

# Programación 1

---

Grado en Ingeniería Informática — Curso 2018-19

<https://webdiis.unizar.es/asignaturas/PROG1/>



Escuela de  
Ingeniería y Arquitectura  
**Universidad Zaragoza**



# Profesores de la asignatura

---

- **Miguel Ángel Latre**
  - Profesor responsable de la asignatura
  - [latre@unizar.es](mailto:latre@unizar.es)
- **Simona Bernardi**
  - [simonab@unizar.es](mailto:simonab@unizar.es)
- **Alfredo Roy**
  - [argolet@unizar.es](mailto:argolet@unizar.es)
- Área de Lenguajes y Sistemas Informáticos
- Departamento de Informática e Ingeniería de Sistemas



# 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)
  - Tiene **continuidad** en asignaturas posteriores de la materia común de *Computación y Programación*
    - Programación 1 → Programación II → Estructuras de Datos y Algoritmos → Tecnología de la Programación → Ingeniería del Software → Proyecto Software → etc.



# Programación 1

---

- La programación:
  - Uno de los **pilares básicos** de la formación de Ingenieros en Informática
  - **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



# 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** para el desarrollo de programas:
  - Lenguaje **C++** y entorno integrado **Code::Blocks**
- ❑ Ejercitarse como **programador** de computadores de forma individual y en equipo aplicando unas determinadas metodologías.

# Resultados de aprendizaje

---

Esta asignatura acredita que quien la apruebe:

1. Conoce **conceptos básicos** ligados a la programación y la informática.
2. Comprende, analiza y resuelve **problemas de tratamiento de información de complejidad baja o media** y construye algoritmos que los resuelven.
3. Define las **estructuras de datos** más adecuadas para representar la información asociada a cada problema.
4. Diseña de forma **descendente** y documenta las **acciones** algorítmicas que resuelven cada problema de forma eficaz y eficiente.
5. Conoce algoritmos para resolver los **problemas más frecuentes** que se presentan al trabajar con **estructuras de datos secuenciales e indexadas**.

# Resultados de aprendizaje

---

Esta asignatura acredita que quien la apruebe (cont.):

6. Conoce y comprende la **sintaxis** y la **semántica** de las construcciones básicas de un **lenguaje de programación: C++**
7. Escribe programas con **buen estilo**, con una **documentación** adecuada, con los comentarios precisos y con las **especificaciones** necesarias.
8. Sabe utilizar **herramientas de edición, compilación, depuración y ejecución** para desarrollar programas, así como sistemas operativos y otros programas con aplicación en ingeniería.
9. Utiliza **estrategias para corregir los programas** cuando no funcionan bien.
10. Pone a punto y ejecuta en un computador programas escritos en un lenguaje de programación determinado: **C++**



# Resumiendo

---

- Aprenderemos algunos **conceptos básicos** ligados a la programación de computadores.
- Aprenderemos a **analizar problemas** concretos, a **plantear soluciones** y a **desarrollar programas** que permitan resolverlos en un computador.
- Aprenderemos a utilizar una **tecnología** para el desarrollo de programas (lenguaje **C++** y entorno integrado **Code::Blocks**).
- Nos ejercitaremos como **programadores** de computadores de forma individual y en equipo, aplicando siempre unas determinadas **metodologías**.



# 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**

- Estructuración indexada de datos
- Representación de cadenas de caracteres
- Estructuración agregada de datos
- Algoritmos básicos de trabajo con estructuras de datos indexadas

## **IV. Diseño de programas que trabajan con datos almacenados en ficheros**

- Entrada y salida de datos
- Trabajo con ficheros de texto
- Trabajo con ficheros binarios
- Trabajo con ficheros: otras posibilidades



# Proceso de aprendizaje

---

- El estudio y **trabajo continuado desde el primer día de clase.**
- El **aprendizaje de conceptos y metodologías de diseño** de programas a través de las clases magistrales.
- La **aplicación** de tales conocimientos al diseño de algoritmos y programas en las clases de problemas.
- Las prácticas en laboratorio en las que el alumno aprenderá, mediante su trabajo previo y en el propio laboratorio, la **tecnología** necesaria para poner a punto programas utilizando C++.
- El **trabajo en equipo** analizando y resolviendo problemas de programación.



