

# Programación 1

## Tema 1

---

Problemas de tratamiento de  
información, algoritmos y programas



Escuela de  
Ingeniería y Arquitectura  
**Universidad** Zaragoza

# Problemas, algoritmos y programas

---

- Problemas de **tratamiento de información**
  - Objetivo: resolución **automática** del problema
    - ¿Quién lo resolverá realmente?: Un computador
  - Necesidad de **programarlo**
    - A través de **algoritmos**

# Algoritmo

---

- Conjunto de operaciones
    - ordenado,
    - finito,
    - carente de ambigüedades,
- que permite hallar la solución de un problema [de tratamiento de información]

# Receta de tortilla de patata

## □ Ingredientes para 4 comensales □

- 4 huevos
- Medio kilo de patatas
- Media cebolla
- Aceite de oliva
- Sal



## Elaboración:

- Corte las patatas en trocitos bien finos. Ponga a calentar abundante aceite de oliva en la sartén. Ponga las patatas en la sartén cuando el aceite esté bien caliente (nunca debe humear). Añada un poco de sal. Si la quiere con cebolla, añada la cebolla picada. Cuando las patatas estén bien doraditas, sáquelas y escúrralas. Bata bien los huevos, con una pizca de sal. Añada las patatas ya fritas y mezcle bien. Retire el aceite sobrante de la sartén y vuelva a ponerla al fuego. Cuando la sartén esté bien caliente, eche la mezcla de huevo y patatas. Cuando ya está hecha o cuajada por debajo, darle la vuelta con un plato plano o una tapadera.

# Índice

---

- ❑ Problemas de tratamiento de información
- ❑ Algoritmos y programas
- ❑ Nuestro modelo de computador
- ❑ Ejemplos de programas
- ❑ Propiedades de un algoritmo

# Ejemplos de problemas de tratamiento de información

---

- ❑ Facilitar la escritura, edición, impresión y preservación digital de un texto
- ❑ Gestionar la información académica de los alumnos de la Universidad de Zaragoza
- ❑ Averiguar el número primo que sigue a 104743
- ❑ Permitir que una o varias personas jueguen en un entorno virtual persiguiendo un determinado objetivo
- ❑ Guiar el rayo láser que realiza queratectomía fotorrefractiva para corregir la miopía en ojos humanos
- ❑ Permitir que varias personas compartan entre sí en Internet información personal como noticias, fotografías, videos, etc.

# Problemas, algoritmos y programas

---

- **Problema** (de tratamiento de información)



- **Método para su resolución**



- **Algoritmo**



- **Programa**

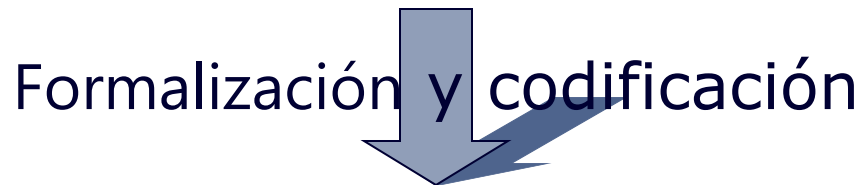
# Problemas, algoritmos y programas

---

- **Problema** (de tratamiento de información)



