

# Tema 0: Presentación

## Programación 2

---

Grado en Ingeniería Informática  
Universidad de Alicante  
Curso 2025-2026



**¿QUÉ SOMOS?**



**¡ESTUDIANTES  
DE INFORMÁTICA!**



**¿QUÉ EXPECTATIVAS  
TENEMOS?**



**¡PROGRAMAR  
JUEGOS Y HACKEAR!**



**¿QUÉ ESTAMOS HACIENDO?**



imgflip.com

**COUT << "HOLA MUNDO!";**



- Asignatura básica en el Grado de Ingeniería en Informática
- 6 créditos ECTS
- La información principal de la asignatura se puede consultar en:  
<https://cvnet.cpd.ua.es/Guia-Docente/?wlengua=es&wcodasi=34008&scaca=2025-26>
- Los materiales de la asignatura están en Moodle
- Los anuncios y tutorías a través de UACloud y Moodle
- Síguenos en X: @prog2ua

- Coordinador:
  - David Tomás
- Profesores de teoría:
  - Jorge Calvo, Paco Castellanos, Mariano López, David Tomás (ING, I2ADE) y Antonio Toral (VAL)
- Profesores de prácticas:
  - Gonzalo Alcalá, Pablo Cubillas, Félix Fuentes (ING), Felicidad García (VAL), Sandra García, Mariano López, Paco Moreno, Juan Antonio Pérez, Roberto Tardío, David Tomás (ING) y Antonio Toral

- Presenciales:
  - En el despacho o por videoconferencia
  - Horarios de tutorías en <http://www.dlsi.ua.es>
  - Se debe reservar cita con el profesor desde UACloud
- No presenciales:
  - A través de UACloud (se responden lo antes posible)
  - No hacer consultas por correo electrónico

*“C++ hace que sea más difícil dispararte a ti mismo, pero cuando lo haces te vuelas la pierna entera”*

Bjarne Stroustrup, creador de C++

- Tema 1. Introducción
- Tema 2. La clase `string`
- Tema 3. Ficheros
- Tema 4. Memoria dinámica
- Tema 5. Introducción a la programación orientada a objetos

# Objetivos

- Analizar los requerimientos de un problema
- Diseñar y codificar programas de tamaño medio
- Desarrollar la capacidad de abstracción y de generalización
- Estructurar correctamente un programa en funciones y clases
- Identificar las soluciones más eficientes
- Desarrollar programas con un estilo de programación adecuado y comprensible
- Desarrollar capacidad crítica para la verificación del algoritmo
- Usar herramientas básicas de programación
- Adquirir conocimientos elementales de programación orientada a objetos

- Teoría (50% de la nota final):
  - Examen tipo test en aula de teoría
  - Conceptos teóricos y ejercicios cortos
  - Fecha: 27 de mayo de 2026
- Prácticas (50% de la nota final):
  - Tres prácticas:  $p1$ ,  $p2$  y  $p3$
  - Suponen un 30% de la nota de prácticas
  - Examen de prácticas que supone un 70% de la nota de prácticas
  - Fecha: 27 de mayo de 2026
  - Nota prácticas =  $0,1 * (p1 + p2 + p3) + 0,7 * examen$
  - Se proporcionará un autocorrector antes de cada entrega
  - La nota de cada práctica se obtendrá con un corrector automático, que será una ampliación del autocorrector



- Para promediar entre teoría y prácticas, se requiere una nota mínima de 4 en ambas partes
- La nota final debe ser igual o superior a 5 para aprobar
- Algoritmo para calcular la nota final:

```
notaPracticas = 0,1*(p1 + p2 + p3) + 0,7*examen;  
  
if (notaTeoria >= 4 && notaPracticas >= 4)  
    notaFinal = 0.5*notaTeoria + 0.5*notaPracticas;  
else  
    notaFinal= kSUSPENSO;
```

- En caso de suspender en junio:
  - La nota de prácticas se guarda para julio si es mayor o igual que 4
  - La nota de teoría no se guarda para julio
- Convocatoria de julio:
  - Teoría y prácticas: examen el 7 de julio de 2026
  - Prácticas: debes entregar las tres prácticas superando todos los correctores para poder presentarte al examen. La nota final se calcula exclusivamente con la nota del examen

