



Grado en Ingeniería Informática

DATOS DE LA ASIGNATURA

Nombre:

Fundamentos de Programación

Denominación en inglés:

Programming Fundamentals

Código:

606010104

Carácter:

Básico

Horas:**Totales****Presenciales****No presenciales****Trabajo estimado:**

150

60

90

Créditos:**Grupos reducidos****Grupos grandes****Aula estándar****Laboratorio****Prácticas de campo****Aula de informática**

3

0

3

0

0

Departamentos:**Áreas de Conocimiento:**

Tecnologías de la Información

Lenguajes y Sistemas Informáticos

Curso:**Cuatrimestre:**

1º - Primero

Primer cuatrimestre

DATOS DE LOS PROFESORES

Nombre:**E-Mail:****Teléfono:****Despacho:**

Ortíz Sañudo, Lourdes

lourdes@uhu.es

959217391

ETP133 / Escuela Técnica Superior de Ingeniería / Campus del Carmen

Pachón Álvarez, Victoria

vpachon@uhu.es

87373

119 Edificio de la Escuela Técnica Superior de Ingeniería

Polo Almohano, M ^a del Pilar	polo@dti.uhu.es	87386	ETP125 - Escuela Técnica Superior de Ingeniería-Campus de El Carmen
*Roche Beltrán, Francisco	roche@dti.uhu.es	959217654	ETP-126 Escuela Técnica Superior de Ingeniería
Domínguez Olmedo, Juan Luis	juan.dominguez@dti.uhu.es	959217371	ETP-123
Guerra Rodríguez, Juan Félix	felix.guerra@dti.uhu.es		144/ETSI/El Carmen

*Profesor coordinador de la asignatura

[Consultar los horarios de la asignatura](#)

1. Descripción de contenidos**1.1. Breve descripción (en castellano):**

Introducción a los Lenguajes de Programación

- Concepto e historia de los lenguajes de programación
- Paradigmas de Programación
- Compiladores e Intérpretes.

Introducción a la Programación Orientada a Objetos

Algoritmos y Tipos de Datos.

- Algoritmos. Tipos de Datos, Operadores y Expresiones.
- Estructuras de Control.
- Tipos Básicos de Datos Estructurados.
- Métodos e interfaces.

Técnicas de Diseño de Programas.

- Diseño Descendente.
- Diseño Modular.

1.2. Breve descripción (en inglés):

Introduction to Programming Languages

- Definition and history of programming languages
- Programming Paradigms
- Compilers and Interpreters.

Introduction to Object Oriented Programming

Algorithms and Data Types.

- Algorithms. Data Types, Operators and Expressions.
- Control Structures.
- Basic Types of Structured Data.
- Methods and interfaces.

Program Design Techniques.

- Design Descending.
- Modular Design.

2. Situación de la asignatura**2.1. Contexto dentro de la titulación:**

La asignatura de Fundamentos de la Programación es la primera asignatura en la que el alumno aprecia el papel central que representa la abstracción en la tarea de programar, a la vez que conoce un lenguaje de programación de alto nivel para codificar los programas, en el paradigma de la programación orientada a objetos. Es por tanto, imprescindible tanto en asignaturas específicas de programación como en aquellas en las que se hace uso de un lenguaje de programación:

- Estructuras de Datos I.
- Estructuras de Datos II.
- Metodología de la Programación.
- Ingeniería del Software.
- Sistemas Operativos.
- Programación Concurrente.

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.

Los alumnos que cursen esta asignatura deberían :

- Saber leer, comprender y expresarse correctamente tanto de forma oral como escrita.
- Tener conocimientos matemáticos a nivel de Bachillerato.
- Conocer el idioma inglés tanto a nivel escrito como oral (nivel medio).

3. Objetivos (Expresados como resultados del aprendizaje):

- Desarrollar una aproximación disciplinada a la especificación, implementación, verificación y documentación de programas.
- Aprender el papel central que representa la abstracción en la tarea de programar
- Desarrollar en el alumno la capacidad de resolución de problemas mediante las técnicas de diseño de algoritmos y aplicarlo a la codificación de programas
- Conocer y utilizar adecuadamente estructuras de datos básicas, algoritmos y esquemas de uso general
- Aprender un lenguaje de programación de alto nivel estructurado, general y extendido que use el paradigma de la programación orientado a objetos.
- Proporcionar los fundamentos teóricos y prácticos básicos para cursar posteriores estudios en programación.

4. Competencias a adquirir por los estudiantes

4.1. Competencias específicas:

- **CB04:** Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
- **CB05:** Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la 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
- **CB5:** Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía
- **CT2:** Desarrollo de una actitud crítica en relación con la capacidad de análisis y síntesis.
- **CT3:** Desarrollo de una actitud de indagación que permita la revisión y avance permanente del conocimiento.
- **CT4:** Capacidad de utilizar las Competencias Informáticas e Informacionales (CI2) en la práctica profesional.

