



Universidad
de Alcalá

GUÍA DOCENTE

Fundamentos de Programación

Grado en
Ingeniería Informática (GII)
Ingeniería en Sistemas de Información (GISI)
Ingeniería de Computadores (GIC)

Universidad de Alcalá

Curso Académico 2022/2023

1^{er} Curso - 1^{er} Cuatrimestre (GII+GISI+GIC)

GUÍA DOCENTE

Nombre de la asignatura:	Fundamentos de Programación
Código:	780003 (GII+GISI+GIC)
Titulación en la que se imparte:	Grado en Ingeniería Informática (GII) Ingeniería en Sistemas de Información (GISI) Ingeniería de Computadores (GIC)
Departamento y Área de Conocimiento:	Ciencias de la Computación Ciencias de la Computación
Carácter:	Básica (GII+GISI+GIC)
Créditos ECTS:	6.0
Curso y cuatrimestre:	1^{er} Curso - 1^{er} Cuatrimestre (GII+GISI+GIC)
Profesorado:	Salvador Sánchez Alonso
Horario de Tutoría:	Consultar al comienzo de la asignatura
Idioma en el que se imparte:	Español

1a. PRESENTACIÓN

La asignatura proporciona una introducción a las técnicas de resolución de problemas mediante el empleo de la programación, con énfasis en la teoría subyacente relacionada con el uso de lenguajes de programación de alto nivel. El objetivo es introducir a los estudiantes en el mundo de la programación, proporcionando los conocimientos básicos para comenzar a programar en cualquier lenguaje, sin asumir conocimientos previos. El curso se divide en 2 partes: la parte teórica, que se centra en los conceptos de diseño de software generales aplicables independientemente del lenguaje de programación, y la parte de laboratorio, que proporciona habilidades básicas en la programación de Python.

La asignatura constituye el primer contacto con la programación, por lo que no se suponen conocimientos previos sobre el tema.

1b. COURSE SUMMARY

This subject aims to introduce students in the world of programming, providing the core knowledge to begin programming in any language, with no assumed previous knowledge. The “Programming Fundamentals” course is divided in 2 parts, the laboratory part provides basic skills in Python programming, while the theoretical part focuses on software design with an emphasis on understanding the theory underlying the use of programming languages.

The subject teaches and illustrates the software design process, and shows how to develop a correct, readable and reusable solution from a problem specification.

2. COMPETENCIAS

Competencias básicas, generales y transversales.

Esta asignatura contribuye a adquirir las siguientes competencias básicas, generales y 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.

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.

CG5 - Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones

informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5, anexo 2, de la resolución BOE-A-2009-12977.

CG8 - Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.

CG9 - Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.

TRU1 - Capacidad de análisis y síntesis.

TRU2 - Comunicación oral y escrita.

TRU3 - Capacidad de gestión de la información.

TRU4 - Capacidad de aprendizaje autónomo.

TRU5 - Capacidad para trabajar en equipo.

Competencias Específicas

Esta asignatura proporciona la(s) siguiente(s) competencia(s) de carácter específico:

CIB3 - Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.

CIB4 - Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

CIB5 - 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.

Resultados de aprendizaje

Al terminar con éxito esta asignatura/enseñanza, los estudiantes serán capaces de:

RA1. Desarrollar la habilidad de crear soluciones algorítmicas a problemas y ser capaz de representarlas en forma de programas de computadora.

RA2. Aplicar la estrategia de implementación descendente y el diseño modular a la construcción de programas, siguiendo los principios de máxima cohesión y mínimo acoplamiento.

RA3. Distinguir entre diferentes técnicas de desarrollo, diseño, prueba y depuración aplicadas a problemas, adquiriendo en particular una visión sistémica de la verificación y validación.

RA4. Experimentar con un lenguaje y entorno de programación de alto nivel, identificando sus capacidades y limitaciones en comparación con otros lenguajes y entornos.

RA5. Explicar los conceptos básicos de almacenamiento y representación de datos, y deducir tanto la estructura de un dato como su compatibilidad con otros en función de su tipo.

RA6. Reconocer las tareas de comprobación que se llevan a cabo durante el procesamiento de un programa, y diseñar programas teniéndolas presentes.

