

**Proyecto/Guía docente de la asignatura**

Asignatura	FUNDAMENTOS DE PROGRAMACIÓN		
Materia	INFORMÁTICA		
Módulo	FORMACIÓN BÁSICA		
Titulación	GRADO EN INGENIERÍA BIOMÉDICA		
Plan	637	Código	47509
Periodo de impartición	2º Cuatrimestre	Tipo/Carácter	Formación Básica
Nivel/Ciclo	Grado	Curso	1º
Créditos ECTS	6		
Lengua en que se imparte	Castellano		
Profesor/es responsable/s	Félix Miguel Trespaderne Rogelio Mazaeda Echevarría		
Datos de contacto (E-mail, teléfono...)	Félix Miguel Trespaderne: trespa@eii.uva.es Rogelio Mazaeda Echevarría: rogelio@cta.uva.es Tutorías: Consultar la web de la UVa		
Departamento	Ingeniería de Sistemas y Automática		



1. Situación / Sentido de la Asignatura

1.1 Contextualización

La asignatura Fundamentos de Programación forma parte de las asignaturas del módulo básico del grado. Esta asignatura es la primera relacionada con la Informática de la titulación.

La procedencia de los estudiantes será fundamentalmente la de alumnos que han cursado el Bachillerato. Dependiendo de las opciones elegidas en los cursos anteriores, habrán tenido diferente grado de contacto con los contenidos desarrollados en esta materia. Así, existirán alumnos que no hayan tenido ninguna formación mientras que otros grupos de alumnos tendrán conocimientos básicos y dispersos según los centros donde hayan cursado asignaturas tales como Tecnologías de la Información y la Comunicación I y II.

Por ello, se pretende ofrecer una formación básica pero sólida, de tal forma que los conocimientos adquiridos sienten las bases para desenvolverse en el resto de materias afines de su formación académica. Se trata de habilitar a los estudiantes para que adquieran las destrezas necesarias para seguir aprendiendo los aspectos relacionados con la Informática, a nivel de manejo de ordenadores y especialmente en el ámbito de la programación.

1.2 Relación con otras materias

Conocimientos básicos de Cálculo, Álgebra y Física. Además, en el 5º cuatrimestre se impartirá Técnicas Computacionales en Biomedicina, la otra asignatura que compone la materia Informática. Otras materias donde los conocimientos adquiridos en la asignatura serán muy importantes son, entre otras, Bioinformática, Señales e Imágenes Médicas y Gestión de Información Biomédica y Sistemas Sanitarios.

1.3 Prerrequisitos

Es recomendable que los estudiantes conozcan los aspectos de lógica impartidos en bachillerato.



2. Competencias

2.1 Básicas

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.

CB5. Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.

2.2 Generales

CG1. Adquirir conocimientos y habilidades adecuados para analizar y sintetizar problemas básicos relacionados con la ingeniería y las ciencias biomédicas, resolverlos utilizando el método científico y comunicarlos de forma eficiente.

CG2. Conocer las bases científicas y técnicas de la ingeniería biomédica, de modo que se facilite el aprendizaje de nuevos métodos y tecnologías, así como el desarrollo de una gran versatilidad para adaptarse a nuevas situaciones.

CG3. Adquirir la capacidad de resolver problemas con iniciativa y creatividad, así como de comunicar y transmitir conocimientos, habilidades y destrezas, comprendiendo la responsabilidad ética, social y profesional de la actividad del ingeniero biomédico.

CG4. Trabajar de forma adecuada en un laboratorio, incluyendo los aspectos de seguridad, manipulación de materiales y eliminación de residuos.

CG5. Adquirir, analizar, interpretar y gestionar información.

2.3 Transversales

CT1. Desarrollar capacidades de comunicación interpersonal y aprender a trabajar en equipos multidisciplinares, multiculturales e internacionales.

CT2. Capacidad de organizar y planificar su trabajo tomando las decisiones correctas basadas en la información disponible, reuniendo e interpretando datos relevantes para emitir juicios dentro de su área de estudio.

CT3. Desarrollar capacidades de aprendizaje autónomo y de por vida.

2.4 Específicas

CE19. Conocer las herramientas informáticas para analizar, calcular, representar y gestionar información en Ingeniería Biomédica

CE23. Integrar conocimientos multidisciplinares asociados a la ingeniería, biología y medicina.