- **Método para su resolución**



- **Programa**

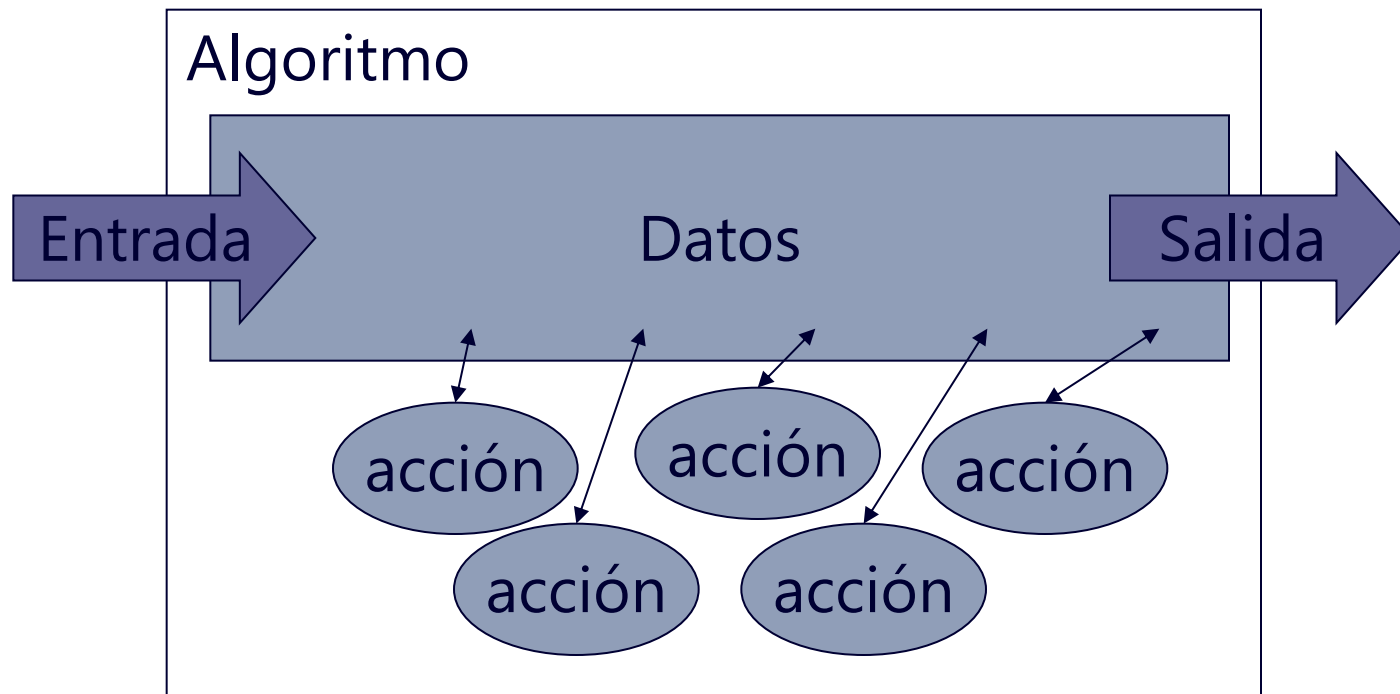


# Algoritmo

---

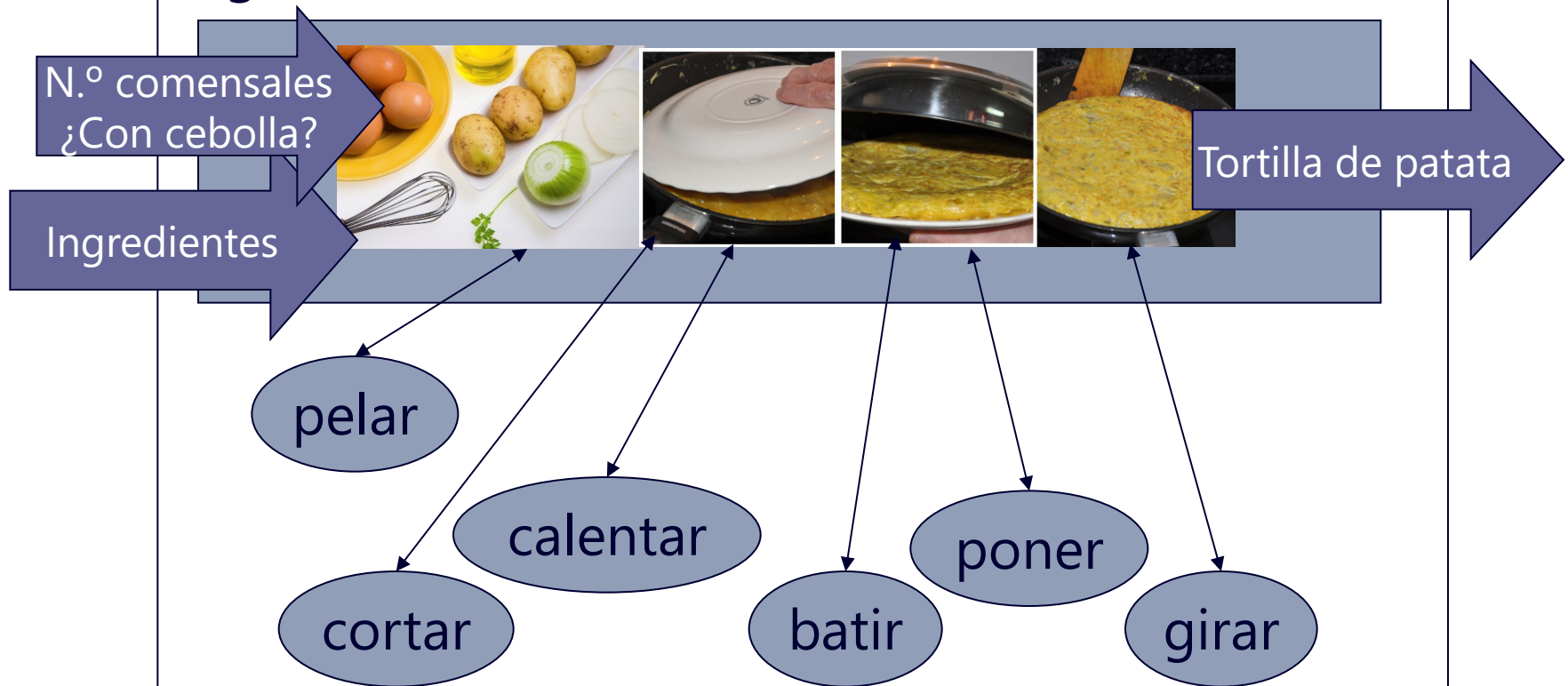
- Conjunto ordenado y finito de operaciones, carente de ambigüedades, que permite hallar la solución de un problema de tratamiento de información
- Consta de
  - **Descripción de la información** asociada al problema
  - **Descripción del modo de tratamiento** de esta información.

