluación formativas y sumativas	x	X	x	x	x	x
Realización de exámenes oficiales	x	X	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*			
<b>Prueba escrita parcial</b>	x	x	Problemas y ejercicios de complejidad similar a los resueltos en la bibliografía y en clase. La prueba escrita versará sobre los Temas 1-4	20% <sup>(*)</sup>	1-3
<b>Examen Final Parte B1</b>	x		Problemas y ejercicios de complejidad similar a los resueltos en la bibliografía y en clase. La prueba escrita versará sobre los Temas 1-4	20% <sup>(*)</sup>	1-3
<b>Examen Final Parte B2</b>	x		Problemas y ejercicios de complejidad similar a los resueltos en la bibliografía y en clase. La prueba escrita versará sobre los Temas 4-12	70% <sup>(*)</sup>	3-6
<b>Trabajo Final (TF) de asignatura</b>	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 dentro del periodo de exámenes de la convocatoria de Junio y utilizando una rúbrica(**).</i>	10% y hasta 0,5 puntos adicionales	2-5

(\*) Habrá una prueba escrita individual (PEI\_B1) parcial que versará sobre los Temas 1-4. Debe superarse con nota total igual o superior a 4 sobre 10 para eliminar esta parte de la asignatura de cara al examen final.

La evaluación final constará de dos Pruebas Escritas Individuales (PEIs); en primer lugar, todos los alumnos realizarán la PEI correspondiente los temas 4-12 (PEI\_B2). Tras un descanso, se realizará la PEI correspondiente a los Temas 1-4 (PEI\_B1) para aquellos alumnos que no liberaron dicha materia en la PEI\_B1 parcial, o aquellos que deseen mejorar su calificación. Todos los alumnos se pueden presentar a esta segunda PEI, teniendo en cuenta que, aquellos que liberaron materia en la prueba parcial, si entregan esta parte al profesor, perderán la calificación obtenida en la PEI parcial.

Para aprobar la asignatura es necesario, pero no suficiente, obtener un mínimo de 4.0 en cada PEI. Si esta condición no se verifica, el alumno suspenderá la asignatura completa, siendo la nota máxima en esta situación 4.0 – Suspenso. Para aclarar este punto, supongamos un alumno que obtiene en PEI\_B1 un 10.0, en PEI\_2 un 3.0, y un 10 en el Trabajo Final (TF), la calificación media es 5.1 pero como no ha llegado al 4.0 en la PEI\_B1 aparecerá en el acta de la asignatura con 4.0 – Suspenso. El trabajo final (TF) no tiene nota mínima.

**La calificación de la asignatura (N) se calculará según la siguiente expresión:**

$$N=0,2*(PEI\_B1)+0,7*(PE\_B2)+0,1*TF$$

**Para aprobar la asignatura es necesario obtener un mínimo de 4.0 en parte PEI\_B1, un mínimo de 4.0 en parte PEI\_B2, y en la calificación final de la asignatura N un mínimo de 5.0**

(\*\*) El Trabajo Final de la asignatura será entregado en las fechas que indiquen los profesores dentro de la convocatoria de exámenes de Junio del presente curso. La puntuación de este trabajo se guardará sólo para la siguiente convocatoria de exámenes, es decir, sólo se mantendrá esa nota para la convocatoria de Agosto del presente curso académico.

Tal como prevé el artículo 5.4 del *Reglamento de las pruebas de evaluación de los títulos oficiales de grado y de máster con atribuciones profesionales* de la UPCT, el estudiante en el que se den las circunstancias especiales recogidas en el Reglamento, y previa solicitud justificada al Departamento y admitida por este, tendrá derecho a una prueba global de evaluación. Esto no le exime de realizar los trabajos obligatorios que estén recogidos en la guía docente de la asignatura.

## 7.2. Mecanismos de control y seguimiento (opcional)

El seguimiento del aprendizaje se realizará mediante las siguientes actividades:

- Resultados de la prueba intermedia.
- Seguimiento del trabajo del alumno en prácticas.
- Seguimiento de las entregas de trabajo individual.
- 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