

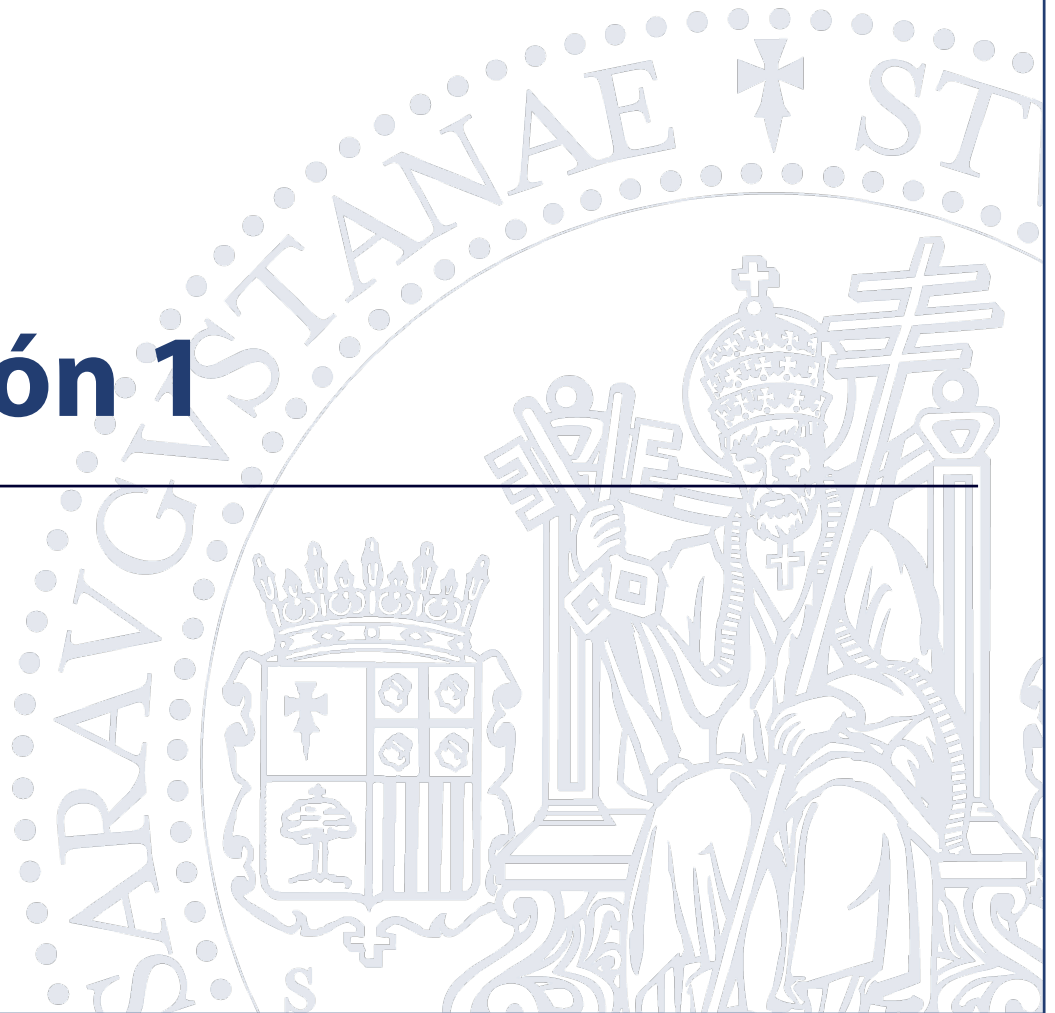
# Programación 1

---

Grado en Ingeniería Informática



Escuela de  
Ingeniería y Arquitectura  
**Universidad Zaragoza**



# Profesorado de la asignatura

- Miguel Ángel Latre
  - [latre@unizar.es](mailto:latre@unizar.es)
  - Responsable de la asignatura
  - Teoría (411)
  - 3 grupos de problemas (412)
  - 5 grupos de prácticas (411, 412 y 413)
- Departamento de Informática e Ingeniería de Sistemas
  - Área de Lenguajes y Sistemas Informáticos