# Esquema de algoritmo



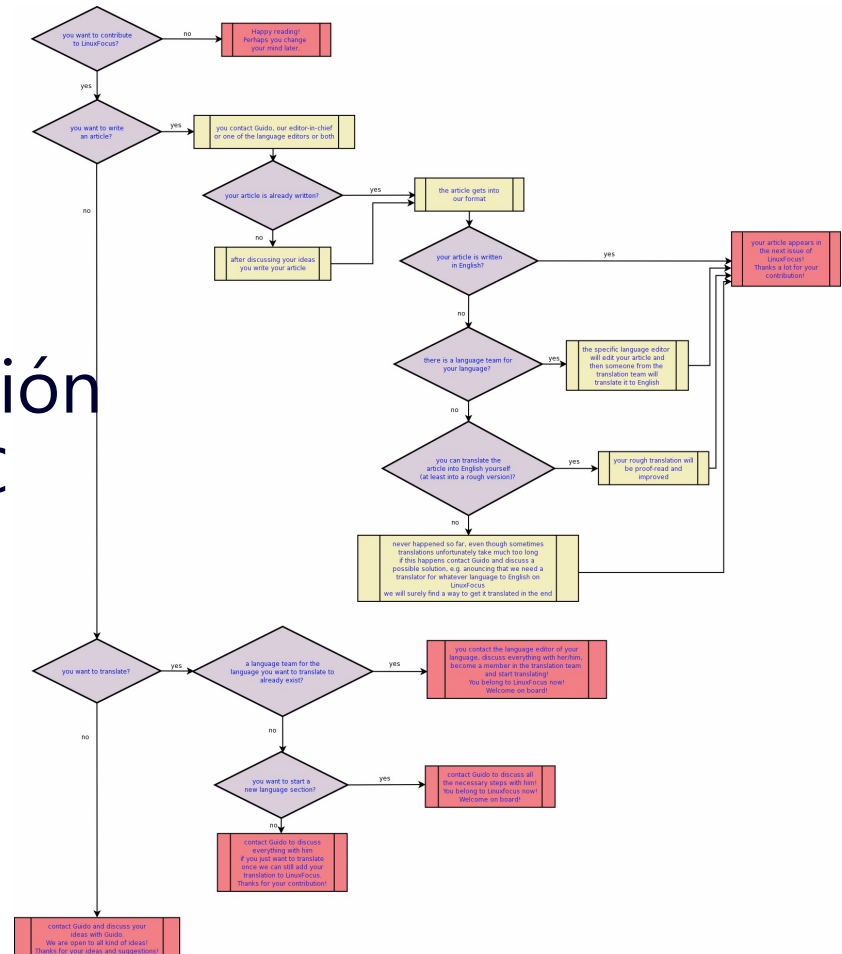
# Algoritmo para cocinar una tortilla de patata

