

## Grado de Ingeniería en Explotación de Minas y Recursos Energéticos

### DATOS DE LA ASIGNATURA

**Nombre:**

Fundamentos de Informática

**Denominación en inglés:**

Fundamentals of Computer Science

**Código:**

606810109

**Carácter:**

Básico

**Horas:**

	Totales	Presenciales	No presenciales
<b>Trabajo estimado:</b>	150	60	90

**Créditos:**

Grupos reducidos				
Grupos grandes	Aula estándar	Laboratorio	Prácticas de campo	Aula de informática
3	0	0	0	3

**Departamentos:**

Tecnologías de la Información

**Áreas de Conocimiento:**

Lenguaje y Sistemas Informáticos

**Curso:**

1º - Primero

**Cuatrimestre:**

Segundo cuatrimestre

### DATOS DE LOS PROFESORES

**Nombre:**

\*Polo Almohano, M<sup>a</sup> del  
Pilar

**E-Mail:**

polo@dti.uhu.es

**Teléfono:**

87386

**Despacho:**

nº 53 Edificio Torreumbría

\*Profesor coordinador de la asignatura

**1. Descripción de contenidos**

## 1.1. Breve descripción (en castellano):

**INTRODUCCIÓN A LA INFORMÁTICA**

Sistemas Informáticos y Ámbitos de Aplicación

Soporte Físico

Introducción a los Sistemas Operativos

**FUNDAMENTOS DE PROGRAMACIÓN**

Algoritmos y Lenguajes de Programación

Elementos Básicos de Programación

Flujo de Control del Programa

Estructuras de Datos.

**INTRODUCCIÓN A LAS BASES DE DATOS**

Tablas

Introducción a las Bases de Datos Relacionales

Consultas y Formularios

## 1.2. Breve descripción (en inglés):

**INTRODUCTION TO COMPUTER SCIENCE**

Computer systems and application areas

Hardware

Introduction to Operating Systems

**FUNDAMENTALS OF PROGRAMMING**

Algorithms and programming languages

Programming basics

Program control flow

Data Structures

**INTRODUCTION TO DATABASES**

Tables

Introduction to Relational Databases

Queries and forms

**2. Situación de la asignatura**

## 2.1. Contexto dentro de la titulación:

Por sus contenidos y dado el marcado carácter instrumental de la materia y que la mayor parte de las asignaturas del Grado necesitan métodos de cálculo susceptibles de ser realizados con ayuda de ordenador, observamos que podrían ser todas prácticamente las materias a las que Fundamentos de Informática (en lo sucesivo FI) apoyará desde sus contenidos, como herramienta indispensable para el egresado.

## 2.2. Recomendaciones:

El alumno debe estudiar la asignatura consultando la bibliografía sugerida por los profesores y asistir con regularidad a las tutorías que estos ofertan. Debido al marcado carácter práctico de la asignatura, se recomienda que el alumnado realice la mayor parte de los ejercicios propuestos de las relaciones de problemas.

**3. Objetivos (Expresados como resultados del aprendizaje):**

- Comprender los conceptos fundamentales sobre la estructura y organización interna de los computadores actuales, tanto a nivel físico como lógico, y familiarizarse con la terminología informática real, así como con los últimos desarrollos tecnológicos.
- Conocer los conceptos fundamentales y los fundamentos básicos necesarios para la utilización de los sistemas operativos.
- Ser capaz de realizar programas de dificultad media/baja siguiendo una o varias metodologías de descripción de algoritmos, utilizando programación estructurada.
- Ser capaz de traducir a un lenguaje de programación concreto, los programas descritos mediante la metodología utilizada.
- Desarrollar en el alumnado la capacidad de resolución de problemas.
- Adquirir conocimientos sobre la creación, manejo y consulta de bases de datos sencillas.

#### 4. Competencias a adquirir por los estudiantes

##### 4.1. Competencias específicas:

- **B03:** Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería

##### 4.2. Competencias básicas, generales o 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
- **CG01:** Capacidad para la resolución de problemas
- **CG03:** Capacidad de organización y planificación
- **CG04:** Capacidad de aplicar los conocimientos en la práctica
- **CG06:** Actitud de motivación por la calidad y mejora continua
- **CG08:** Capacidad de adaptación a nuevas situaciones
- **CG12:** Capacidad para el aprendizaje autónomo y profundo
- **CG14:** Capacidad de gestión de la información en la solución de situaciones problemáticas
- **CG17:** Capacidad para el razonamiento crítico

## 5. Actividades Formativas y Metodologías Docentes

### 5.1. Actividades formativas:

- Sesiones de Teoría sobre los contenidos del Programa.
- Sesiones de Resolución de Problemas.
- Sesiones Prácticas en Laboratorios Especializados o en Aulas de Informática.