3. CONTENIDOS

Bloques de contenido	Total de clases, créditos u horas *
Fundamentos de la Programación Introducción, sintaxis y semántica de los lenguajes de programación de alto nivel. Conceptos básicos: variables, tipos, expresiones y asignaciones. Entrada/Salida básica. Estructuras de control selectivas e iterativas. Modularización: funciones y paso de parámetros, descomposición modular de los programas.	<ul style="list-style-type: none"> • 16 horas
Metodología del desarrollo del Software y Algoritmos y resolución de problemas Estrategias de resolución de problemas: el papel de los algoritmos en el proceso de resolución de problemas, implementación de estrategias para algoritmos, pruebas y depuración del código, concepto de algoritmo y propiedades de los mismos.	<ul style="list-style-type: none"> • 14 horas
Estructuras de datos fundamentales Representación interna de los datos; tipos primitivos; tipos estructurados; definición y uso de nuevos tipos.	<ul style="list-style-type: none"> • 18 horas
Recursión Concepto, clasificación de la recursión, transformación de algoritmos recursivos, ejemplos clásicos.	<ul style="list-style-type: none"> • 8 horas

En total son 60 horas (56 horas de clase presencial + 4 horas de evaluación).

4. METODOLOGÍAS DE ENSEÑANZA APRENDIZAJE. ACTIVIDADES FORMATIVAS

4.1. Distribución de créditos (especificar en horas)

Número de horas presenciales:	28 horas de clase 28 horas de trabajo en laboratorio 4 horas de evaluación final (Total 60 horas)
Número de horas del trabajo propio del estudiante:	90 horas (Incluye horas de estudio, elaboración de actividades, preparación de exámenes)
Total horas	150 horas

4.2. Estrategias metodológicas, materiales y recursos didácticos

La asignatura se organiza como una asignatura cuatrimestral de 6 ECTS, empleándose en el proceso de enseñanza-aprendizaje de sus contenidos las siguientes actividades formativas:

- Clases Teóricas presenciales.
- Clases Prácticas presenciales.
- Prácticas en Laboratorio presenciales.
- Tutorías: individuales y/o grupales.

Además, en función de la naturaleza de las distintas partes de la materia objeto de estudio, se podrán utilizar, entre otras, las siguientes actividades formativas:

- Elaboración de trabajos individuales o en equipo.
- Puesta en común de la información, problemas y dudas que aparezcan en la realización de los trabajos.
- Utilización de la Plataforma de Aula Virtual.

<p>Actividades presenciales:</p>	<ul style="list-style-type: none"> • En el aula: Exposición y discusión de los conocimientos básicos de la asignatura. Planteamiento y resolución teórica de ejercicios y supuestos relacionados. Actividades (lecturas, discusiones, casos, etc.) orientadas a la enseñanza de las competencias específicas de la asignatura. • En el laboratorio: Planteamiento, desarrollo y solución de ejercicios prácticos utilizando herramientas, técnicas y métodos objeto de estudio de la asignatura, contribuyendo al desarrollo de la capacidad de análisis, razonamiento crítico y comprensión de las prácticas realizadas.
<p>Actividades no presenciales:</p>	<ul style="list-style-type: none"> • Análisis y asimilación de los contenidos de la materia, resolución de problemas, consultas bibliográficas, preparación de trabajos individuales y/o grupales, realización de autoevaluaciones, sesiones de revisión post-evaluación. Todas ellas orientadas al desarrollo de métodos para la organización y planificación del trabajo personal. • Tutorías: asesoramiento individual y/o en grupos durante el proceso de enseñanza-aprendizaje, bien en forma presencial o bien a distancia.

Recursos y materiales:

- Bibliografía de referencia.
- Entornos hardware/software de desarrollo.
- Plataforma de Aula Virtual y manuales de uso.
- Projectores.

5. EVALUACIÓN: Procedimientos, criterios de evaluación y calificación

Preferentemente se ofrecerá a los alumnos un sistema de evaluación continua que tenga características de evaluación formativa de manera que sirva de realimentación en el proceso de enseñanza-aprendizaje por parte del alumno.

5.1. PROCEDIMIENTOS

La evaluación debe estar inspirada en los criterios de evaluación continua (Normativa de Evaluación de los Aprendizajes, NEA, art 3). No obstante, respetando la normativa de la Universidad de Alcalá se pone a disposición del alumno un proceso alternativo de evaluación final de acuerdo a la Normativa de Evaluación de los Aprendizaje (aprobada en Consejo de Gobierno de 24 de marzo de 2011 y modificada en Consejo de Gobierno de 5 de mayo de 2016) según lo indicado en su Artículo 10, los alumnos tendrán un plazo de quince días desde el inicio del curso para solicitar por escrito al Director de la Escuela Politécnica Superior su intención de acogerse al modelo de evaluación no continua aduciendo

las razones que estimen convenientes. La evaluación del proceso de aprendizaje de todos los alumnos que no cursen solicitud al respecto o vean denegada la misma se realizará, por defecto, de acuerdo al modelo de evaluación continua. El estudiante dispone de dos convocatorias para superar la asignatura, una ordinaria y otra extraordinaria.