## Algoritmo tortillaDePatata



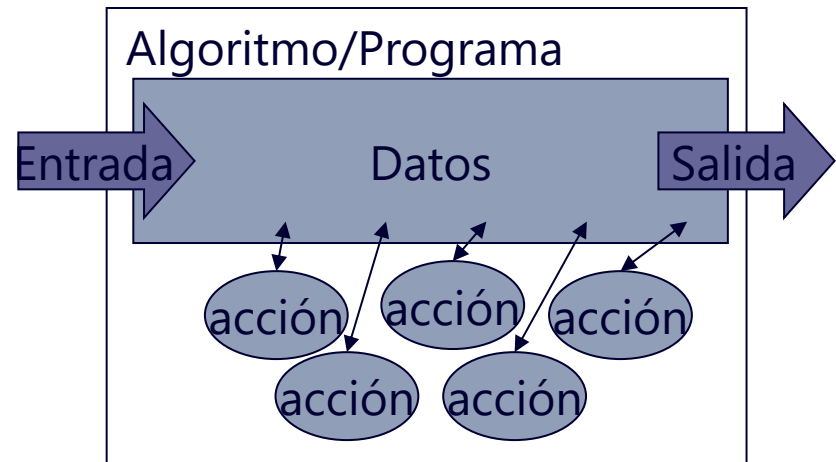
# Expresión de un algoritmo

- Lenguaje natural
- Notación algorítmica
- Notación gráfica
  - Diagramas de flujo
- Lenguaje de programación
  - Ada, Pascal, Módulo-2, C
  - **C++**, Java
  - Python, Lisp, Prolog
  - Fortran, Cobol