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.
- Actividades Académicamente Dirigidas por el Profesorado: seminarios, conferencias, desarrollo de trabajos, debates, tutorías colectivas, actividades de evaluación y autoevaluación.

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 y de resolución de problemas

- Consisten en clases magistrales donde se impartirá la base teórica de la asignatura y se expondrán ejemplos aclaratorios de la misma al grupo. La sesión teórica tendrá una duración de 1h, dos veces por semana, sumando un total de 30h y se irán intercalando con las sesiones de problemas a lo largo del curso, de manera que una vez finalizado un tema teórico con sus correspondientes sesiones académicas de teoría, se impartirán sesiones de problemas.
- La metodología usada para impartir la teoría y los ejemplos aclaratorios será la exposición mediante cañón y/o uso de pizarra.

Sesiones prácticas en laboratorio

- Consisten en el diseño e implementación de programas escritos en C++ utilizando orientación a objetos desde el primer mes. Los alumnos dispondrán con antelación de una serie de cuadernillos de prácticas con un conjunto de problemas a resolver, con los que afianzarán los conocimientos adquiridos en las sesiones teóricas y que les servirán de entrenamiento para afrontar las diferentes pruebas prácticas con garantías de éxito.
- El trabajo se realizará de forma individual.
- Las sesiones prácticas tendrán una duración de 2h cada una, sumando un total de 30h.
- **Será obligatorio asistir como mínimo al 80% de las sesiones prácticas de laboratorio, para aquellos alumnos con evaluación continua.**

6. Temario desarrollado:

TEORIA:

Bloque I - Introducción a los Lenguajes de Programación.

TEMA 1. LENGUAJES DE PROGRAMACIÓN

- 1.1. Concepto e historia de los lenguajes de programación.
- 1.2. Paradigmas de Programación.
- 1.3. Clasificación. Lenguaje de bajo, medio y alto nivel.
- 1.4. Traductores: Compiladores e Intérpretes. Proceso de generación de código ejecutable. Compilación y Enlazado.
- 1.5. Programación Orientada a Objetos.

Bloque II - Algoritmos y Tipos de Datos.

TEMA 2. ALGORITMOS. Tipos de Datos, Operadores y Expresiones

- 2.1. Concepto de algoritmo. Estructura general.
- 2.2. Palabras clave, Identificadores, constantes y comentarios. Variables y objetos.
- 2.3. Tipos de datos y Clases. Clasificación.
- 2.4. Operaciones básicas de entrada-salida.
- 2.5. Operadores de asignación, aritméticos, relacionales y lógicos.
- 2.6. Expresiones y orden de precedencia.
- 2.7. Funciones genéricas y métodos.

TEMA 3. ESTRUCTURAS DE CONTROL

- 3.1. Sentencias y Bloques de sentencias.
- 3.2. Sentencias Secuenciales.
- 3.2. Sentencias Condicionales.
- 3.3. Sentencias Iterativas.
- 3.4. Macros de Sentencias.

TEMA 4. TIPOS DE DATOS ESTRUCTURADOS

- 4.1. Registros. Registros jerarquizados.
- 4.2. Tablas. Tablas de registros.
- 4.3. Esquemas de recorrido y búsqueda.
- 4.4. Cadenas.
- 4.5. Objetos y clases.

Bloque III - Técnicas de diseño de programas.

TEMA 5. DISEÑO DESCENDENTE

- 5.1. Introducción al diseño descendente y modular.
- 5.2. Programación Estructurada.
- 5.3. Variables globales y locales.
- 5.4. Variables miembro privadas y públicas.
- 5.5. Parámetros por valor y por referencia.
- 5.6. Constructores y destructores.
- 5.7. Paso de estructuras de datos a funciones y métodos.
- 5.8. Sobrecarga de funciones y métodos.

LABORATORIO:

Bloque II - Algoritmos y Tipos de Datos.

Prácticas 1, 2 y 3. Introducción a C++, uso de clases. Uso de cadenas y tablas.

Bloque III - Técnicas de diseño de programas.

Práctica 4. Diseño descendente.

7. Bibliografía

7.1. Bibliografía básica:

- METODOLOGÍA DE LA PROGRAMACIÓN I: INTRODUCCIÓN AL DISEÑO ORIENTADO A OBJETOS. A. Márquez, Lourdes Ortiz, M^a Pilar Polo, Fco. Roche y Ana M^a Roldán. Servicio de Publicaciones de la Universidad de Huelva.
- COMO PROGRAMAR EN C/C++. H.M. Deitel. Edt. PEARSON Prentice Hall.
- C++ ESTÁNDAR. E. Hernández Orallo. Edt. Paraninfo, Thomson Learning.
- PROGRAMACIÓN EN C++ PARA INGENIEROS. F. Xhafa, P. Vázquez, J. Marco, X. Molinero y A. Martín. Edt. Thomson.
- EL LENGUAJE DE PROGRAMACIÓN C++, B. Stroustrup. Ed. PEARSON Addison Wesley.
- PROGRAMACIÓN Y DISEÑO EN C++, J.P. Cohoon, J.W. Davidson. Edt. McGraw-Hill

7.2. Bibliografía complementaria:

- RESOLUCIÓN DE PROBLEMAS CON C++. W. Savitch. PEARSON Addison Wesley.

