



## FICHA IDENTIFICATIVA

### DATOS DE LA ASIGNATURA

**Código:** 34686

**Nombre:** Programación avanzada

**Ciclo:** Grado

**Créditos ECTS:** 6

**Curso académico:** 2025-26

### TITULACIONES

Titulación	Centro	Curso	Periodo
1400 - Grado en Ingeniería Informática	Escola Tècnica Superior d'Enginyeria	4	Primer cuatrimestre
1407 - Grado en Ingeniería Multimedia	Escola Tècnica Superior d'Enginyeria	4	Primer cuatrimestre

### MATERIAS

Titulación	Materia	Carácter
1400 - Grado en Ingeniería Informática	Materia Optativa	OPTATIVA
1407 - Grado en Ingeniería Multimedia	Optatividad	OPTATIVA

### COORDINACIÓN

BARBER MIRALLES FERNANDO

## RESUMEN

La asignatura "Programación Avanzada" es una asignatura optativa de cuarto curso del Grado de Ingeniería Informática.

En esta asignatura se muestran técnicas de programación especialmente adecuadas para mejorar la eficiencia de los programas, incluyendo entre otras los templates y la metaprogramación.

La asignatura estará sobre todo centrada en C++. Se estudiarán los últimos estándares de C++ (C++11, C++14, ...) y C++ será el lenguaje de programación para los trabajos de la asignatura.

Estas tecnologías son especialmente interesantes en la programación de videojuegos y otros sistemas, en donde la eficiencia es imprescindible, así como para todo aquel que quiera sacar el máximo provecho del lenguaje C++.

## CONOCIMIENTOS PREVIOS



## RELACIÓN CON OTRAS ASIGNATURAS DE LA MISMA TITULACIÓN

No se han especificado restricciones de matrícula con otras asignaturas del plan de estudios.

## OTROS TIPOS DE REQUISITOS

Es muy conveniente que los alumnos hayan cursado las asignaturas Informática, Programación y Estructuras de Datos y Algoritmos. Es imprescindible tener un nivel medio de programación en C++. También es recomendable haber cursado Sistemas operativos y tener conocimientos básicos de la arquitectura de los procesadores.

## COMPETENCIAS / RESULTADOS DE APRENDIZAJE

-

C2 - Capacidad para adquirir, obtener, formalizar y representar el conocimiento humano en una forma computable para la resolución de problemas mediante un sistema informático en cualquier ámbito de aplicación, particularmente los relacionados con aspectos de computación, percepción y actuación en ambientes o entornos inteligentes.

Programar de forma correcta en los diferentes lenguajes específicos de los sistemas multimedia teniendo en cuenta las restricciones de tiempo y coste.

## DESCRIPCIÓN DE CONTENIDOS

### 1. Introducción a la programación eficiente

- Estrategias generales. Profiling.
- Ordenación sin comparaciones. Countsort. Radixsort.
- Coste amortizado de algoritmos. Vector expansible.

### 2. El estándar C++11

- Modificaciones al núcleo del lenguaje.
- Modificaciones a la librería estándar.

### 3. Mejora de la eficiencia en C++

- Optimizaciones comunes en C++.
- Memoria dinámica en C++. Modificación de new y delete.



#### 4. Uso de templates

- Funciones template.
- Clases template.
- Templates Variadic.

#### 5. Uso avanzado de templates

- Polimorfismo y templates.
- Traits templates y Policy classes.
- Metaprogramación.
- Templates de expresiones: Arrays numéricos eficientes.
- Arrays numéricos eficientes en otros lenguajes: Python.

#### 6. Mejora en la eficiencia de programas recursivos

- Inmersión de parámetros.
- Desplegado y plegado. Transformación recursivo-iterativa.
- Programación dinámica. Algoritmo de Floyd.

### VOLUMEN DE TRABAJO (HORAS)

#### ACTIVIDADES PRESENCIALES

Actividad	Horas
Teoría	30,00
Prácticas en aula	10,00
Laboratorio	20,00
<b>Total horas</b>	<b>60,00</b>

#### ACTIVIDADES NO PRESENCIALES

Actividad	Horas
Asistencia a otras actividades	0,00
Elaboración de trabajos individuales o en grupo	26,00
Estudio y trabajo autónomo	10,00
Preparación de clases	39,00
Preparación de actividades de evaluación	15,00
Resolución de casos prácticos	0,00
<b>Total horas</b>	<b>90,00</b>

### METODOLOGÍA DOCENTE

En las actividades teóricas de carácter presencial se desarrollarán los temas de la asignatura



proporcionando una visión global e integradora, analizando con mayor detalle los aspectos clave y de mayor complejidad, fomentando, en todo momento, la participación del alumnado. Estas actividades se complementan con actividades prácticas con el objetivo de aplicar los conceptos básicos y ampliarlos con el conocimiento y la experiencia que se vayan adquiriendo durante la realización de los trabajos propuestos. Comprenden los siguientes tipos de actividades presenciales:

- Clases de problemas y cuestiones en aula
- Sesiones de discusión y resolución de problemas y ejercicios previamente trabajados por el alumnado
- Prácticas de laboratorio
- Realización de cuestionarios individuales de evaluación en el aula con la presencia del profesorado.