# Profesorado de la asignatura

---

- Rafael Tolosana
  - [rafaelt@unizar.es](mailto:rafaelt@unizar.es)
  - Teoría (412)
  - Problemas (411)
  - Prácticas (412/413)
- Departamento de Informática e Ingeniería de Sistemas
  - Área de Lenguajes y Sistemas Informáticos



# Profesorado de la asignatura

---

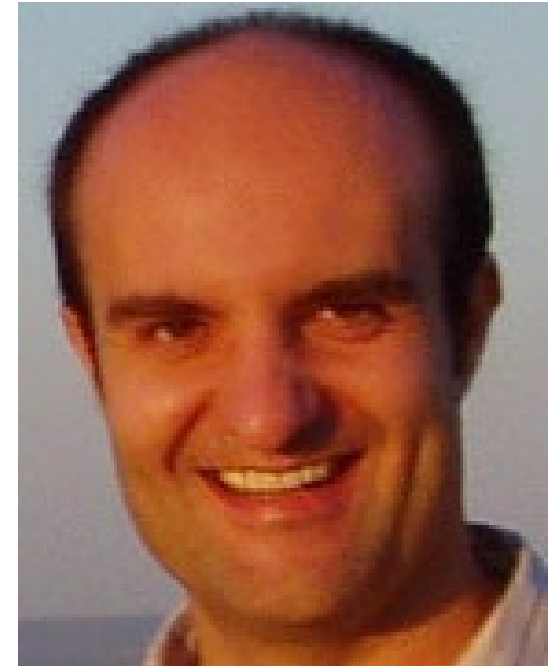
- Ignacio Huitzil
  - [ihuitzil@unizar.es](mailto:ihuitzil@unizar.es)
  - Teoría (413)
- Departamento de Informática e Ingeniería de Sistemas
  - Área de Lenguajes y Sistemas Informáticos



# Profesorado de la asignatura

---

- José Merseguer
  - [jmerse@unizar.es](mailto:jmerse@unizar.es)
  - 2 grupos de problemas (411)
  - 2 grupo de prácticas (413)
- Departamento de Informática e Ingeniería de Sistemas
  - Área de Lenguajes y Sistemas Informáticos



# Profesorado de la asignatura

---

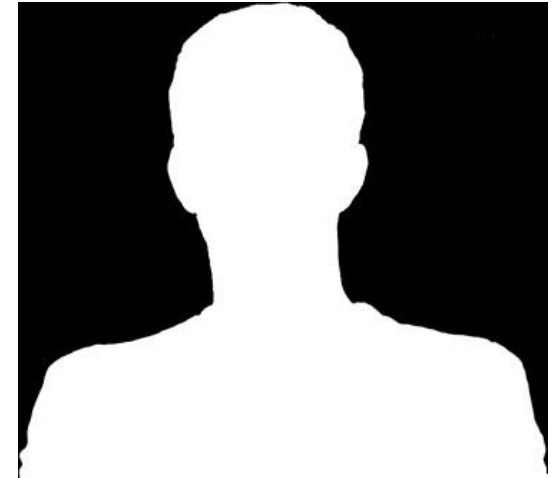
- Javier Zarazaga
  - [javy@unizar.es](mailto:javy@unizar.es)
  - 1 grupo de prácticas (412)
- Departamento de Informática e Ingeniería de Sistemas
  - Área de Lenguajes y Sistemas Informáticos



# Profesor ayudante doctor

---

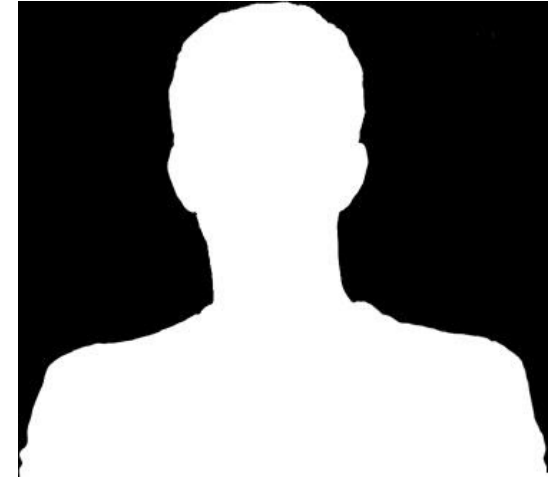
- ???
  - ???
  - 4 grupos de prácticas  
(411, 412, 413)
- Departamento de Informática e Ingeniería de Sistemas
  - Área de Lenguajes y Sistemas Informáticos