### 5.2. Metodologías docentes:

- Clase Magistral Participativa.
- Desarrollo de Prácticas en Laboratorios Especializados o Aulas de Informática en grupos reducidos.
- Resolución de Problemas y Ejercicios Prácticos.
- Evaluaciones y Exámenes.

### 5.3. Desarrollo y justificación:

#### **Sesiones Académicas de Teoría (16%):**

- Consisten en clases magistrales donde se impartirá la base teórica de la asignatura y se expondrán ejemplos aclaratorios de la misma. Las sesiones académicas de teoría se irán intercalando con las sesiones académicas de problemas a lo largo del curso.
- La metodología usada para impartir la teoría y los ejemplos aclaratorios será la exposición mediante mesa tecnológica conjuntamente con pizarra. La profesora solicitará la participación activa del alumno mediante preguntas rápidas. Dicha participación podrá ser tenida en cuenta en la evaluación final.
- En la página web de la asignatura se encontrarán las transparencias y otros materiales de referencia necesarios para el seguimiento de las sesiones.

#### **Sesiones Académicas de Problemas (4%):**

- Consisten en la realización problemas relacionados con los conceptos y métodos operativos de la asignatura. La participación activa del alumnado podrá ser tenida en cuenta en la evaluación final.

#### **Sesiones Prácticas de Laboratorio (Sala de Informática) (20%):**

- Consisten en el diseño y utilización de pequeñas bases de datos y en el diseño e implementación de programas en C++, siendo esta última actividad la que conllevará un mayor número de sesiones y por lo tanto la que tendrá más peso en la nota de la parte práctica.
- A lo largo del cuatrimestre se irán resolviendo por parte del alumno y de manera individual, una serie de guiones prácticos (que estarán disponibles al comienzo del cuatrimestre) cuyo único objetivo será afianzar los conceptos impartidos en las sesiones teóricas, de modo que el alumno adquiera soltura en:
  - El diseño de una pequeña base de datos relacional, junto con la creación de formularios, consultas e informes.
  - El planteamiento de algoritmos que resuelvan un problema y la implementación de los mismos en el lenguaje C++.
- Habrá dos pruebas prácticas. La primera de ellas se realizará tras finalizar con el guión de bases de datos y la segunda (que estará relacionada con los guiones de C++) en la última sesión práctica del cuatrimestre.
- La asistencia a prácticas es obligatoria, excepto para los casos recogidos en el artículo 9 *Modalidades especiales de evaluación y/o aprendizaje* de la *Normativa de Evaluación para la Titulaciones de Grado de la Universidad de Huelva*, siempre que haya sido debidamente acreditada dicha situación por parte del alumno al comienzo del cuatrimestre, según dicha Normativa.

#### **Trabajo individual del alumno (60%)**

## 6. Temario desarrollado:

### **Bloque Temático I: INTRODUCCIÓN A LA INFORMÁTICA**

#### **TEMA 1. CONCEPTO DE ORDENADOR Y SISTEMAS OPERATIVOS**

- 1.1. Introducción.
- 1.2. La Máquina de Von Neumann.
- 1.3. Concepto de Ordenador.
  - 1.3.1. Componentes de un Ordenador.
  - 1.3.2. El Ordenador físicamente.
- 1.4. Introducción al concepto de Sistema Operativo.

### **Bloque Temático II: FUNDAMENTOS DE PROGRAMACIÓN**

#### **TEMA 2. CONCEPTOS BÁSICOS DE ALGORÍTMICA**

- 2.1. Definición de Algoritmo.
  - 2.1.1. Propiedades de los Algoritmos.
- 2.2. ¿Qué es un Programa?
  - 2.2.1. Cómo se construye un Programa.
- 2.3. Traducción de ideas a un lenguaje de programación concreto: El problema de la implementación.
- 2.4. Lenguajes de programación. Clasificación.

#### **TEMA 3. EL PROCESO DE COMPILACIÓN, DEL CÓDIGO FUENTE AL CÓDIGO MÁQUINA**

- 3.1. Fase de compilación o interpretación, y linkado (link, montado o enlace).
- 3.2. Fase de ejecución de un programa.