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:

Convocatoria I (o de curso)

La *evaluación* será *continua*, excepto para aquellos alumnos que soliciten acogerse a la *evaluación única final* siguiendo el procedimiento previsto en la presente guía docente.

Evaluación continua:

- **Teoría/Problemas.** Consistirá en una prueba escrita, el día fijado por la ETSI en el calendario de exámenes para la asignatura, en la que el alumno deberá resolver distintos problemas y/o cuestiones teóricas referentes a los temas de teoría desarrollados durante el cuatrimestre. Competencias B4 y B5. La puntuación máxima de este criterio es el **50% de la nota final en acta**.
- **Prácticas** (exámenes prácticos en aula de informática). A lo largo del cuatrimestre se realizarán 2 pruebas parciales sobre las prácticas de laboratorio. La **primera prueba práctica**, se realizará una vez finalizadas las sesiones de laboratorio establecidas para trabajar con los cuadernillos de prácticas de los temas 1, 2, 3 y 4, y en ella se evaluarán los conocimientos adquiridos hasta ese momento mediante la resolución de uno o varios ejercicios de complejidad similar a los propuestos en los cuadernillos. La **segunda prueba práctica** se llevará a cabo en la última sesión de laboratorio del cuatrimestre y estará relacionada con los ejercicios trabajados en el cuadernillo de prácticas del tema 5, consistiendo por tanto en la resolución de uno o varios ejercicios de una complejidad equivalente a la de los propuestos en el último cuadernillo. Competencias B4 y B5. La puntuación máxima de este criterio es del **50% de la nota final en acta**. La **primera prueba** tendrá una puntuación máxima de 1.5 puntos y la **segunda** de 3.5 puntos. Al exigirse una **asistencia mínima al 80%** de las sesiones prácticas, aquellos alumnos que no cumplan con este requisito serán calificados en esta convocatoria con un **No Presentado** en prácticas.

Evaluación única final:

- **Teoría/Problemas.** Consistirá en una prueba escrita, el día fijado por la ETSI en el calendario de exámenes para la asignatura, en la que el alumno deberá resolver distintos problemas y/o cuestiones teóricas referentes a los temas de teoría desarrollados durante el cuatrimestre. Competencias B4 y B5. La puntuación máxima de este criterio es el **50% de la nota final en acta**.
- **Prácticas.** Consistirá en un **único examen práctico** en un aula de informática, para resolver uno o varios ejercicios con los que se pueda evaluar si el alumno ha adquirido los conocimientos relacionados con los 5 temas de la asignatura. Dicho examen se llevará a cabo, el día fijado por la ETSI en el calendario de exámenes para la asignatura, y con la suficiente antelación el profesor coordinador concretará y hará públicas, en la plataforma de enseñanza virtual, la hora de comienzo de la prueba y las aulas destinadas a la misma. Competencias B4 y B5. La puntuación máxima de esta prueba es del **50% de la nota final en acta**. El estudiante que haya solicitado *evaluación única final*, **podrá ser evaluado de prácticas aun cuando no haya asistido a las mismas** durante el cuatrimestre.

Convocatoria II (o de recuperación de curso)

Evaluación continua y evaluación única final:

- **Teoría/Problemas.** Se realizará una prueba escrita (el día fijado por la ETSI en el calendario de exámenes para la asignatura) en la que el alumno deberá resolver distintos problemas y/o cuestiones teóricas referentes a la totalidad de temas desarrollados durante el curso. Competencias B4 y B5. La puntuación máxima de este criterio es el **50% de la nota final en acta**. La nota obtenida, en su caso en la convocatoria I se guardará para esta convocatoria, siempre que no se presente al examen en septiembre.
- **Prácticas** (examen práctico en aula de informática). Se realizará un **único examen práctico** en un aula de informática, para resolver uno o varios ejercicios con los que se pueda evaluar si el alumno ha adquirido los conocimientos

relacionados con la totalidad del temario de la asignatura. Dicho examen se llevará a cabo, el día fijado por la ETSI en el calendario de exámenes para la asignatura, y con la suficiente antelación el profesor coordinador concretará y hará públicas, en la plataforma de enseñanza virtual, la hora de comienzo de la prueba y las aulas destinadas a la misma. Competencias B4 y B5. La puntuación máxima de este criterio es del **50% de la nota final en acta**. La nota obtenida, en su caso en la convocatoria I se guardará para esta convocatoria, siempre que no se presente al examen en septiembre.