Convocatoria ordinaria

Evaluación continua:

En la convocatoria ordinaria el método de evaluación por defecto es la evaluación continua, con características de evaluación formativa para servir de realimentación en el proceso de enseñanza-aprendizaje por parte del alumno

Evaluación mediante examen final:

Opcionalmente, y de manera justificada, el alumno podrá solicitar ante el director del centro la evaluación mediante prueba única, lo que deberá de ser solicitado por escrito y en los plazos reglamentados. Dicho examen constará de una prueba única, escrita, donde se evaluarán todos los contenidos de la asignatura, tanto de la parte teórica como de la parte de laboratorio.

Convocatoria extraordinaria

El procedimiento será el mismo que el descrito para la evaluación mediante examen final en la convocatoria ordinaria.

5.2. EVALUACIÓN

CRITERIOS DE EVALUACIÓN

Se utilizarán los siguientes criterios para la evaluación de la asignatura, atendiendo al grado de adquisición de las competencias por parte del estudiante.

- CE1.** El alumno demuestra el dominio de los rudimentos básicos de la programación mediante la escritura de código simple, completo, robusto y eficiente.
- CE2.** El alumno demuestra aptitud en la resolución de problemas mediante programas de computadora.
- CE3.** El alumno distingue las diferentes estructuras de control de la lógica de un programa y es capaz de utilizar las más adecuadas en cada circunstancia.
- CE4.** A partir de la descripción de un problema complejo, el alumno es capaz de diseñar programas modulares cuya estructura viene determinada por los principios de cohesión y acoplamiento según las buenas prácticas de la programación descendente.
- CE5.** El alumno es capaz de diseñar soluciones iterativas y recursivas para un mismo problema y distingue las ventajas e inconvenientes de cada una.
- CE6.** El alumno es capaz de realizar un correcto análisis de las estructuras y tipos de datos necesario, así como de proponer alternativas de diseño en función de las necesidades específicas del problema.
- CE7.** El alumno entiende un código escrito por otra persona y es capaz de modificarlo para corregirlo o para ampliar su funcionalidad.

INSTRUMENTOS DE EVALUACIÓN

Esta sección resume los instrumentos de evaluación que serán aplicados a cada uno de los criterios de evaluación.

- **Prueba de Evaluación Intermedia (PEI-1):** Resolución de cuestiones teórico-prácticas sobre estructuras de control, rudimentos de la programación modular y conceptos básicos.
- **Prueba de Laboratorio (PL-1):** Elaboración de programas sencillos en un lenguaje de programación.
- **Prueba de Evaluación Intermedia (PEI-2):** Resolución de cuestiones teórico-prácticas sobre datos estructurados, recursividad, manejo de estructuras de datos y algoritmos de tratamiento de datos.
- **Prueba de Laboratorio (PL-2):** Elaboración de programas de complejidad intermedia en un lenguaje de programación.
- **Prueba de Evaluación Final (PEF):** Consistente en la resolución de supuestos teórico-prácticos, así como en la solución de problemas complejos mediante el uso de programas en un lenguaje de programación.

CRITERIOS DE CALIFICACIÓN

Esta sección cuantifica los criterios de evaluación para la superación de la asignatura.

Convocatoria Ordinaria - Evaluación Continua

En la convocatoria **ordinaria–evaluación continua** la relación entre las competencias, resultados del aprendizaje, criterios e instrumentos de evaluación, es la siguiente.

