



Grado en Fundamentos  
de la Arquitectura  
Curso 2016/2017

IDI108

**Fundamentos de  
Informática**



**Asignatura: Fundamentos de Informática**

**Carácter: Básica**

**Idioma: Español**

**Modalidad: Presencial**

**Créditos: 6**

**Curso: Segundo**

**Semestre: Primero**

**Grupo: 2FARQ**

**Curso académico: 2016/2017**

**Profesores/Equipo Docente: Jorge Martínez Ladrón de Guevara**

## **1. REQUISITOS PREVIOS**

- Es necesario que el alumno tenga conocimientos elementales del sistema operativo Windows y de Internet.

## **2. BREVE DESCRIPCIÓN DE CONTENIDOS**

- Introducción a la Informática.
- Arquitectura básica de ordenadores.
- Introducción a la programación.
- Programación estructurada
- Conceptos básicos de programación orientada a objetos con Java.

## **3. RESULTADOS DEL APRENDIZAJE**

Las clases de teoría y práctica serán la base sobre las que el alumno adquirirá los conocimientos y la **capacidad de aplicar con criterio** los principios y conceptos básicos de informática, así como otras competencias como la **capacidad de comunicarse** utilizando correctamente el lenguaje científico-técnico y le faculte finalmente para aprender por si mismo otros conceptos y aplicaciones de la informática (**autoaprendizaje**). También el estudio individual junto con la necesidad de buscar información por Internet, la enseñaran a aprender por si mismo, (capacidad de autoaprendizaje) lo que podrá utilizar para profundizar en esta materia y también le ayudará en otras como Diseño Asistido por Ordenador donde ya habrá asumido el hábito de explorar menús, uso de ayudas, manejo de ficheros, búsqueda de librerías y recursos software especializadas en Internet etc.

El objetivo de la asignatura es que el alumno conozca los fundamentos de la informática, aunque la mayor parte de la asignatura consistirá en el diseño de algoritmos y su programación en Java.

- Los estudiantes habrán demostrado poseer y comprender los conocimientos relativos a la introducción a la Informática, arquitectura básica de ordenadores y una Introducción a la programación.

- Los estudiantes tendrán la capacidad para reunir los datos necesarios para el desarrollo de aplicaciones informáticas básicas aplicando los juicios y criterios que garanticen su buen funcionamiento.
- Los estudiantes podrán transmitir estas soluciones informáticas, empleando con soltura los conceptos e ideas adquiridos en esta materia.
- Los estudiantes habrán desarrollado habilidades de aprendizaje en el campo de la Informática que les permitan aplicarlas a asignaturas posteriores, así como a proyectos en su vida profesional, con un alto grado de autonomía.

## 4. ACTIVIDADES FORMATIVAS Y METODOLOGÍA

**Clases de teoría y prácticas:** (1,8 ECTS) Apoyándose en transparencias el profesor explica los conceptos y plantea los ejemplos y problemas. El material presentado en las transparencias de clase no constituirá la totalidad del contenido. El alumno deberá completar sus conocimientos con referencias bibliográficas, enlaces por Internet, tutoriales y ayuda del entorno integrado de desarrollo. El profesor propondrá a los alumnos el desarrollo de un programa de ordenador, iniciado y desarrollado parcialmente en las propias sesiones presenciales en aula informática y completado por el alumno en su trabajo individual.

**Tutorías:** (0.6 ECTS) Consulta al profesor por parte de los alumnos sobre la materia en los horarios de tutorías o empleando mecanismos de tutoría telemática (correo electrónico y uso del campus virtual de la Universidad).

**Estudio individual y trabajo de asignatura:** (3,6 ECTS) Estudio individual del alumno utilizando los apuntes y programas explicados en clase, libros de la biblioteca, o apuntes del profesor o módulos de software disponibles en el campus virtual. Para facilitar el aprendizaje y la realización de las prácticas de programación, el alumno puede acceder, en un horario amplio, a las aulas informáticas donde dispondrá de los entornos de desarrollo y ordenadores con todos los programas informáticos de la asignatura. Deben entregar el programa final al profesor para ser evaluado.

## 5. SISTEMA DE EVALUACIÓN

**Prueba escrita:** Se realizarán dos exámenes escritos, uno parcial (que no libera materia) y otro final, donde se evaluarán:

- El aprendizaje de los contenidos adquiridos por el alumno en las clases de teoría y prácticas de programación, en las tutorías y en su estudio individual.

- La utilización adecuada de los términos informáticos, así como el acierto en la estructuración de la información, aplicando con criterio las técnicas, y algoritmos de programación descritos en las clases y requeridos en el examen.

El examen parcial pondera un 15% y el final un 60% de la nota final en la convocatoria ordinaria.

**Evaluación de la participación del alumno en clase y del trabajo de asignatura obligatorio:** La participación del alumno será valorada por el profesor a lo largo de las clases. Junto con el trabajo de programación, este capítulo ponderará el 25 % restante de la nota final. Nuevamente se evaluará no solo los conocimientos sino la adquisición de competencias en su conjunto, tales como la calidad de la expresión y aptitud del alumno para comunicar, expresada por escrito en sus trabajos y verbalmente en sus intervenciones y participación en clase.

La ponderación tanto del examen parcial como de los conceptos de participación y trabajos de programación, solo se aplicará si el alumno obtiene al menos un 4.5 en el examen final. Una nota inferior a 4 en el trabajo de programación supone el suspenso automático de la asignatura en la convocatoria ordinaria.

En la **convocatoria extraordinaria** el examen pondera un 90% y la participación y el trabajo de programación un 10 %. Esta ponderación también se aplica solo en el caso de que el alumno obtenga al menos un 4.5 en este examen final extraordinario.