Convocatoria III (o de recuperación en curso posterior) y Convocatoria Extraordinaria para Finalización de Título

Evaluación única final:

- Se evaluará del total de conocimientos y competencias que figuren en la guía docente del curso anterior, de modo que se permita obtener el 100% de la calificación correspondiente de la asignatura.
- La evaluación única final para estas convocatorias, se realizará conforme al sistema de evaluación, para la *Convocatoria II*, aprobado y recogido en la guía docente del curso académico inmediatamente anterior a aquel en el que se efectúa dicha realización.

Regla para el cálculo de la Nota Final en Acta

La siguiente **regla será de aplicación en las tres convocatorias** y se basa en una nota máxima de 5 puntos por cada criterio de evaluación:

Nota_Final_Acta = Nota_Teoría/Problemas + Nota_Prácticas

Según el **Artículo 24 "Sistema de calificaciones"** del **Capítulo VI "Calificaciones"**, del *Reglamento de evaluación para las titulaciones de grado y máster oficial de la universidad de Huelva* (aprobado por Consejo de Gobierno de 13 de marzo de 2019), en su **Apartado 6** indica: "*En el caso de asignaturas cuyas guías docentes contemplen un examen final que suponga el 50% o más de la ponderación de la calificación final de la asignatura y el estudiante decidiera no realizarlo, figurará en el acta con la anotación de "No presentado" ...*".

Así pues:

Convocatoria I:

- **Evaluación continua:** Si un alumno **no se presenta al examen de teoría**, aparecerá en acta como **No presentado**.
- **Evaluación única final:** Si un alumno **no se presenta a la prueba teórica o a la práctica**, aparecerá en acta como **No presentado**.

Convocatoria II:

- Si un alumno **no se presenta a la prueba teórica o a la práctica**, aparecerá en acta como **No presentado**. Esto es así independientemente del tipo de evaluación (continua o única final).

Requisitos para la concesión de la mención "Matrícula de Honor" para discriminar situaciones de equidad

Cuando el número de alumnos que optan a ser evaluados con Matrícula de Honor, supera al número de posibilidades, se discriminará teniendo en cuenta los siguientes requisitos, en el orden indicado:

1. En primer lugar se ordenarán los alumnos en función de la nota final en actas.
2. Si continúa la equidad, se pasará a ordenarlos sólo por la nota de prácticas.
3. Si continúa la equidad se realizará un sorteo.

Procedimiento de solicitud de *evaluación única final* (para la Convocatoria I)

Según se recoge en el **Artículo 8 "Evaluación única final"** del *Reglamento de evaluación* (de 13 de marzo de 2019): "*Para acogerse a la evaluación única final, el estudiante, en las dos primeras semanas de impartición de la asignatura, o en las dos semanas siguientes a su matriculación, si se ha producido con posterioridad al inicio de la asignatura, lo comunicará de acuerdo con el procedimiento previsto en la guía docente de la asignatura... Esto implicará la renuncia expresa a la evaluación continua, sin posibilidad de que el estudiante pueda cambiar de sistema... No obstante, por causas excepcionales sobrevenidas y justificadas ... podrá solicitarse la evaluación única final fuera de los citados plazos, bajo el mismo procedimiento administrativo*".

Así pues, el alumno que quiera acogerse a la *evaluación única final*, deberá **rellenar una solicitud**, que **estará a su disposición en la plataforma de enseñanza virtual**, a la cual **adjuntará si fuese necesario la documentación acreditativa** de la situación que expone. **Dicha solicitud**, junto con la documentación (si se precisa) **deberá ser enviada por correo electrónico al profesor coordinador de la asignatura**. Para enviar dicho correo, el alumno deberá **utilizar obligatoriamente su cuenta de correo electrónico de la Universidad de Huelva**.

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	2	0	0	2	0		Presentación y Tema 1	
#2	2	0	0	2	0		Tema 2	
#3	2	0	0	2	0		Tema 2	
#4	2	0	0	2	0		Tema 3	
#5	2	0	0	2	0		Tema 3	
#6	2	0	0	2	0		Tema 4	
#7	2	0	0	2	0		Tema 4	
#8	2	0	0	2	0		Tema 4	
#9	2	0	0	2	0		Tema 5	
#10	2	0	0	2	0	Prueba Práctica 1	Tema 5	
#11	2	0	0	2	0		Tema 5	
#12	2	0	0	2	0		Tema 5	
#13	2	0	0	2	0		Tema 5	
#14	2	0	0	2	0		Tema 5	
#15	2	0	0	2	0	Prueba Práctica 2	Tema 5	
	30	0	0	30	0			