3. Objetivos

Conocer los fundamentos básicos de la informática y su interrelación con los lenguajes de programación.
Presentar los conceptos básicos de programación.
Familiarizarse con las herramientas y procedimientos de una metodología de programación.
Entender las técnicas de programación imperativa y orientadas a objetos y su utilización en problemas multidisciplinares asociados a la ingeniería, biología y medicina.



4. Contenidos y/o bloques temáticos

Bloque 1: Fundamentos de Informática

Carga de trabajo en créditos ECTS:

1,5

a. Contextualización y justificación

En este bloque se estudia con nivel introductorio aquellos fundamentos de informática complementarios e interrelacionados con los contenidos del Bloque 2 de Fundamentos de Programación. Debe tenerse en cuenta que no existe una asignatura básica que recoja aspectos como la arquitectura de ordenadores, sistemas operativos o bases de datos en la titulación.

b. Objetivos de aprendizaje

- Conocer las diversas familias de lenguajes de programación.
- Estudiar las diferencias entre lenguajes compilados y lenguajes interpretados.
- Entender los diferentes sistemas de representación de datos y su interrelación con los tipos usados en los lenguajes de programación.
- Presentar los componentes fundamentales de las arquitecturas Von Neumann y Harvard.
- Comprender el ciclo de instrucción y las interrupciones.
- Estudiar las funciones básicas de un sistema operativo.
- Entender las implicaciones que el uso de las sentencias de un lenguaje de alto nivel tiene sobre los niveles inferiores de la arquitectura de un computador.
- Presentar los conceptos básicos de las bases de datos.

c. Contenidos

- Introducción a la Informática
- Lenguajes de Programación. Paradigmas.
- Sistemas de representación.
- Arquitectura de Computadores.
- Sistemas Operativos.
- Bases de Datos.

d. Métodos docentes

Actividades presenciales:

- Clases de teoría: Lección magistral.
- Tutorías y evaluación: Resolución de ejercicios y problemas.

Actividades no presenciales:

- Trabajo individual: Estudio/trabajo personal.

e. Plan de trabajo

Para este bloque se estiman 15 horas presenciales distribuidas en 15 horas teóricas en el aula. El tiempo de dedicación no presencial del alumno medio es de unas 22.5 horas.

f. Evaluación

La evaluación de este bloque se efectuará en los exámenes oficiales de la asignatura con un peso del **20%** de la nota final.

g Material docente

- Transparencias
- Documentación interactiva (Cuadernos de Jupyter)
- Videos en forma de píldoras de conocimiento

g.1 Bibliografía básica

- R. E. Bryant, D. R. O'Hallaron. *Computer Systems. A Programmer's Perspective*.
Pearson 2015. ISBN 978-1292101767
- A. Tanenbaum, T. Austin. *Structured Computer Organization*.
Pearson 2012. ISBN 978-0273769248
- A. Silberschatz, H. Korth, S. Sudarshan. *Fundamentos de Bases de Datos*.
McGraw-Hill Education 2014. ISBN 978-8448190330

g.2 Bibliografía complementaria

- W. Stallings. *Computer Organization and Architecture*.
Pearson 2015. ISBN 978-0134101613
- J.L. Hennessy, D.A. Patterson. *Computer Architecture: A Quantitative Approach*.
Morgan Kaufmann 2017. ISBN 978-0128119051

h. Recursos necesarios

Aula con proyector multimedia y pizarra para sesiones de teoría.
Plataforma educativa para publicar material didáctico, guías de ejercicios, soluciones, tareas, etc.
Acceso al material bibliográfico recomendado.

i. Temporalización

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
Bloque 1: Fundamentos de Informática (1.5 ECTS)	Véanse horarios oficiales de la asignatura. http://www.med.uva.es/grado-en-ingenieria-biomedica/grado-en-ingenieria-biomedica-horarios/
La temporalización detallada de los contenidos se publicará en el Campus Virtual de la asignatura antes del inicio de la impartición de la asignatura.	