# Proceso de aprendizaje

---

- El aprendizaje de la programación exige un **trabajo continuado**:
  - **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”*



# Grupos de prácticas y problemas (grupo de mañanas)

---

- Nuevos estudiantes:
  - Grupo 1: Apellido entre [**AAA-BES**]
  - Grupo 2: Apellido entre [**BET-CON**]
  - Grupo 3: Apellido entre [**COÑ-GÓMEZ ORT**]
  - Grupo 4: Apellido entre [**GÓMEZ ORU-ZZZ**]
- Resto de estudiantes:
  - Confirmar con el profesor la adscripción a un grupo



# Horario (grupo de mañanas)

---

- **Clases teóricas**

2 horas semanales

Aula 18 del edificio Torres Quevedo

- Lunes y viernes 12:10 a 13:00

- **Clases de problemas**

1 hora semanal

- Grupos 3 y 4: jueves de 12:10 a 13:00 (aula 19)

- Grupos 1 y 2: jueves de 13:10 a 14:00 (aula 18)



# Horario (grupo de mañanas)

---

## □ **Clases prácticas de laboratorio**

2 horas cada dos semanas

Horario: 10:00–12:00

Laboratorio L.0.04 del edificio Ada Byron

■ Grupo 1: lunes A

**Corregido**

■ Grupo 2: lunes B

**Corregido**

■ Grupo 3: martes A

**Corregido**

■ Grupo 4: martes B

**Corregido**





# Grupos de prácticas y problemas (grupo de tardes)

---

- Nuevos estudiantes:
  - Grupo 1: Apellido entre [**AAA-LUO**]
  - Grupo 2: Apellido entre [**LUP-PEL**]
  - Grupo 3: Apellido entre [**PEM-SOL**]
  - Grupo 4: Apellido entre [**SOM-ZZZ**]
- Resto de estudiantes:
  - Confirmar con el profesor la adscripción a un grupo



# Horario (grupo de tardes)

---

## □ **Clases teóricas**

2 horas semanales

Aula 18 del edificio Torres Quevedo

- Martes de 17:10 a 18:00
- Miércoles de 18:10 a 19:00

## □ **Clases de problemas**

1 hora semanal

- Grupos 3 y 4: jueves de 18:10 a 19:00 (aula 14)
- Grupos 1 y 2: jueves de 19:10 a 20:00 (aula 18)



# Horario (grupo de tardes)

---

## □ **Clases prácticas de laboratorio**

2 horas cada dos semanas

Horario: 15:00–17:00

Laboratorio L.0.04 del edificio Ada Byron

■ Grupo 1: martes A

Profesor: Miguel Latre

■ Grupo 2: martes B

Profesor: Simona Bernardi

■ Grupo 3: lunes A

Profesor: Alfredo Roy

■ Grupo 4: lunes B

Profesor: Alfredo Roy



# Prácticas

---

- **Puestos de trabajo del laboratorio L.0.04**
  - Equipos con sistema operativo **Linux (CENT OS)**, Windows, ...
  - **Clúster** (hendrix) hace funciones de **servidor de ficheros**
  - **Cuenta** para el acceso a los equipos y al clúster
    - El administrador del DIIS informará del nombre y contraseña de la cuenta por correo electrónico a la **cuenta de UNIZAR**
- **¿Dónde desarrollar programas C++?**
  - Trabajando **desde cualquier puesto del L.0.04** (o de otro laboratorio del DIIS) con el **entorno de desarrollo de programas Code::Blocks**
  - Trabajando **desde tu computador personal** con el **entorno de desarrollo de programas Code::Blocks**