# Normas sobre las prácticas

- Se puede consultar el grupo asignado en UACloud
- No se puede cambiar de grupo (si hay justificación por motivos laborales o familiares se puede solicitar el cambio en la secretaría de la EPS, pero no al profesor)
- Comienzo de las prácticas: lunes 26 de enero
- Las prácticas son individuales
- La asistencia es obligatoria: máximo 3 faltas sin justificar
- Es necesario seguir estrictamente las instrucciones de los enunciados de las prácticas (especialmente en lo referente al formato de la salida)

- La única forma de aprender a programar es programando
- Las prácticas están pensadas para que el alumnado aprenda de forma práctica los conocimientos de la asignatura a lo largo del cuatrimestre
- Un alumno que copia (o al que le hacen las prácticas) raramente aprende y aprueba la asignatura

- Reglamento para la evaluación de los aprendizajes de la UA, Artículo 14.1:

*“Durante la prueba, el alumnado está obligado a observar las normas sobre autenticidad del ejercicio y privacidad del mismo.”*

- Copiarse una práctica de otra persona, total o parcialmente, incumple el Artículo 14.1 (*autenticidad*)
- Difundir una práctica (para que otros se la copien o se inspiren), también incumple el Artículo 14.1 (*privacidad*)

## Sobre las copias en prácticas (3/3)

- Reglamento para la evaluación de los aprendizajes de la UA, Artículo 14.4:

*“En cualquier caso, las actuaciones fraudulentas en una prueba de evaluación darán lugar a la calificación de suspenso, con calificación numérica de cero en dicha prueba, sin perjuicio del procedimiento disciplinario que contra el o la estudiante se pudiera incoar y en su caso la sanción que fuera procedente de acuerdo con la legislación vigente.”*

- Los alumnos implicados en copias tendrán un 0 en su práctica y se mandará un informe a la EPS para que se tomen medidas disciplinarias

# Uso de la IA en Programación 2

- Sabemos que usáis **ChatGPT** (o similares) para ayudaros a desarrollar vuestro código
- Es una herramienta que va a ser fundamental en vuestra carrera como programadores/as y está bien que la uséis, pero...
- **¿Qué se permite hacer con IA en Programación 2?**
  - Mientras estás programando
    - Obtener explicaciones sobre el uso de funciones
    - Documentar tu código
    - Revisar el estilo
  - Una vez terminado el código
    - Crear tests
    - Depurar/optimizar
- **¿Qué NO puedes hacer con IA en Programacion 2?**
  - Pedirle que te haga tu código pasándole el enunciado sin más...
  - ... porque no vas a aprender...
  - ... **y el día del examen no te vamos a dejar usar la IA**

# Planificación temporal

Lunes	Miércoles	Jueves	Viernes	Entrega
26/01 T0	28/01 T0	29/01 T0	30/01 T0	-
02/02 T1 (1)	04/02 T1 (1)	05/02 T1 (1)	06/02 T1 (1)	-
09/02 T1 (2)	11/02 T1 (2)	12/02 T1 (2)	13/02 T1 (2)	-
16/02 T1 (3)	18/02 T1 (3)	19/02 T1 (3)	20/02 T1 (3)	p1
23/02 T2	25/02 T2	26/02 T2	27/02 T2	-
02/03 T3 (1)	04/03 T3 (1)	05/03 T3 (1)	06/03 T3 (1)	-
09/03 T3 (2)	11/03 T3 (2)	12/03 T3 (2)	13/03 T3 (2)	-
16/03 T4	18/03 T4	19/03 -	20/03 T4	-
23/03 T5 (1)	25/03 T5 (1)	26/03 T4	27/03 T5 (1)	p2
30/03 Ejer.	01/04 Ejer.	02/04 -	03/04 -	-
06/04 -	08/04 -	09/04 -	10/04 -	-
13/04 -	15/04 Ejer.	16/04 -	17/04 -	-
20/04 T5 (2)	22/04 T5 (2)	23/04 T5 (1)	24/04 T5 (2)	-
27/04 T5 (3)	29/04 T5 (3)	30/04 T5 (2)	01/05 -	-
04/05 T5 (4)	06/05 T5 (4)	07/05 T5 (3)	08/05 T5 (3)	-
11/05 T5 (5)	13/05 T5 (5)	14/05 T5 (4)	15/05 T5 (4)	-
18/05 Ejer.	20/05 Ejer.	21/05 T5 (5)	22/05 T5 (5)	p3



## Para aprobar Programación 2...

- Tienes que practicar mucho
- Tienes que hacer los ejercicios de teoría y las prácticas
- No puedes aprobar empollando una semana antes
- Preguntas tus dudas en clase de teoría y prácticas, sin miedo
- Si te pierdes, haz tutorías presenciales o virtuales