# Profesor interino

---

- ???
  - ???
  - 6 grupos de prácticas  
(411, 412, 413)
- Departamento de Informática e Ingeniería de Sistemas
  - Área de Lenguajes y Sistemas Informáticos





# Programación 1

---

- Asignatura de formación básica (obligatoria)
  - 6,0 créditos ECTS
  - Primer paso en el aprendizaje de la programación de computadores (se parte de cero)
  - **Continuidad** en asignaturas posteriores
  - **Imprescindible** para estudiar prácticamente cualquier otra materia de las que intervienen en los estudios



# Planteamiento

---

- ❑ **Diseño de programas** que resuelvan **problemas de tratamiento de información**
- ❑ Presentación de herramientas para **representar información** y para **tratarla**
- ❑ Planteamiento de **metodologías** para facilitar y sistematizar el **análisis de problemas** y el **diseño de programas** que los resuelvan
- ❑ Más información: *Guía docente curso 2025–2026 de Programación 1*
  - [https://estudios.unizar.es/estudio/asignatura?anyo\\_academico=2025&asignatura\\_id=30204&estudio\\_id=20250148&centro\\_id=110&plan\\_id\\_nk=439](https://estudios.unizar.es/estudio/asignatura?anyo_academico=2025&asignatura_id=30204&estudio_id=20250148&centro_id=110&plan_id_nk=439)



# Objetivos

---

- ❑ Conocer los **conceptos básicos** ligados a la programación
- ❑ Saber **analizar problemas** concretos, **plantear soluciones** y **desarrollar programas** que permitan resolverlos en un computador
- ❑ Aprender a utilizar y familiarizarse con un **entorno tecnológico** específico:
  - Lenguaje **C++** y entorno integrado **Visual Studio Code**
- ❑ Ejercitarse como **programador**



# Programa

---

## **I. Conceptos y elementos básicos de Programación**

- Problemas de tratamiento de información, algoritmos y programas
- Lenguajes de programación y ejecución de un programa
- Información, datos, operaciones y expresiones

## **II. Diseño de los primeros programas**

- Diseño de algunos programas elementales
- Instrucciones simples y estructuradas
- Problemas de cálculos con enteros
- Desarrollo modular y descendente de programas
- Problemas de cálculo con números reales



# Programa

---

## **III. Diseño de programas que trabajan con estructuras de datos**

- Vectores
- Cadenas de caracteres
- Registros
- Algoritmos básicos de trabajo con vectores

## **IV. Diseño de programas que trabajan con ficheros**

- Entrada y salida de datos
- Ficheros de texto
- Ficheros binarios
- Ficheros: otras posibilidades

## **V. Metodología de diseño de programas**



# Metodología de aprendizaje

---

- ❑ Clases de teoría
- ❑ Clases de problemas
- ❑ Clases prácticas



# Clases de teoría

---

- Aprendizaje de conceptos y metodologías de diseño de programas
  - 2 horas semanales
  - Asistencia voluntaria



# Clases de teoría

---

- Grupo 411
  - Aula 17 TQ
  - Miércoles y viernes 9:00 a 9:50
- Grupo 412
  - Aula 17 TQ
  - Miércoles y viernes de 16:00 a 16:50
- Grupo 413
  - Aula 26 TQ
  - Miércoles de 18:00 a 18:50
  - Viernes de 14:00 a 14:50





# Clases de problemas

---

- Aplicación de conceptos y metodologías al diseño de algoritmos y programas
  - 1 hora semanal
  - Con grupos desdoblados en 2 o 3 subrupos, en coordinación con IC
  - Asistencia voluntaria



# Clases de problemas

## Grupo 411

---

Grupo	Horario	Aula	Profesor	Inicio 1.º apellido
4112	Jueves 8:00 a 8:50	24 TQ	José Merseguer	
4111	Jueves 9:00 a 9:50	17 TQ	José Merseguer	
4113	Viernes 12:00 a 12:40	18 TQ	Rafael Tolosana	



