

## 1. DATOS BÁSICOS

<b>Asignatura</b>	Computación Científica I
<b>Titulación</b>	Grado en Física
<b>Escuela/ Facultad</b>	Arquitectura, Ingeniería y Diseño
<b>Curso</b>	2025-2026
<b>ECTS</b>	6 ECTS
<b>Carácter</b>	BA
<b>Idioma/s</b>	Español
<b>Modalidad</b>	Presencial
<b>Semestre</b>	S1
<b>Curso académico</b>	1º
<b>Docente coordinador</b>	M Cruz Gaya López (Coordinadora Asignatura)
<b>Docente</b>	M Cruz Gaya López; Miguel Ordoñez

## 2. PRESENTACIÓN

En esta asignatura, se proporcionan fundamentos de programación de computadoras, pero orientados a sus aplicaciones al ámbito científico. Se busca que el estudiante se familiarice con los paradigmas clásicos de la programación y los aplique al diseño y desarrollo de programas de poca extensión y carácter aplicado. Como lenguaje de programación se ha elegido Python 3 por ser uno de los más utilizados actualmente y porque proporciona un balance equilibrado entre sencillez de uso y potencia. Debemos destacar que el énfasis de la asignatura está en el pensamiento algorítmico y en las estructuras de datos básicas y no tanto en los detalles específicos del lenguaje de programación elegido. Al mismo tiempo, se hace hincapié en la pulcritud en el estilo de programación y sus buenas prácticas, para que en lo sucesivo los estudiantes puedan generar código de manera limpia y estructurada.

Como la asignatura se desarrolla en el primer semestre del primer curso de la titulación, no se asume ningún conocimiento previo de programación. La asignatura tiene su continuación natural en Computación Científica II, prevista para el segundo semestre del primer curso. En conjunto, estas dos asignaturas proporcionan una base en Física Computacional. Los estudiantes interesados tendrán la posibilidad de ampliar sus conocimientos en esta rama en asignaturas optativas de cursos superiores.

## 3. COMPETENCIAS Y RESULTADOS DE APRENDIZAJE

A continuación, se muestran los resultados de aprendizaje que se desarrollan en el título divididos en conocimientos, habilidades y competencias (CONXX, HABXX, COMPXX). Junto a cada uno se listan los resultados de aprendizaje de materia que contribuyen a su consecución.

### Conocimientos

CON10. Adquirir una base sólida de programación en lenguajes ampliamente utilizados en el ámbito científico.

- Comprender la estructura física y lógica de un ordenador.
- Conocer los principios de los paradigmas de programación procedural, orientada a objetos y genérica.
- Describir el proceso de traducción de programas.

### Habilidades

HAB06. Resolver problemas aplicados físicos y/o matemáticos a través de la codificación de programas sencillos.

- Adquirir una base sólida de programación en lenguajes ampliamente utilizados en el ámbito científico, siendo capaz de resolver problemas aplicados físicos y/o matemáticos a través de la codificación de programas sencillos.
- Adquirir pensamiento algorítmico.

### Competencias

- COMP01. Capacidad de planificación y de realización de trabajo autónomo en la gestión de proyectos relacionados con las diferentes áreas de la Física.
- COMP07. Comprender y saber usar los métodos matemáticos y numéricos utilizados en Física y en el manejo de los datos experimentales.
- COMP08. Ser capaz de trabajar con un alto grado de autonomía, aceptando responsabilidades en la planificación y gestión de tecnologías o proyectos relacionados con las diferentes áreas de la física.

## 4. CONTENIDOS

La materia está organizada en siete temas:

- Tema 1 Introducción a la informática

Objetivos: Introducir el vocabulario básico relacionado con la informática que le permite comprender el contenido de la asignatura. Estructura de un computador (hardware y software).

Traducción de programas.