## 6. BIBLIOGRAFÍA

### Bibliografía básica

- Título: Java. Manual de referencia. Autor: Herbert Schildt, Editorial: Mc Graw Hill, 2001.

### Bibliografía Complementaria

- Título: Java 2 Iniciación y Referencia. Autor: Jesús Sánchez Allende, Editorial: Mc Graw Hill, 2005.
- J. Galve, y otros, "Algorítmica diseño y análisis de algoritmos funcionales e imperativos". Ra-Ma, 1993.
- A. Gómez Vieites y M. Veloso Espiñeira, "Redes de ordenadores e Internet", Ra-Ma, 2003.
- Prieto: "Introducción a la informática". McGraw-Hill, 2002.
- C. Horstmann y G. Cornell, "Java 2. Vol I. Fundamentos", Prentice Hall, 2003.
- Programación en Java 2. Serie Schaum. Jesús Sánchez Allende y otros. McGraw-Hill. 2005.
- Domine HTML y DHTML, José López Quijado. Ed. RA-MA.

## 7. BREVE CURRICULUM

### **Jorge Martínez Ladrón de Guevara**

#### **Profesor departamento de Informática**

Ingeniero en Sistemas Computacionales por el Tecnológico de Monterrey (México). Durante más de doce años ha trabajado en proyectos de desarrollo de software en empresas de ingeniería. Responsable del desarrollo de sistemas informáticos de gestión para empresas del sector eléctrico y de la defensa. Desde 2002 es profesor en programas de postgrado de ICADE Business School. Desde 2004 es profesor asociado en la Universidad Nebrija, donde imparte clases de Fundamentos de la Informática.

## 8. LOCALIZACIÓN DEL PROFESOR

#### **Profesor de la asignatura**

##### **Jorge Martínez Ladrón de Guevara**

Departamento de Ingeniería Informática

E-mail: [jmartine@nebrija.es](mailto:jmartine@nebrija.es)

Tfno.: +34 - 91.452.11.00

#### **Coordinador de la asignatura**

Dr. D. Juan Antonio Maestro

Escuela Politécnica Superior

Despacho 406

E-Mail: [jmaestro@nebrija.es](mailto:jmaestro@nebrija.es)

Web: <http://www.nebrija.es/~jmaestro>

Tfno.: +34 - 91.452.11.00 – Extensión 5809

Tutorías: Bajo petición al correo electrónico anterior

## 9. CONTENIDO DETALLADO DE LA ASIGNATURA

TÍTULO: Grado en Fundamentos de la Arquitectura

CURSO ACADÉMICO: 2016/2017

ASIGNATURA: Fundamentos de Informática

CURSO: Segundo SEMESTRE: Primero CRÉDITOS ECTS: 6

Sesión	Sesiones de Teoría, Problemas y Evaluación continua	Estudio individual y trabajos del alumno	Horas Presenciales	Horas Estudio y Trabajo
1	Introducción a la asignatura. Arquitectura de ordenadores	Hoja ejercicios 1	1,5	2
2	Arquitectura de ordenadores		1,5	
3	Algoritmos. Diagramas de flujo y pseudocódigo	Hoja ejercicios 2	1,5	8
4	Algoritmos. Diagramas de flujo y pseudocódigo		1,5	
5	Algoritmos. Diagramas de flujo y pseudocódigo		1,5	
6	Introducción a Java. El entorno de desarrollo. Estructura de un programa java	Hoja ejercicios 3	1,5	10
7	Datos, tipos de datos, operadores y variables		1,5	
8	Expresiones y sentencias.		1,5	
9	Entrada/salida básica. Parámetros.		1,5	
10	Estructuras de control condicionales: if y switch		1,5	
11	Sesión de ejercicios		1,5	
12	Estructuras de control repetitivas/iterativas: for, while y do.		1,5	
13	Sesión de ejercicios		1,5	
14	<b>EXAMEN PARCIAL</b>		1,5	10
15	Arrays. Declaración, creación y uso	Hoja ejercicios 4	1,5	10
16	Manipulación de vectores y arrays		1,5	
17	Manipulación de vectores y arrays		1,5	
18	Subrutinas y métodos		1,5	
19	Subrutinas y métodos		1,5	
20	Caso práctico: desarrollo de aplicaciones creativas con palabras: poemas, cuentos....		1,5	
21	Sesión de ejercicios		1,5	
22	Sesión de ejercicios		1,5	
23	Caso práctico: desarrollo de aplicaciones creativas con imágenes digitales: filtros de color y textura, filtros aleatorios, fusión de imágenes....		1,5	
24	Sesión de ejercicios		1,5	
25	Sesión de ejercicios		1,5	
26	Sesión de ejercicios		1,5	
27	Sesión de ejercicios		1,5	
28	Introducción a la programación orientada a objetos		1,5	
29	Introducción a la programación orientada a objetos		1,5	
	<b>Evaluaciones Finales Ordinaria y Extraordinaria</b>		1,5	20
	<b>Tutorías</b>		15	
	<b>Laboratorio: Trabajo obligatorio de la asignatura</b>	Caso práctico: Desarrollo de una aplicación Java		20
		<b>TOTAL:</b>	<b>60</b>	<b>90</b>
		<b>TOTAL HORAS:</b>		<b>150</b>

	ECTS	Horas	Sesiones
Clases de Teoría y problemas	1,8	45	30,0
Trabajo de la asignatura	0,8	20	13,3
Tutorías	0,6	15	10,0
Estudio individual	2,8	70	46,7
<b>TOTAL</b>	<b>6</b>	<b>150</b>	

	ECTS	Horas
Presencial	2,4	60
No presencial	3,6	90
<b>TOTAL</b>	<b>6</b>	<b>150</b>