# Clases de problemas

## Grupo 412

---

Grupo	Horario	Aula	Profesor	1.º apellido INF	1.º apellido MATINF
4123	Miércoles 15:00 a 15:50	17 TQ	Miguel Latre		
4122	Jueves 17:00 a 17:50	24 TQ	Miguel Latre		
4121	Jueves 18:00 a 18:50	17 TQ	Miguel Latre		



# Clases de problemas

## Grupo 413

---

Grupo	Horario	Aula	Profesor	Inicio 1.º apellido
4131	Miércoles 14:00 a 14:50	26 TQ	Rafael Tolosana	
4132	Jueves 14:00 a 14:50	22 TQ	Rafael Tolosana	



# Clases prácticas

---

- Tecnología necesaria para programar, ejecutar y depurar utilizando
  - Un lenguaje concreto: C++
  - Un entorno concreto: Visual Studio Code
- 6 prácticas
- 9 sesiones de 2 horas
- **Asistencia voluntaria**
- Individuales



# Grupos de prácticas

## Grupo 411

---

- 6 subgrupos de prácticas

Subgrupos	Horario	Laboratorio	Profesores
13;15	Lunes de 8:00 a 9:50	L.0.04, edificio Ada Byron	
14;16	Martes de 8:00 a 9:50	L,0.03, edificio Ada Byron	
11;12	Martes de 10:00 a 11:50	L.0.01, edificio Ada Byron	



# Grupos de prácticas

## Grupo 412

- 6 subgrupos de prácticas

Subgrupos	Horario	Laboratorio	Profesores
23	Lunes de 15:00 a 16:50	L0.05b, edificio Ada Byron	
24;25	Lunes de 15:00 a 16:50	L0.01, edificio Ada Byron	
26	Lunes de 19:00 a 20:50	L0.03, edificio Ada Byron	
21;22	Martes de 15:00 a 16:50	L0.04, edificio Ada Byron	



# Grupos de prácticas

## Grupo 413

---

- 5 subgrupos de prácticas

Subgrupos	Horario	Laboratorio	Profesores
35	Lunes de 19:00 a 20:50	L0.03, edificio Ada Byron	
33;34	Martes de 17:00 a 18:50	L0.04, edificio Ada Byron	
31;32	Viernes A de 15:00 a 16:50	L0.03, edificio Ada Byron	





# Fechas de las prácticas

Práctica	Lunes	Martes	Viernes
Práctica 1	15-9-2025	16-9-2025	12-9-2025
Práctica 2	29-9-2025	30-9-2025	26-9-2025
Práctica 3	15-10-2025	14-10-2025	17-10-2025
	20-10-2025	21-10-2025	24-10-2025
Práctica 4	27-10-2025	28-10-2025	31-10-2025
	3-11-2025	4-11-2025	7-11-2025
Práctica 5	17-11-2025	18-11-2025	21-11-2025
Práctica 6	24-11-2025	25-11-2025	28-11-2025
	1-12-2025	2-12-2025	5-12-2025

No hay prácticas la 1.ª semana B, la 2.ª semana B y la 5.ª semana A

Laboratorios  
edificio Ada Byron

# Localización

Parada tranvía  
«Campus Río Ebro»

Aulas 17 y 26,  
edificio Torres  
Quevedo

Fuente: SIGEUZ 3.1. Universidad de Zaragoza (<https://sigeuz.unizar.es/>)

# Clases prácticas

---

- Prácticas **individuales**
- **Preparación** de las prácticas (antes de la sesión):
  - Lectura completa de los guiones de prácticas
  - Resolución de los problemas en la medida de lo posible
  - Identificación de dudas
- **Durante las sesiones** de prácticas:
  - Consulta de dudas al profesor
  - Supervisión del trabajo por el profesor
  - Finalización del trabajo de prácticas



# Prácticas

---

- Las prácticas son de **asistencia voluntaria**.
- Las prácticas son de **entrega obligatoria**.
  - En el enunciado de cada práctica se indicará qué hay que entregar en concreto.
- **Plazo de entrega genérico:** sábados de las semanas B antes de las 18:00, a través de Moodle
  - En el enunciado de cada práctica se indicará la fecha en concreto.