- Tema 2. Algoritmos. Estructura de un programa en Python
- Tema 3. Control de flujo
- Tema 4. Listas.
- Tema 5. Funciones
- Tema 6. Tuplas, diccionarios y excepciones.
- Tema 7. Ficheros

Cada uno de estos temas comparten los siguientes objetivos añadiendo los elementos de programación nuevos indicados en el nombre del tema. Los objetivos para todos ellos son:

- Desarrollar pensamiento algorítmico trasladando un problema a la secuencia de pasos que lo solucionan.
- Utilizar el lenguaje de programación Python para crear programas que solucionen problemas de complejidad baja y media.
- Utilizar un entorno de programación para la generación de software.

## 5. METODOLOGÍAS DE ENSEÑANZA-APRENDIZAJE

A continuación, se indican los tipos de metodologías de enseñanza-aprendizaje que se aplicarán:

- **Aprendizaje cooperativo:** los alumnos aprenden a colaborar con otras personas (compañeros y profesores) para resolver de forma creativa, integradora y constructiva los interrogantes y problemas identificados a partir de los casos planteados, utilizando los conocimientos y los recursos materiales disponibles.
- **Aprendizaje basado en problemas:** Se plantearán problemas con el objetivo de que los alumnos los solucionen trabajando en equipo o individualmente.
- **Clase magistral:** exposiciones realizadas por el profesor con las herramientas tecnológicas necesarias para la máxima comprensión de los conceptos impartidos.
- **Aprendizaje basado en enseñanzas de taller:** los alumnos adquieren los conocimientos mediante el dominio de la instrumentación que necesitarán en su profesión. Implica "aprender haciendo",
- **Actividades académicas dirigidas:** trabajos más autónomos, individuales y grupales, con búsqueda de información, síntesis escrita y debates y defensa pública de trabajos.

## 6. ACTIVIDADES FORMATIVAS

A continuación, se identifican los tipos de actividades formativas que se realizarán y la dedicación en horas del estudiante a cada una de ellas:

**Modalidad presencial:**

Actividad formativa	Número de horas
Lecciones magistrales	28
Clases magistrales asíncronas	24
Exposiciones orales de trabajos y debates	4
Elaboración de informes	27
Evaluación	12
Actividades prácticas (problemas, trabajos, proyectos, talleres y/o laboratorios)	76
<b>TOTAL</b>	<b>150</b>

## 7. EVALUACIÓN

A continuación, se relacionan los sistemas de evaluación, así como su peso sobre la calificación total de la asignatura:

**Modalidad presencial:**

SISTEMA DE EVALUACIÓN	PESO
Pruebas presenciales de conocimiento individuales, de carácter teórico y/o práctico	50.0
Defensa Oral	5.0
Entrega de Informes/ trabajos/ proyectos ejercicios grupales y/o individuales	35.0
Observación del desempeño	10.0

En el Campus Virtual, cuando accedas a la asignatura, podrás consultar en detalle las actividades de evaluación que debes realizar, así como las fechas de entrega y los procedimientos de evaluación de cada una de ellas.

### 7.1. Convocatoria ordinaria

Para superar la asignatura en convocatoria ordinaria deberás obtener una calificación mayor o igual que 5,0 sobre 10,0 en la calificación final (media ponderada) de la asignatura.

En todo caso, será necesario que obtengas una calificación mayor o igual que 5,0 en las pruebas de conocimiento, para que la misma pueda hacer media con el resto de actividades.

### 7.2. Convocatoria extraordinaria

Para superar la asignatura en convocatoria ordinaria deberás obtener una calificación mayor o igual que 5,0 sobre 10,0 en la calificación final (media ponderada) de la asignatura.

En todo caso, será necesario que obtengas una calificación mayor o igual que 5,0 en la prueba de conocimiento, para que la misma pueda hacer media con el resto de actividades.

Se deben entregar las actividades no superadas en convocatoria ordinaria, tras haber recibido las correcciones correspondientes a las mismas por parte del docente, o bien aquellas que no fueron entregadas.