Además de las actividades presenciales, los estudiantes deberán realizar tareas personales (fuera del aula) sobre: trabajos monográficos, búsqueda bibliográfica dirigida, cuestiones y problemas, así como la preparación de clases y exámenes. Estas tareas se realizarán principalmente de manera individual, con objeto de potenciar el trabajo autónomo, pero adicionalmente se incluirán trabajos que requieran la participación de pequeños grupos de estudiantes (2-4) para fomentar la capacidad de integración en grupos de trabajo.

Se utilizará la plataforma de e-learning (Aula Virtual) de la Universitat de València como soporte de comunicación con el alumnado. A través de ella se tendrá acceso al material didáctico utilizado en clase, así como los problemas y ejercicios a resolver.

## EVALUACIÓN

La evaluación de la asignatura se llevará a cabo mediante el siguiente esquema:

- Evaluación continua (*N\_Continua*), basada en la participación y grado de implicación en el proceso de enseñanza-aprendizaje, teniendo en cuenta la asistencia regular a las actividades presenciales previstas y la resolución de cuestiones y problemas propuestos y trabajos a entregar.
- Prueba objetiva individual (*N\_Examenes*), consistente en varios exámenes, o pruebas de conocimiento, que constarán tanto de cuestiones teórico-prácticas como de problemas. Algunos de los exámenes podrán ser sustituidos por trabajos.
- Evaluación de las actividades prácticas (*N\_Practicas*) a partir de la consecución de objetivos en las sesiones de laboratorio y de problemas, y la elaboración de trabajos/memorias.

La nota final de la asignatura se calculará mediante la siguiente fórmula:

$$\text{Nota Final} = 20\% N_{\text{Continua}} + 50\% N_{\text{Examenes}} + 30\% N_{\text{Practicas}}$$



Es un requisito obtener una nota mínima de 4,5 sobre 10 en N\_Exámenes y N\_Practicas para poder aprobar la asignatura.

La nota de N\_Continua no es recuperable, manteniéndose para la 2ª convocatoria.

En cualquier caso, la evaluación de la asignatura se hará de acuerdo con el Reglamento de evaluación y calificación de la Universitat de València para los títulos de grado y master aprobado por Consejo de Gobierno de 30 de mayo de 2017 (ACGUV 108/2017)

La copia o plagio manifiesto de cualquier actividad que forma parte de la evaluación supondrá la imposibilidad de superar la asignatura, sometiéndose seguidamente a los procedimientos disciplinarios oportunos indicados en el *PROTOCOLO DE ACTUACIÓN ANTE PRÁCTICAS FRAUDULENTAS EN LA UNIVERSITAT DE VALÈNCIA* ([ACGUV 123/2020](#)).

[//www.uv.es/sgeneral/Protocols/C83sp.pdf](http://www.uv.es/sgeneral/Protocols/C83sp.pdf)>ACGUV 123/2020).

## BIBLIOGRAFÍA

- B. Stroustrup, *The C++ Programming Language*, 4ª Ed., Addison-Wesley, 2013.
- T.H. Cormen et al, *Introduction to Algorithms*, 4ª Ed., The MIT Press, 2022.
- D. Vandevoorde, N.M. Josuttis, *C++ Templates: The Complete Guide*, 2ª Ed. Addison-Wesley, 2017.
- N.M. Josuttis, "The C++ Standard Library: A Tutorial and Reference", 2º Ed. Addison-Wesley. 2012.
- P. Deitel, H. Deitel, "C++ How to Program", 9ª Ed, Prentice Hall. 2013.
- S. Meyers, "Effective C++: 55 Specific Ways to Improve your Programs and Designs", 3ª Ed. Addison-Wesley. 2005.
- S. Meyers, "More Effective C++: 35 New Ways to Improve your Programs and Designs", Addison-Wesley. 1995.
- S. Meyers, "Effective STL: 50 Specific Ways to Improve your Use of the Standard Template Library", Addison-Wesley. 2001.
- A. Alexandrescu, "Modern C++ Design: Generic Programming and Design Patterns Applied", Addison-Wesley. 2001.



- M. Gregoire, "Professional C++", John Wiley & Sons. 2018.
- S. Meyers, "Effective Modern C++", Ed. OReilly Media, 2014.
- B. Stroustrup, A tour of C++, 3ª Ed, Addison-Wesley, 2022.