Competencia	Resultado de Aprendizaje	Criterio de Evaluación	Instrumento de evaluación	Peso en la calificación
CG8, CG9, CIB4, CB1-2, CB5, TRU1-4	RA1, RA2, RA5	C1, C3, C7	PEI-1	10%
CG8, CG9, CIB4, CB1-2, CB5, TRU1-5	RA1, RA2, RA4, RA6	C1, C2, C3, C7	PL-1	10%
CG5, CG8, CG9, CIB3-5, CB3-4, TRU1-4	RA1, RA2, RA3, RA4, RA5, RA6	C1, C2, C3, C5, C7	PEI-2	40%
CG5, CG8, CG9, CIB3-5, CB3-4, TRU1-5	RA1, RA2, RA3, RA4, RA6	C1, C2, C3, C4, C5, C6	PL-2	40%

Como criterio general, se otorgará la calificación de "No presentado" a aquellos alumnos que en convocatoria ordinaria se presenten a la evaluación de menos del 50% de todas las prácticas.

Convocatoria Ordinaria - Evaluación Final

En la convocatoria **ordinaria–evaluación final** la relación entre las competencias, resultados del aprendizaje, criterios e instrumentos de evaluación, es la siguiente.

Competencia	Resultado de Aprendizaje	Criterio de Evaluación	Instrumento de evaluación	Peso en la calificación
CIB3-5, CG5, CG8, CG9, CB1-5, TRU1-5	RA1, RA2, RA3, RA4, RA5, RA6	CE1, CE2, CE3, CE4, CE5, CE6, C7	PEF	100%

Convocatoria extraordinaria

En el caso de la convocatoria extraordinaria se mantendrán los mismos porcentajes que se han establecido en el caso de la evaluación mediante examen final.

Competencia	Resultado de Aprendizaje	Criterio de Evaluación	Instrumento de evaluación	Peso en la calificación
CIB3-5, CG5, CG8, CG9, CB1-5, TRU1-5	RA1, RA2, RA3, RA4, RA5, RA6	CE1, CE2, CE3, CE4, CE5, CE6, C7	PEF	100%

Todas las pruebas podrán realizarse en las aulas de teoría o laboratorio, o a través del Aula Virtual, siendo la superación de las prácticas y de las actividades de la asignatura requisito necesario para la superación de la asignatura.

Como resultado del proceso de evaluación el alumno obtendrá una calificación que dependerá de su actividad en las distintas pruebas de la asignatura. El resultado de cada prueba arrojará información bien mediante indicadores cuantitativos de adquisición de competencias, bien mediante una calificación cualitativa, que a modo de orientación podrá determinarse en función del grado de dominio mostrado en las tareas propuestas por los profesores responsables de la asignatura:

Sobresaliente	Notable	Aprobado	Suspense
Excelente dominio de los conocimientos básicos. Elaboración de ideas a partir de la reflexión y aplicación de los conocimientos adquiridos. Cumplimiento de todas las tareas programadas.	Dominio de los conocimientos básicos. Alto nivel de reflexión. Cumplimiento adecuado de la mayoría de las tareas programadas.	Domina los conocimientos básicos. Nivel medio de reflexión. Cumplimiento de un número suficiente de las tareas programadas.	Bajo nivel de comprensión y aplicación de ideas. Nivel bajo de reflexión. Falta de implicación en las tareas propuestas por el profesor.

6. BIBLIOGRAFÍA

6.1. Bibliografía básica

- MARZAL, A. y GRACIA, I. Introducción a la programación con Python - UJI. Ed. Publicacions de la Universitat Jaume I.
- PEÑA, R. Resolución de problemas para ingenieros con Python estructurado. Ed. Garceta.
- SEVERANCE, C. Python for informatics: <http://www.pythonlearn.com/book.php>

6.2. Bibliografía complementaria

a) Python

- Learn Python the hard way - <http://learnpythonthehardway.org/book/>
- A bite of Python - <http://www.swaroopch.com/notes/python/>
- Think Python - <http://www.greenteapress.com/thinkpython/>
- John M. Zelle (2010) Python Programming: An Introduction to Computer Science. Editorial Franklin, Beedle & Associates. 2ª ed.

b) Teoría de la programación

- GARCÍA MOLINA, F., MONTOYA DATO, J. y otros. Una Introducción a la Programación: Un enfoque algorítmico. Ed. Thomson Paraninfo. 2005.
- JOYANES, L. Fundamentos de la programación, 1ª Ed. Ed. McGraw-Hill. 1992.

NOTA INFORMATIVA

La Universidad de Alcalá garantiza a sus estudiantes que, si por exigencias sanitarias las autoridades competentes impidieran la presencialidad total o parcial de la actividad docente, los planes docentes alcanzarían sus objetivos a través de una metodología de enseñanza-aprendizaje y evaluación en formato online, que retornaría a la modalidad presencial en cuanto cesaran dichos impedimentos.