#### **TEMA 4. ELEMENTOS BÁSICOS DE PROGRAMACIÓN**

- 4.1. Definiciones. Palabras reservadas básicas del lenguaje C++.
- 4.2. Tipos, Variables y Constantes.
- 4.3. Operaciones básicas aritmético-lógicas. Operadores y expresiones.
  - 4.3.1. Operación de asignación.
  - 4.3.2. Operadores aritméticos.
  - 4.3.3. Operadores relacionales.
  - 4.3.4. Operadores lógicos.
  - 4.3.5. Expresiones.
- 4.4. Operaciones básicas de entrada-salida.

#### **TEMA 5. CONTROL DE FLUJO DEL PROGRAMA**

- 5.1. Sentencias
- 5.2. Sentencias compuestas (bloques)
- 5.3. Estructuras de selección
  - 5.3.1. if else
  - 5.3.2. switch
- 5.4. Estructuras de repetición
  - 5.4.1. while
  - 5.4.2. do while
  - 5.4.3. for

#### **TEMA 6. DISEÑO DESCENDENTE: SUBPROGRAMAS**

- 6.1. Niveles de abstracción.
- 6.2. Solución de problemas utilizando técnicas de Diseño Descendente.
- 6.3. Estilo en la creación de un programa.
- 6.4. Subprogramas.
- 6.5. Parámetros y Variables Locales. Variables Globales.
  - 6.5.1. Ámbito de las variables.
  - 6.5.2. Variables locales.
  - 6.5.3. Parámetros por valor y parámetros por referencia
  - 6.5.4. Variables Globales
- 6.6. Prototipos de Funciones
- 6.7. Sección de Includes: Ficheros cabecera
- 6.8. Compilación y enlazado (link)

#### **TEMA 7. ESTRUCTURAS DE DATOS COMPLEJAS**

- 7.1. Vectores o Arrays unidimensionales
- 7.2. Vectores multidimensionales
- 7.3. Inicialización de arrays
- 7.4. Cadenas o vectores de cadenas: Inicialización
- 7.5. Paso de cadenas y arrays a funciones
- 7.6. Estructuras
- 7.7. Arrays y estructuras
- 7.8. Paso de estructuras a funciones
- 7.9. Paso de elementos de estructuras a funciones

### **Bloque Temático III: INTRODUCCIÓN A LAS BASES DE DATOS**

#### **TEMA 8. INTRODUCCIÓN A LAS BASES DE DATOS**

- 8.1 Introducción a los conceptos básicos relacionados con la Bases de Datos.
- 8.2 Creación de Bases de datos. Tablas, Consultas y Formularios
- 8.3 Relaciones entre tablas.
- 8.4 Asistentes de Bases de Datos.

## 7. Bibliografía

### 7.1. Bibliografía básica:

- Deitel, Paul J., "Cómo programar en C++", 9ª ed. Pearson Educación, 2014
- Davis, Stephen R., "Beginning programming with C++ for dummies" [Recurso electrónico] Hoboken, NJ: Wiley 2010
- Lee, Mark, "C++ programming for the absolute beginner" [Recurso electrónico]. 2ª ed.
- Schildt, Herb., "C++ soluciones de programación" [Recurso electrónico]. McGraw-Hill Interamericana, 2009
- Joyanes Aguilar, Luis, "Programación en C++: Un enfoque práctico". [Recurso electrónico]. McGraw-Hill Interamericana de España, 2006
- Luján Mora, Sergio, "C++ paso a paso". [Recurso electrónico]. Publicaciones de la Universidad de Alicante,
- García Sánchez, José D., "Problemas resueltos de programación en lenguaje C++". Thomson, 2004
- Pons Capote, Olga, "Introducción a las sistemas de bases de datos". Paraninfo
- Sosa Flores, Miguel, "Diseño de Bases de Datos Relacionales". E-libro (Servicio en línea de la B.U.H.)
- Grau Fernández, Luis, "Problemas de bases de datos". ISBN 8496094693

### 7.2. Bibliografía complementaria:

- Prieto, A; Lloris, A; Torres, J.C. "Introducción a la Informática". Editorial McGraw-Hill.
- L. Joyanes. "Fundamentos de Programación. Algoritmos y Estructuras de Datos". McGraw-Hill
- Miguel Anasagasti, P. "Fundamentos de los Computadores", Paraninfo

## 8. Sistemas y criterios de evaluación.

### 8.1. Sistemas de evaluación:

- Examen de teoría/problemas
- Examen de prácticas

### 8.2. Criterios de evaluación y calificación:

## EXAMEN DE TEORÍA/PROBLEMAS

Consiste en una prueba en la que el alumno deberá resolver distintos problemas y/o cuestiones teóricas referentes al temario desarrollado durante el cuatrimestre. Se evaluarán las competencias B03, G01, G07, G09 y G14.