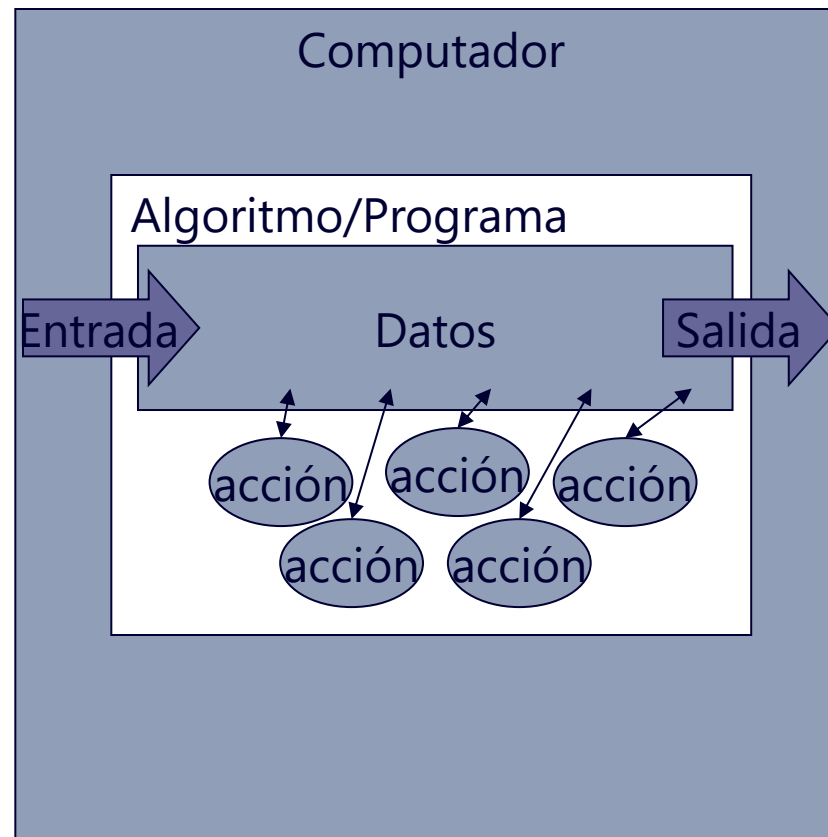


# Nuestro modelo de computador

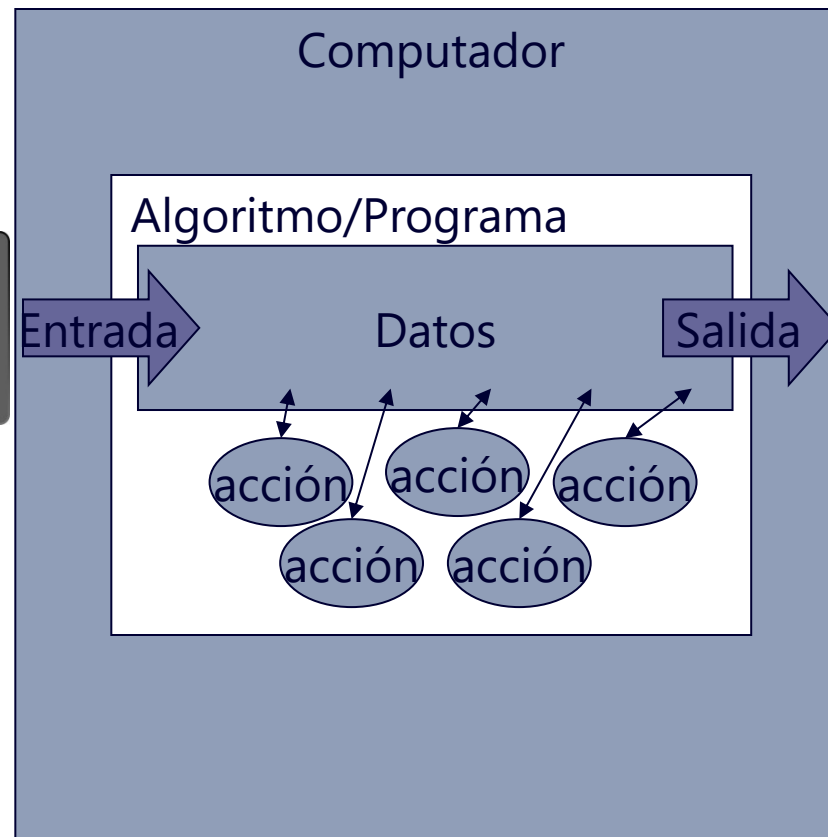
Computador



# Nuestro modelo de computador



# Nuestro modelo de computador





# Conjunto de operaciones

---

- ❑ **Realizar una operación aritmética** entre datos numéricos
- ❑ **Dar valor** a un dato
- ❑ **Modificar el valor** de un dato
- ❑ **Invocar** a otro algoritmo
- ❑ **Ejecutar condicionalmente** una acción
- ❑ **Repetir la ejecución** de una acción
- ❑ **Escribir** un dato en la pantalla
- ❑ **Leer** un dato del teclado
- ❑ ...





# Un primer programa en varios lenguajes

---





# Un programa en Python

```
# Acción a ejecutar:  
print ('Bienvenidos a UNIZAR')
```

# Un programa en Ada

```
with ada.text_IO;  
  
procedure bienvenida is  
  -- Programa que escribe en la pantalla una línea  
  -- de texto con un mensaje de bienvenida  
  
begin  
  -- Acciones que ejecutará el programa cada vez  
  -- que sea invocado  
  ada.text_IO.put("Bienvenidos a UNIZAR");  
  ada.text_IO.new_line;  
end bienvenida;
```

# Un primer programa en C++

```
#include <iostream>

/*
 * Programa que escribe en la pantalla el mensaje
 * "Bienvenidos a la Universidad"
 */
int main() {
    // una única instrucción:
    std::cout << "Bienvenidos a la Universidad" << std::endl;
}
```