Bloque 2: Fundamentos de ProgramaciónCarga de trabajo en créditos ECTS: **4,5****a. Contextualización y justificación**

En este bloque temático se examinará en detalle el paradigma de programación dominante, el imperativo, y se proporcionará una introducción a la programación orientada a objetos.

Para ello, se estudiará el lenguaje de programación interpretado (Python). También se analizará el lenguaje C++ para establecer elementos comparativos entre diferentes enfoques de programación en las primeras semanas del bloque.

Python es un lenguaje versátil, intuitivo y gratuito, ideal para profesionales que necesitan un lenguaje para su uso a nivel científico, siendo un lenguaje de referencia utilizado cada vez más en biomedicina y ciencias de datos.

Por otro lado, frente a la mucho menos pronunciada curva de aprendizaje que ofrece Python y al considerable menor tiempo invertido en el desarrollo de aplicaciones, C++ es un lenguaje difícil de aprender, pero extremadamente poderoso y ampliamente utilizado en diferentes ámbitos, incluida la ingeniería biomédica. A lo largo del grado o posteriormente en su vida profesional, el alumno o egresado, según sus necesidades, podrá profundizar de forma autodidacta a partir de la base conseguida con Python no sólo en C++, sino en otros lenguajes utilizados en el ámbito biomédico como R, MATLAB, etc.

b. Objetivos de aprendizaje

Entender las técnicas de programación imperativa y orientadas a objetos y su utilización en problemas multidisciplinares asociados a la ingeniería, biología y medicina.

Familiarizarse con los entornos de desarrollo modernos de programación para implementar programas sencillos. Presentar los patrones básicos de programación que permiten desarrollar algoritmos sencillos de acuerdo a unas especificaciones dadas.

Conocer las estructuras y colecciones de datos más habituales y bibliotecas para su uso.

Conocer bibliotecas para la representación y visualización de datos.

Presentar técnicas para la lectura y el almacenamiento persistente de información.

Familiarizarse con herramientas para crear y compartir documentos que contienen código fuente, ecuaciones, visualizaciones y texto.

Entender la importancia de redactar programas siguiendo una regla de estilo y de generar una documentación asociada.

c. Contenidos

- Variables, operadores, expresiones y sentencias
- Control de flujo
- Funciones
- Estructuras de datos
- Ficheros.

d. Métodos docentes

Actividades presenciales:

- Problemas: Resolución de ejercicios y problemas. Aprendizaje basado en problemas.
- Prácticas de laboratorio: Estudio de casos.
- Tutorías y evaluación: Resolución de ejercicios y problemas.

Actividades no presenciales:

- Trabajo individual: Estudio/trabajo personal.
- Trabajo en grupo: Aprendizaje basado en problemas y aprendizaje cooperativo.

e. Plan de trabajo

Para este bloque se estiman 45 horas presenciales distribuidas en 15 horas prácticas de resolución de problemas en el aula y 30 horas de prácticas en laboratorio. El tiempo de dedicación no presencial del alumno medio es de unas 67.5 horas.

f. Evaluación

La evaluación teórica de este bloque se efectuará en los exámenes oficiales de la asignatura con un peso del **10%** de la nota final.

La evaluación práctica de este bloque se efectuará mediante **evaluación continua**:

Pruebas realizadas en el ordenador en el horario de prácticas de laboratorio a lo largo del periodo lectivo con un peso total del **70%** de la nota final. Alguna de las pruebas podrá corresponder a un trabajo práctico realizado en grupo con un peso no superior al **10%** de la nota final.

La programación temporal y peso relativo de cada prueba se anunciará en el Campus Virtual al inicio del curso, aunque podrá variar en función de las circunstancias, tales como festivos, huelgas, coincidencia con exámenes y pruebas de otras asignaturas, etc.

g Material docente

- Transparencias
- Documentación interactiva (Cuadernos de Jupyter)
- Videos en forma de píldoras de conocimiento
- Colección de programas resueltos ilustrando elementos clave de la programación

g.1 Bibliografía básica

Félix Miguel Trespaderne, Rogelio Mazaeda Echevarría, Fundamentos de Programación en Python, https://www2.eii.uva.es/fund_inf/python

J. V. Guttag. *Introduction to computation and programming using Python*. Cambridge, MA, USA: The MIT Press 2016. ISBN-13: 978-0262529624