# Prácticas

---

- Correo electrónico enviado cuando se abre la cuenta:
  - De: "J.A. Gutierrez" <jagutierrez at unizar.es>
  - Para: <.....@unizar.es>
  - Asunto: Cuenta en el nuevo servidor de docencia del DIIS.
  - Cuerpo:
    - Hola
    - Por estar cursando alguna asignatura impartida por el Dpto. de Ingeniería de Sistemas e Informática, se te ha creado una cuenta en el servidor de practicas "hendrix".
    - Los detalles de su utilización te los dará la persona que imparte las asignaturas en cuestión.
    - Datos de acceso:
      - Nombre de usuario: XXXX
      - Clave de entrada: XXXX
      - Nombre de la persona: XXXX



# Prácticas

---

- Prácticas **individuales**
  - Es positivo intercambiar experiencias y resolver dudas con compañeros
- **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



# 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

## □ Lugar

- Despacho D.2.22, edificio Ada Byron

## □ Horario

- Martes: 12:00 a 13:00
- Miércoles y jueves: 16:30 a 18:00
- Viernes: 10:00 a 12:00
- <http://webdiis.unizar.es/~latre/tutorias>





# Localización

Despacho  
D.2.22, edificio  
Ada Byron

Laboratorio  
L.0.04, edificio  
Ada Byron

Aula 19, edificio  
Torres Quevedo

Aula 18, edificio  
Torres Quevedo

Parada tranvía  
«Campus Río Ebro»

Aula 14, edificio  
Torres Quevedo

Mapa: 675888.72, 4617018.2 (UTM30N, EDE0) - Escala 1:2625

Datos cartográficos base suministrados por el Servicio de Información Geográfica del Ayuntamiento de Zaragoza. Otras fuentes de datos puntuales: [OpenStreetMap](#), [CC-BY-SA](#)



# Carga de trabajo

---

- Estimación de **150 horas efectivas** de trabajo:
  - 56 horas de **actividades presenciales activas**
    - clases «teóricas»
    - clases de problemas
    - prácticas en laboratorio
  - 91 horas de **estudio personal efectivo**
    - estudio de apuntes y textos
    - resolución de problemas
    - preparación clases y prácticas
    - desarrollo de programas
  - 3 horas de **examen final escrito**



# ¿Qué hacer para aprender?

---

- Asistir a clase
  - Habiendo leído previamente cada lección
  - Atendiendo y participando activamente en la clase
  - Repasando después y comprendiendo cada lección
- Resolver los problemas de programación propuestos como trabajos voluntarios evaluables
- Resolver los problemas de programación de cada práctica:
  - 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 Prueba escrita**
  - Nota mínima: 4,0
  - Ponderación: 70%
- **P2 Trabajos y prueba de programación en laboratorio.**
  - Ponderación: 30%
  - Febrero
    - Trabajos específicos de programación. Ponderación 15%
    - Prueba de programación en laboratorio. Ponderación 15%
  - Septiembre
    - Prueba de programación en laboratorio. Ponderación 30%
- **Pruebas voluntarias**
  - A lo largo del cuatrimestre
  - Permiten subir la nota, hasta 1 punto, a quienes aprueben en febrero y hayan hecho todas las pruebas



# Actividades de evaluación

---

- Las calificaciones obtenidas en febrero en las pruebas P1 y P2 se mantienen en septiembre, salvo que se opte por presentarse a la prueba correspondiente en septiembre, en cuyo caso prevalecerá la nueva calificación



# Bibliografía básica

---

- **Javier Martínez:** *Curso de Programación 1*. Servicio de reprografía de la EINA. 2017
- **Miguel Ángel Latre y Javier Martínez:** *Prácticas de Programación 1*. Web de la asignatura. 2018
- **Miguel Ángel Latre y Javier Martínez:** *Diversos materiales docentes*. Web de la asignatura: <http://webdiis.unizar.es/asignaturas/PROG1>



# Bibliografía de consulta

---

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



# Web de la asignatura

<https://webdiis.unizar.es/asignaturas/PROG1/>



**Programación I**

Asignatura de Programación I

El lunes 17 de septiembre, inicio del curso de Programación I

leave a comment

Las clases de **Programación I** del curso 2018-19 comienzan el próximo lunes 17 de septiembre de 2015 y se imparten en el aula 18 del edificio Torres Quevedo. En la primera clase se presentará la asignatura y se darán las directrices sobre su organización y el modo de abordar su estudio:

- Lunes 17 de septiembre a las 13:00 para los alumnos del grupo de mañana de primer curso.
- Lunes 17 de septiembre a las 19:00 para los alumnos del grupo de tarde de primer curso.