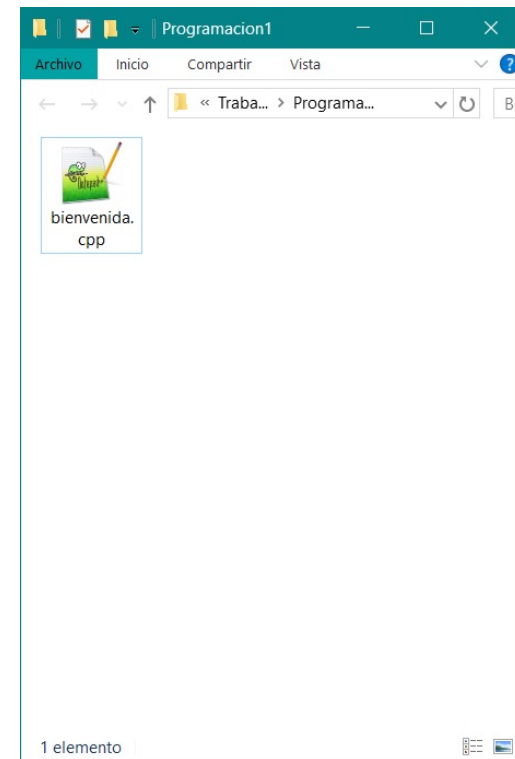
- Edición del código fuente





# ¿Cómo se ejecuta el código C++?

- Edición del código fuente



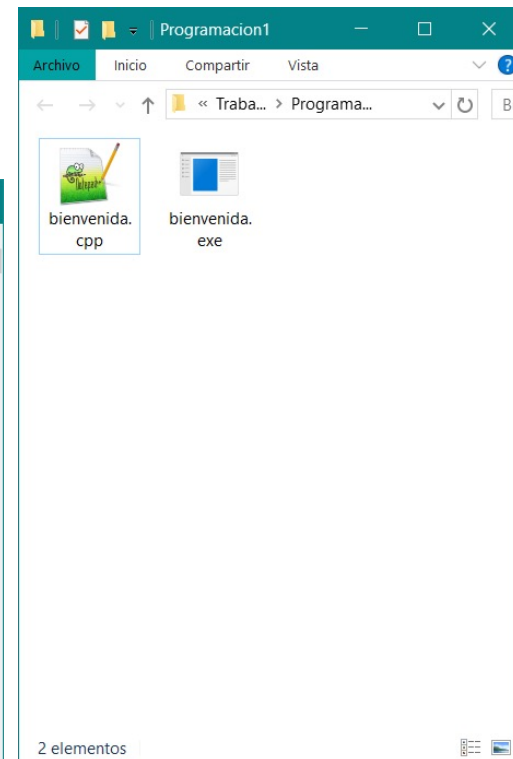
# ¿Cómo se ejecuta el código C++?

- ❑ Edición del código fuente
- ❑ Compilación del código fuente

```
C:\Windows\System32\cmd.exe

E:\Miguel\Documentos\Trabajo\Programacion1>g++ -o bienvenida.exe bienvenida.cpp

E:\Miguel\Documentos\Trabajo\Programacion1>
```