A. Downey. *Think Python. How to Think Like a Computer Scientist*. Needham, MA, USA: Green Tea Press 2015. ISBN-13: 978-1491939369

A. Marzal, I. Gracia, P. García. *Introducción a la programación con Python 3*. Col·lecció Sapientia Universitat Jaume I 2014. ISBN-13: 978-8469711781.

g.2 Bibliografía complementaria

S. Chazallet. *Python 3. Los fundamentos del lenguaje*. Eni 2016. ISBN-13: 978-2409006142

J. M. Zelle. *Python Programming: An Introduction to Computer Science*. Franklin Beedle & Assoc. 2016. ISBN-13: 978-1590282755

B. Stroustrup. *Programming: Principles and Practice Using C++*. Addison-Wesley Educational Publishers Inc. 2014. ISBN-13: 978-0321992789.

<https://docs.python.org/3/>

<https://www.python-course.eu/>

<https://python.swaroopch.com/>

<http://www.greenteapress.com/thinkpython/thinkpython.html>

<http://www.learncpp.com/cpp-tutorial/>

<https://en.cppreference.com/>

h. Recursos necesarios

Aula con proyector multimedia y pizarra para sesiones de teoría y de laboratorio con ordenadores para los alumnos.

Plataforma educativa para publicar material didáctico, guías de ejercicios, soluciones, tareas, etc.

Acceso al material bibliográfico recomendado.

Distribución Anaconda, con intérprete Python 3, entorno web Jupyter Notebooks y entorno de desarrollo Spyder.

Compilador de C/C++ (gcc) y entorno de desarrollo Code::Blocks.

i. Temporalización

CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
Bloque 2: Fundamentos de Programación (4.5 ECTS)	Véanse horarios oficiales de la asignatura. 1/3 en aula y 2/3 en laboratorio. http://www.med.uva.es/grado-en-ingenieria-biomedica/grado-en-ingenieria-biomedica-horarios/
La temporalización detallada de los contenidos se publicará en el Campus Virtual de la asignatura antes del inicio de la impartición de la asignatura.	

5. Métodos docentes y principios metodológicos

Actividades presenciales:

- Clases de teoría: Lección magistral.
- Prácticas de laboratorio: Estudio de casos.
- Problemas, tutorías y evaluación: Resolución de ejercicios y problemas, aprendizaje basado en problemas y aprendizaje cooperativo.

Actividades no presenciales:

- Trabajo individual: Estudio/trabajo personal.
- Trabajo en grupo: Aprendizaje basado en problemas y aprendizaje cooperativo.

6. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Teoría en aula	15	Trabajo individual	60
Prácticas en aula	15	Trabajo en grupo	30
Prácticas en laboratorio	30	--	--
Total presencial	60	Total no presencial	90

7. Sistema y características de la evaluación

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Evaluación continua basada en pruebas parciales	70%	Realizadas en el horario de Laboratorio
Evaluación final	30%	Período de exámenes

CRITERIOS DE CALIFICACIÓN

- **Convocatoria ordinaria:**
 - Examen sobre toda la materia, con un peso del **30%**: nota mínima del **12%**
 - El **70%** de la nota procede de la evaluación continua: nota mínima del **28%**
 - La suma de las notas debe igualar o superar el **50%**.
- **Convocatoria extraordinaria:**
 - Examen sobre toda la materia, con un peso del **30%**: nota mínima del **12%**
 - Examen de recuperación de la evaluación continua en aula de ordenadores, con un peso del **70%**: nota mínima del **28%**
 - La suma de las notas debe igualar o superar el **50%**.
 - El alumno, **con anterioridad** a la realización de las pruebas, podrá elegir conservar alguna de las dos notas (prueba escrita y/o evaluación continua), obtenidas durante la evaluación ordinaria.

8. Consideraciones finales

La planificación académica podrá sufrir modificaciones de acuerdo con la actualización de las condiciones sanitarias.

En el Campus Virtual de la asignatura se dispondrá de:

- Presentación de la asignatura
- Programa de la asignatura
- Planificación de la asignatura
- Material didáctico
- Enlaces a bibliografía online
- Cronograma de las diferentes actividades de la asignatura.

Con anterioridad a cada práctica de laboratorio, se subirán los guiones de las mismas.