# Prácticas

---

- Un 30 % de la calificación obtenida con los trabajos que se corrijan forma parte de la calificación de la convocatoria ordinaria.



# Tutorías

---

- Tutorías académicas
  - Supervisión del trabajo de los alumnos, orientación, resolución de dudas, recomendación de bibliografía, revisión de trabajos y pruebas, etc., dentro del ámbito de la asignatura
  - En general, de forma presencial o telemática (preguntad al profesor).
- Horarios y mecanismos de reserva publicados en Moodle



# Sobre el uso del correo electrónico

---

- Medio de comunicación **asíncrono**
  - El emisor manda el mensaje en un determinado momento.
  - El receptor lo lee en un momento posterior.
  - El receptor puede contestar, si lo estima pertinente, todavía más tarde.



# Sobre el uso del correo electrónico

---



**Pablo Simón**

@kanciller

...

Consejo a estudiantes: para escribir a un email a un profesor sígase el "método Montoya" como ejemplo de buenas prácticas:

- "Hola, me llamo Iñigo Montoya. Tú mataste a mi padre. Prepárate a morir"

Estructura del email - Saludo. Presentación. Exposición del caso. Requerimiento.

10:48 a. m. · 12 oct. 2022

<https://x.com/kanciller/status/1580118074933465089>



Escuela de  
Ingeniería y Arquitectura  
**Universidad Zaragoza**



# Carga de trabajo

---

- Estimación de **150 horas efectivas** de trabajo:
  - 57 horas de **actividades presenciales activas**
    - clases «teóricas»
    - clases de problemas
    - prácticas
  - 87 horas de **estudio personal efectivo**
    - estudio de apuntes y textos
    - resolución de problemas
    - preparación clases y prácticas
    - desarrollo de programas
  - 6 horas de pruebas de evaluación



# Proceso de aprendizaje

---

- El aprendizaje de la programación exige un **trabajo continuado desde el primer día de clase:**
  - **comprensión** de conceptos,
  - **análisis y la resolución de problemas** utilizando lápiz y papel,
  - **puesta a punto en computador** de un buen número de programas.
- *“A programar se aprende programando”*



# ¿Qué hacer para aprender?

---

- Asistir a clase
  - Habiendo realizado el trabajo previo que se haya encargado
  - Atendiendo y participando activamente en la clase
  - Repasando después y comprendiendo cada lección
- Resolver los problemas de programación propuestos en las clases de problemas, en las prácticas y en los trabajos obligatorios de otros años:
  - Lápiz y papel
  - Programación en computador
  - Validación del código desarrollado
- Colaborar con otros compañeros y consultarles
- Consultar dudas a los profesores en sus horarios de tutorías



# Actividades de evaluación

---

- **P1. Examen escrito**
  - Nota mínima: 5,0
  - Ponderación: 70 %
- **Parte práctica**
  - Ponderación: 30 %
  - **P2: Evaluación continua**
    - Solo en convocatoria ordinaria
    - Entrega de prácticas.
  - **P3: Examen global de prácticas**
    - En convocatoria ordinaria o extraordinaria
    - Examen de programación en laboratorio
- **Pruebas voluntarias**
  - Entrega de problemas voluntarios junto con las prácticas
  - 10 % adicional **solo en convocatoria ordinaria** a la calificación final de cada estudiante que **haya aprobado** con las calificaciones P1+P2 o P1+P3.



# Actividades de evaluación

---

- Las calificaciones del estudiante obtenidas en la primera convocatoria en cualquiera de las pruebas (P1 y P2/P3) se mantienen en la segunda convocatoria, salvo que el estudiante opte por presentarse a la prueba correspondiente en esta nueva convocatoria, en cuyo caso prevalecerá la nueva calificación.