## 8. CRONOGRAMA

En este apartado se indica el cronograma con fechas de entrega de actividades evaluables de la asignatura:

Actividades evaluables	Fecha
Ejercicios individuales	Semanas 3-14
Examen	Semana 6 y Semana 13
Proyecto	Semana 14-16
Evaluación final	Semana 17

Este cronograma podrá sufrir modificaciones por razones logísticas de las actividades. Cualquier modificación será notificada al estudiante en tiempo y forma.

## 9. BIBLIOGRAFÍA

Existen numerosos manuales de introducción a la programación en Python que son adecuados para este curso. De entre ellos, se lista uno de carácter básico, pero con numerosos ejemplos detallados (Gaddis 2019), y otro algo más avanzado y aplicado al análisis de datos (Gutttag 2016).

- Tony Gaddis (2021). Starting Out with Python, 5th edition. Editorial Pearson Education. Nueva York.

- John V. Gutttag (2016). Introduction to Computation and Programming Using Python: With Application to Understanding Data, 2nd edition. Editorial The MIT Press. Cambridge, MA. Otra

Además de los manuales especializados, hay gran cantidad de recursos en línea de gran calidad y utilidad para el curso. Se sugieren siguientes:

- Documentación oficial del lenguaje Python, en la página web de la Python Software Foundation [US]: <https://www.python.org/doc/>

- Documentación de “Fundamentos de Python 1”, desarrollado por OpenEDG Python Institute en asociación con Cisco Networking Academy. <https://www.netacad.com/>

- Online Python Tutor (Philip Guo), herramienta gratuita en línea que permite ejecutar línea a línea códigos de Python 3: <http://www.pythontutor.com/>

- Python Course (Bernd Klein), curso gratuito en línea y sin publicidad de gran calidad y actualizado (en inglés y alemán): <https://www.python-course.eu/>

## 10. UNIDAD DE ORIENTACIÓN EDUCATIVA Y DIVERSIDAD

Desde la Unidad de Orientación Educativa y Diversidad (ODI) ofrecemos acompañamiento a nuestros estudiantes a lo largo de su vida universitaria para ayudarles a alcanzar sus logros académicos. Otros de los pilares de nuestra actuación son la inclusión del estudiante con necesidades específicas de apoyo educativo, la accesibilidad universal en los distintos campus de la universidad y la equiparación de oportunidades.

Desde esta Unidad se ofrece a los estudiantes:

1. Acompañamiento y seguimiento mediante la realización de asesorías y planes personalizados a estudiantes que necesitan mejorar su rendimiento académico.
2. En materia de atención a la diversidad, se realizan ajustes curriculares no significativos, es decir, a nivel de metodología y evaluación, en aquellos alumnos con necesidades específicas de apoyo educativo persiguiendo con ello una equidad de oportunidades para todos los estudiantes.
3. Ofrecemos a los estudiantes diferentes recursos formativos extracurriculares para desarrollar diversas competencias que les enriquecerán en su desarrollo personal y profesional.
4. Orientación vocacional mediante la dotación de herramientas y asesorías a estudiantes con dudas vocacionales o que creen que se han equivocado en la elección de la titulación.

Los estudiantes que necesiten apoyo educativo pueden escribirnos a:  
[orientacioneducativa@universidadeuropea.es](mailto:orientacioneducativa@universidadeuropea.es)

## **11. ENCUESTAS DE SATISFACCIÓN**

¡Tu opinión importa!

La Universidad Europea te anima a participar en las encuestas de satisfacción para detectar puntos fuertes y áreas de mejora sobre el profesorado, la titulación y el proceso de enseñanza-aprendizaje.

Las encuestas estarán disponibles en el espacio de encuestas de tu campus virtual o a través de tu correo electrónico.

Tu valoración es necesaria para mejorar la calidad de la titulación.

Muchas gracias por tu participación.