## EXAMEN PRÁCTICO EN EL AULA DE INFORMÁTICA

- **Evaluación continua** (para el alumnado no exento de asistir): Tras finalizar con el trabajo del guión de **bases de datos** se realizará una prueba práctica valorada en **0.5 puntos**. En la última sesión práctica del cuatrimestre y por lo tanto tras haber trabajado con los guiones de **C++**, se realizará una prueba práctica de programación en dicho lenguaje, valorada en **2.5 puntos**.
- **Examen práctico final** (para el alumnado exento de asistir a prácticas o que tenga más de 1 falta sin justificar): Se trata de una prueba práctica (con ejercicios de bases de datos y de programación en C++) que se realizará el mismo día de la convocatoria del examen teórico, a una hora que será comunicada con la suficiente antelación.
- La asistencia a prácticas es obligatoria, de modo que aquellos alumnos con 2 ó más faltas no justificadas no podrán ser evaluados de manera continua, teniendo estos que asistir al examen práctico final.
- La suma de las puntuaciones de ambas pruebas prácticas o en su caso el resultado del examen práctico final, será la nota del apartado práctico.
- Con estas pruebas/examen se evaluarán las competencias B03, G01, G03, G04, G06, G07, G08, G09, G12, G14 y G17

## CRITERIOS DE EVALUACIÓN Y CALIFICACIÓN

Se pueden ver resumidos en la siguiente tabla:

### (A) Conocimientos Teóricos:

- **Peso Total: 70%**
- Se realizará un examen teórico escrito.

### (B) Conocimientos Prácticos:

- **Peso Total: 30%**
- Se evaluará como se indica a continuación:
  - Convocatoria de Junio: Mediante evaluación continua o examen práctico final según corresponda al alumno.
  - Convocatoria de Septiembre: Examen práctico el mismo día que el examen teórico, a una hora que se avisará con tiempo suficiente.

Sea:

- **A** la nota, **sobre 10**, obtenida en el **examen teórico**.
- **B** la nota, **sobre 3**, obtenida en **prácticas**.

La **NOTA EN ACTA** del alumno se calculará del siguiente modo:

- Si  $A \geq 4$  entonces la nota en acta será:  $(A \times 0.7) + B$
- Si  $A < 4$  entonces la nota en acta será:  $A \times 0.7$

### Criterio para guardar las notas en convocatorias posteriores:

- La nota del **Examen Teórico** se guarda hasta la convocatoria de **Septiembre**, siempre que se obtenga una **calificación igual o superior a 4 sobre 10**.
- La nota de **Prácticas** se guarda hasta la convocatoria de **Septiembre**, independientemente de su calificación. El alumno tendrá la opción de poder renunciar a su nota de la convocatoria de Junio y presentarse al examen práctico de Septiembre, que se realizará el mismo día del examen teórico.

### 9. Organización docente semanal orientativa:

	Semanas	Grupos Grandes	Grupos Reducidos Aula Estándar	Grupos Reducidos Aula de Informática	Grupos Reducidos Laboratorio	Grupos Reducidos prácticas de campo	Pruebas y/o actividades evaluables	Contenido desarrollado
#1	3	0	2	0	0			Bloque Temático I
#2	3	0	2	0	0			Bloque Temático III
#3	3	0	2	0	0			Bloque Temático II
#4	3	0	2	0	0			Bloque Temático II
#5	3	0	2	0	0	Prueba práctica de BD		Bloque Temático II
#6	3	0	2	0	0			Bloque Temático II
#7	3	0	2	0	0			Bloque Temático II
#8	3	0	2	0	0			Bloque Temático II
#9	3	0	2	0	0			Bloque Temático II
#10	3	0	2	0	0			Bloque Temático II
#11	0	0	2	0	0			
#12	0	0	2	0	0			
#13	0	0	2	0	0			
#14	0	0	2	0	0			
#15	0	0	2	0	0	Prueba práctica de C++		
	30	0	30	0	0			