



## Guía docente

# 820090 - PRE - Programación para Ingenieros

Última modificación: 29/01/2026

**Unidad responsable:** Escuela de Ingeniería de Barcelona Este

**Unidad que imparte:** 723 - CS - Departamento de Ciencias de la Computación.

**Titulación:** GRADO EN INGENIERÍA BIOMÉDICA (Plan 2009). (Asignatura optativa).  
GRADO EN INGENIERÍA DE LA ENERGÍA (Plan 2009). (Asignatura optativa).  
GRADO EN INGENIERÍA ELÉCTRICA (Plan 2009). (Asignatura optativa).  
GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA (Plan 2009). (Asignatura optativa).  
GRADO EN INGENIERÍA MECÁNICA (Plan 2009). (Asignatura optativa).  
GRADO EN INGENIERÍA QUÍMICA (Plan 2009). (Asignatura optativa).  
GRADO EN INGENIERÍA DE MATERIALES (Plan 2010). (Asignatura optativa).

**Curso:** 2025

**Créditos ECTS:** 6.0

**Idiomas:** Catalán, Castellano, Inglés

## PROFESORADO

**Profesorado responsable:**

**Otros:**

## CAPACIDADES PREVIAS

Este es un segundo curso de programación. Preferiblemente es deseable que los alumnos ya tengan adquiridas las capacidades básicas de programación (hechas en Informática de Q1).

## COMPETENCIAS DE LA TITULACIÓN A LAS QUE CONTRIBUYE LA ASIGNATURA

### Específicas:

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

### Transversales:

2. USO SOLVENTE DE LOS RECURSOS DE INFORMACIÓN - Nivel 3: Planificar y utilizar la información necesaria para un trabajo académico (por ejemplo, para el trabajo de fin de grado) a partir de una reflexión crítica sobre los recursos de información utilizados.

## METODOLOGÍAS DOCENTES

Semanalmente habrá una sesión teórica y una sesión práctica en laboratorio con ordenadores. Paralelamente el alumno desarrollará un proyecto cuyo enunciado él mismo propondrá. Algunas sesiones de laboratorio durante el curso se dedicarán a la propuesta del alumno. Aprendizaje colaborativo. Aprendizaje basado en proyectos (PBL).

## OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

Objetivos de aprendizaje:

1. Introducir al alumno en la Programación Orientada a Objetos
2. Introducir al alumno en la Programación Orientada a Eventos y en el desarrollo de aplicaciones con interficie gráfica (GUI)



## HORAS TOTALES DE DEDICACIÓN DEL ESTUDIANTADO

Tipo	Horas	Porcentaje
Horas grupo pequeño	30,0	20.00
Horas grupo grande	30,0	20.00
Horas aprendizaje autónomo	90,0	60.00

**Dedicación total:** 150 h

## CONTENIDOS

### Introducción

**Descripción:**

Se explican las tareas a realizar en la asignatura y el método de evaluación.

**Objetivos específicos:**

Recordatorio sobre la programación con acciones.

**Actividades vinculadas:**

Propuesta de proyecto por parte del alumno

**Dedicación:** 10h

Grupo grande/Teoría: 2h

Grupo pequeño/Laboratorio: 4h

Aprendizaje autónomo: 4h

### Análisis y ciclo de vida

**Descripción:**

En este tema se da a los alumnos una visión general del proceso de desarrollo de una aplicación informática.

**Objetivos específicos:**

- Describir el ciclo de vida de una aplicación informática, y por extensión, de un proyecto de desarrollo informático
- Describir cómo realizar un análisis de requerimientos

**Actividades vinculadas:**

Ánalysis de requerimientos de la propuesta.

**Dedicación:** 6h

Clases teóricas: 2h

Aprendizaje autónomo: 4h



## Diseño de aplicaciones

### Descripción:

Se explican y se ponen en práctica diversas metodologías de diseño de aplicaciones.

### Objetivos específicos:

Diseño orientado a objetos

Diseño de patrones

Diseño visual

### Actividades vinculadas:

Diseño orientado a objetos de la propuesta

Diseño de patrones para la propuesta

Diseño visual de la propuesta

### Dedicación: 42h

Grupo grande/Teoría: 16h

Grupo pequeño/Laboratorio: 2h

Aprendizaje autónomo: 24h

## Programación en entorno gráfico

### Descripción:

Se explican las nociones necesarias para poder utilizar elementos visuales de programación y se desarrolla un proyecto.

Patrones de diseño Model View Presenter

### Dedicación: 20h

Grupo pequeño/Laboratorio: 10h

Aprendizaje autónomo: 10h

## Ampliación de programación

### Descripción:

Módulos en Python

Ficheros

Errores i excepciones

Depuración i traza de código

Gestión de versiones con Git y Github

### Dedicación: 8h

Grupo grande/Teoría: 4h

Grupo pequeño/Laboratorio: 4h

## Desarrollo

### Descripción:

Desarrollo de la propuesta por parte del alumno hasta llegar a obtener un programa completo y ejecutable

### Dedicación: 64h

Grupo grande/Teoría: 10h

Grupo pequeño/Laboratorio: 10h

Aprendizaje autónomo: 44h



## SISTEMA DE CALIFICACIÓN

---

La asignatura se evalúa en sucesivas entregas de un proyecto que el alumno desarrolla a lo largo del curso. Las distintas entregas y sus pesos son:

Diseño visual de la Aplicación 15%  
Diseño Orientado a Objetos 20%  
Propuesta de patrones 5%  
Trabajos Semanales 20%  
Proyecto Final 40%

## NORMAS PARA LA REALIZACIÓN DE LAS PRUEBAS.

---

No hay exámen final.

## BIBLIOGRAFÍA

---

### Básica:

- Yourdon, Edward. Techniques of program structure and design. Englewood Cliffs, NJ: Prentice-Hall, 1975. ISBN 013901702X.
- Parnas, D. L. "On the criteria to be used in decomposing systems in modules". Communicatios of the ACM [en línea]. [Consulta: 22/04/2020]. Disponible a: <https://doi-org.recursos.biblioteca.upc.edu/10.1145/361598.361623>.- Summerfield, Mark. Rapid GUI programming with Python and Qt : the definitive guide to PyQt programming. Upper Saddle River, NJ: Prentice Hall, 2007. ISBN 9780132354189.
- Matthes, Eric. Python Crash Course : a hands-on, project-based introduction to programming [en línea]. San Francisco: No Starch Press, 2016 [Consulta: 09/06/2020]. Disponible a: <https://ebookcentral.proquest.com/lib/upcatalunya-ebooks/detail.action?docID=4503145>. ISBN 9781593277390.