# ¿Cómo se ejecuta el código C++?

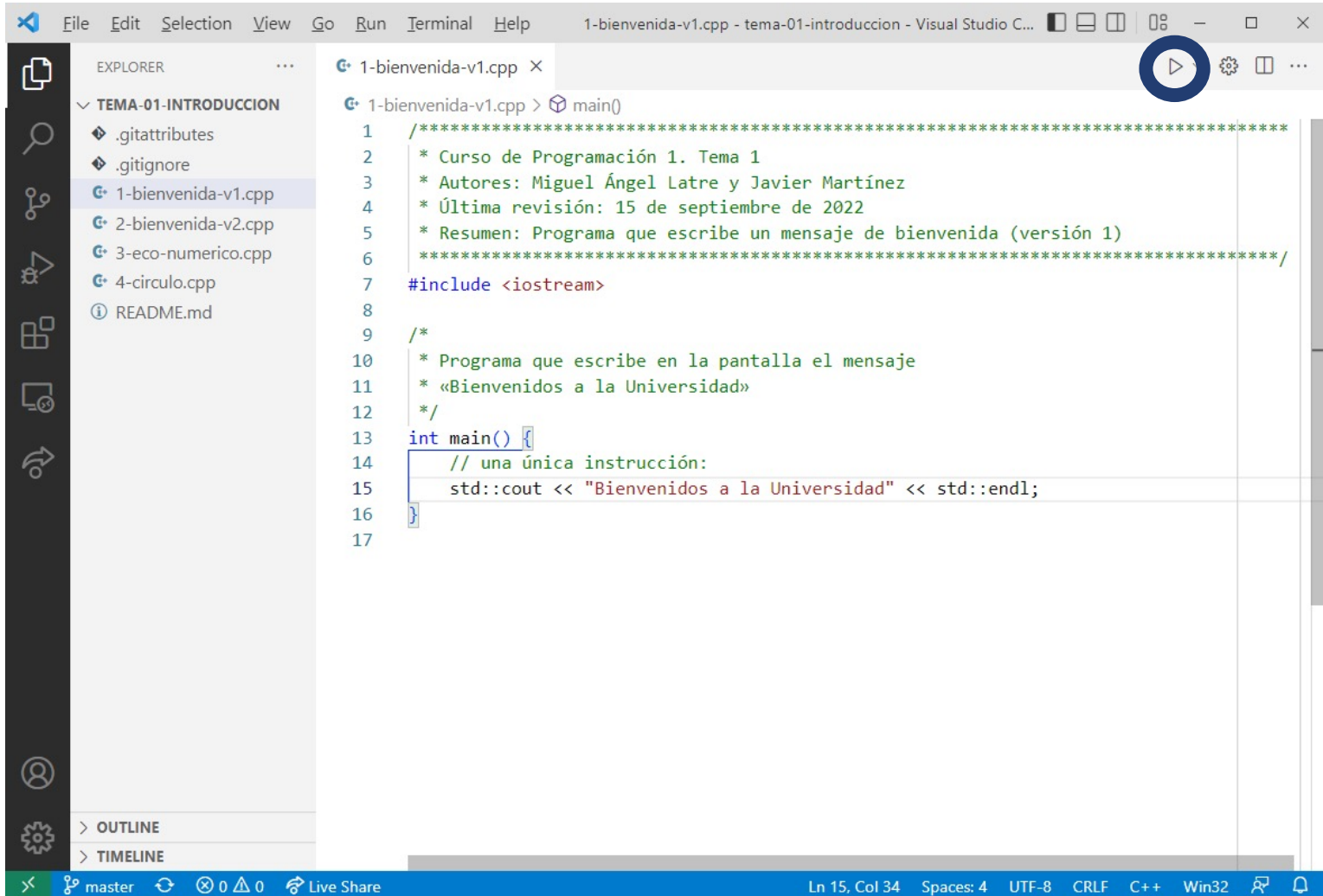
---

- ❑ Edición del código fuente
- ❑ Compilación del código fuente
- ❑ Ejecución del código ejecutable

```
C:\Windows\System32\cmd.exe
E:\Miguel\Documentos\Trabajo\Programacion1>g++ -o bienvenida.exe bienvenida.cpp
E:\Miguel\Documentos\Trabajo\Programacion1>bienvenida
Bienvenidos a la Universidad
E:\Miguel\Documentos\Trabajo\Programacion1>_
```



# Compilación y ejecución en Visual Studio Code



```
1-bienvenida-v1.cpp > main()
1  /*****
2   * Curso de Programación 1. Tema 1
3   * Autores: Miguel Ángel Latre y Javier Martínez
4   * Última revisión: 15 de septiembre de 2022
5   * Resumen: Programa que escribe un mensaje de bienvenida (versión 1)
6   *****/
7  #include <iostream>
8
9  /*
10 * Programa que escribe en la pantalla el mensaje
11 * «Bienvenidos a la Universidad»
12 */
13 int main() {
14     // una única instrucción:
15     std::cout << "Bienvenidos a la Universidad" << std::endl;
16 }
17
```



# Compilación y ejecución en Visual Studio Code

The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays the file structure for 'TEMA-01-INTRODUCCION', including files like .gitattributes, .gitignore, 1-bienvenida-v1.cpp, 1-bienvenida-v1.exe, 2-bienvenida-v2.cpp, 3-eco-numerico.cpp, 4-circulo.cpp, and README.md. The main editor window shows the code for 1-bienvenida-v1.cpp, which includes a main function that prints 'Bienvenidos a la Universidad'. The Terminal panel at the bottom shows the output of the program, which is 'Bienvenidos a la Universidad'. A yellow arrow points from the terminal output to the '1-bienvenida-v1.exe' file in the Explorer panel.