Las siguientes clases se impartirán según los horarios habituales del cuatrimestre de otoño.

Durante la primera semana de curso, del 17 al 21 de septiembre, no hay programadas sesiones de prácticas. Las primeras sesiones de prácticas de Programación I se celebrarán a partir del lunes 24 de septiembre. Es de vital importancia preparar y participar en la primera sesión de prácticas de la asignatura ya que en ella se ha de aprender el manejo de las herramientas de programación que van a ser utilizadas a lo largo de todo el cuatrimestre.

Written by profesor [Edit](#)  
Septiembre 12th, 2018 at 5:03 pm

Posted in [curso 2018-19](#), [información académica](#)

**Calificaciones de Programación I (septiembre)**

leave a comment

Se pueden consultar las [calificaciones](#) correspondientes a la segunda convocatoria del curso 2017-18. También se pueden consultar los enunciados y un juego de soluciones de los [exámenes escrito y práctico](#) celebrados el pasado 7 de septiembre.

La revisión de exámenes se celebrará el jueves 13 de septiembre a las 13:00 en la sala de juntas del departamento de Informática e Ingeniería de Sistemas (planta 1ª del edificio Ada Byron).

Written by prog1 [Edit](#)  
Septiembre 11th, 2018 at 6:27 pm

Posted in [información académica](#)

**Información**

- [La asignatura](#)
- [Programa](#)
- [Horarios y organización de clases y sesiones prácticas](#)
- [Profesores](#)
- [Evaluación de la asignatura](#)
- [Sistema de evaluación](#)
- [Convocatorias de examen](#)
- [Calificaciones](#)
- [Estadísticas](#)
- [Material docente](#)
- [Código C++ y datos](#)
- [Transparencias](#)
- [Programar en C++](#)
- [Documentación C++](#)
- [Restricciones en el uso de recursos del lenguaje C++](#)
- [Entorno de programación en línea](#)

**Enlaces**

- [Coordinador de la titulación](#)
- [Grado en Ingeniería Informática](#)
- [La EINA \(Escuela de Ingeniería y Arquitectura\)](#)
- [Universidad de Zaragoza](#)

## Información

[La asignatura](#)

[Programa](#)

[Horarios y organización de clases y sesiones prácticas](#)

[Profesores](#)

[Evaluación de la asignatura](#)

[Sistema de evaluación](#)

[Convocatorias de examen](#)

[Calificaciones](#)

[Estadísticas](#)

[Material docente](#)

[Código C++ y datos](#)

[Transparencias](#)

[Programar en C++](#)

[Documentación C++](#)

[Restricciones en el uso de recursos del lenguaje C++](#)

[Entorno de programación en línea](#)

<https://moodle2.unizar.es/add/course/view.php?id=24245>



Escuela de  
Ingeniería y Arquitectura  
**Universidad Zaragoza**

# Problemas, algoritmos y programas

---

- Problemas de **tratamiento de información**
  - Objetivo: resolución **automática** del problema
  - ¿Quién? Un computador
  - Necesidad de **programarlo**





# Algoritmo

---

- Conjunto de operaciones

- ordenado,
- finito,
- carente de ambigüedades,

que permite hallar la solución de un problema de tratamiento de información



# Ejemplo

---

- ❑ Resolución de un problema ~~de tratamiento de información~~
- ❑ Hacer una 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).



Escuela de  
Ingeniería y Arquitectura  
**Universidad Zaragoza**