# Actividades de evaluación

---

- Fecha examen escrito y global de prácticas
  - Convocatoria ordinaria: 13-1-2026, mañana
  - Convocatoria extraordinaria: 23-6-2026, tarde



# Bibliografía básica

---

- **Javier Martínez:** *Curso de Programación 1*. 2017
- **Profesores de la asignatura y Javier Martínez:** *Prácticas de Programación 1*. Moodle. 2025
- **Profesores de la asignatura, Javier Martínez y otros:** *Diversos materiales docentes.*
  - Curso en Moodle:  
<https://moodle.unizar.es/add/course/view.php?id=137684>



# Bibliografía de consulta

---

- Páginas web con documentación sobre el lenguaje C++
  - <https://www.cplusplus.com/>
  - <https://www.cprogramming.com/>
  - [https://es.wikibooks.org/wiki/Programación\\_en\\_C++](https://es.wikibooks.org/wiki/Programación_en_C++)
- Manual en línea de bibliotecas predefinidas en C++
  - <https://www.cplusplus.com/reference/>
- Entorno de ejecución en línea
  - <https://cpp.sh/>
- Entorno de visualización de la ejecución en línea
  - <https://pythontutor.com/cpp.html>
- Hay muchos textos para apoyar la enseñanza de un primer curso de programación utilizando el lenguaje C++.



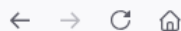


# Curso en Moodle

---

- <https://moodle.unizar.es/add/course/view.php?id=137684>





## &gt; General

## &gt; Transparencias, pla...

Destacado

## &gt; Prácticas

## &gt; Material adicional por temas

## &gt; Programar en C++

## &gt; Ejercicios básicos

## v Exámenes y trabajos de cursos...

Exámenes anteriores

Exámenes actualizadosEnunciad...

Examen actualizado de 1.ª conv...

Examen actualizado de 2.ª conv...

Examen actualizado de 1.ª conv...

Examen actualizado de 2.ª conv...

Examen actualizado de 1.ª conv...

Examen actualizado de 2.ª conv...

Examen actualizado de 1.ª conv...

Examen actualizado de 2.ª conv...

Examen de 1.ª convocatoria del ...

Examen de 2.ª convocatoria del ...

Enunciado y soluciones comenta...

## Programación 1 (2025/2026)

Curso

Participantes

Calificaciones

Mi asistente

## General



Avisos

Guía docente de la asignatura

Presentación de la asignatura

866.3 KB

## Tutorías de los profesores

Deben reservarse con antelación a través de los siguientes calendarios Google.

## • Miguel Ángel Latre

- Despacho: D.2.22 (edificio Ada Byron)
- Calendario tutorías: <https://calendar.app.google/GuG3MUA5gmfDyLup7>
  - Pueden ser en modo presencial o remoto. Es necesario indicar este modo al realizar la reserva. Las tutorías remotas se celebrarán en la sala de Google Meet creada al reservar la cita.

## • Rafael Tolosana

- Despacho D0.01 (edificio Ada Byron)
- Horario de tutorías:



- Con cita previa (al menos 24 horas antes) a través de la página web de citas de Google. En la reserva, especificar nombre de asignatura, tema a tratar y si presencial o remoto.

## • Iñacio Huitzil



# Curso en Moodle

---

- ❑ Presentación de la asignatura y guía docente
- ❑ Horarios de tutorías
- ❑ Planificación, transparencias del curso y enunciados de problemas
- ❑ Acceso a todo el código fuente
  - Ejemplos de teoría
  - Soluciones a los problemas
  - Código base para las prácticas
- ❑ Material adicional por temas
  - Capítulos de los apuntes del profesor Javier Martínez
  - Enlaces a tutoriales de Cplusplus.com
- ❑ Enunciados de prácticas
- ❑ Material sobre C++ y Visual Studio Code
- ❑ Ejercicios básicos
- ❑ Exámenes resueltos



# Para la próxima clase...

---

- ❑ Buscad y leed atentamente una receta para hacer tortilla de patata



File:Tortilla de Patatas (Corte transversal).jpg. (9 de marzo de 2017).  
*Wikimedia Commons, the free media repository.* Accedido el 18 de septiembre de 2017.  
[https://commons.wikimedia.org/w/index.php?title=File:Tortilla de Patatas \(Corte transversal\).jpg&oldid=236535592](https://commons.wikimedia.org/w/index.php?title=File:Tortilla_de_Patatas_(Corte_transversal).jpg&oldid=236535592).



# Para la primera práctica...

---

- Podéis instalaros ya el compilador de C++ y el entorno Visual Studio Code
  - Instrucciones en Moodle
    - [Sección Programar en C++](#)