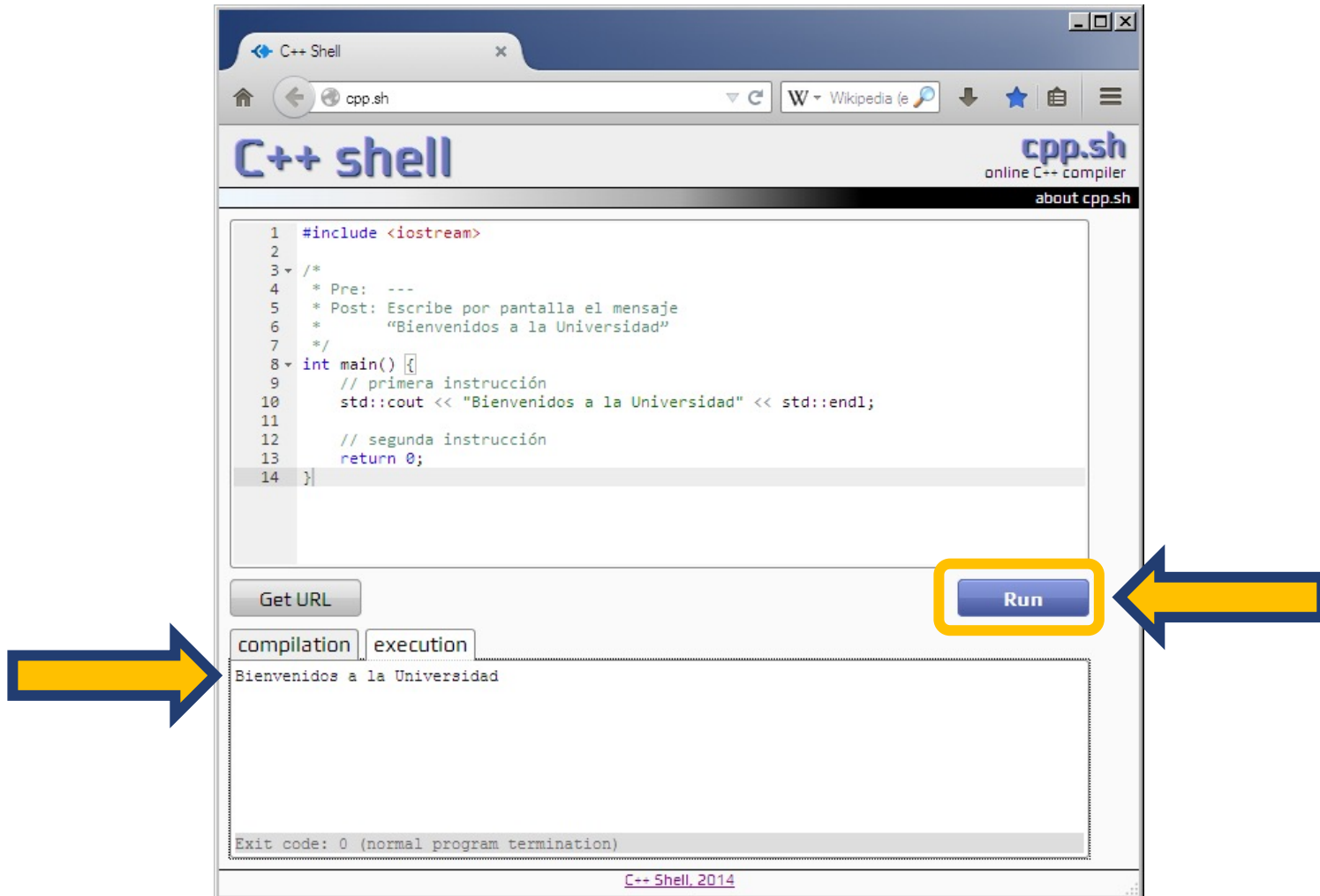
```
1-bienvenida-v1.cpp > main()
1  /*****
2   * Curso de Programación 1. Tema 1
3   * Autores: Miguel Ángel Latre y Javier Martínez
4   * Última revisión: 15 de septiembre de 2022
5   * Resumen: Programa que escribe un mensaje de bienvenida (versión 1)
6   *****/
7  #include <iostream>
8
9  /*
10 * Programa que escribe en la pantalla el mensaje
11 * «Bienvenidos a la Universidad»
12 */
13 int main() {
14     // una única instrucción:
15     std::cout << "Bienvenidos a la Universidad" << std::endl;
16 }
17
```

Windows PowerShell  
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma <https://aka.ms/pscore6>

```
PS C:\Users\Latre\Documents\Trabajo\Docencia\prog1f\codigo-c++-publico\tema-01-introduccion> cd
"c:\Users\Latre\Documents\Trabajo\Docencia\prog1f\codigo-c++-publico\tema-01-introduccion";
if ($?) { g++ 1-bienvenida-v1.cpp -o 1-bienvenida-v1 }; if ($?) { .\1-bienvenida-v1 }
Bienvenidos a la Universidad
PS C:\Users\Latre\Documents\Trabajo\Docencia\prog1f\codigo-c++-publico\tema-01-introduccion>
```

# Compilación y ejecución en cpp.sh



The screenshot shows the cpp.sh online C++ compiler interface. The browser window has a single tab titled "C++ Shell" and the address bar shows "cpp.sh". The page header includes the "C++ shell" logo and the text "online C++ compiler" and "about cpp.sh". The main area contains a code editor with the following C++ code:

```
1 #include <iostream>
2
3 /*
4  * Pre: ---
5  * Post: Escribe por pantalla el mensaje
6  *       "Bienvenidos a la Universidad"
7  */
8 int main() {
9     // primera instrucción
10    std::cout << "Bienvenidos a la Universidad" << std::endl;
11
12    // segunda instrucción
13    return 0;
14 }
```

Below the code editor, there are two buttons: "Get URL" and "Run". The "Run" button is highlighted with a yellow rectangle and a yellow arrow points to it from the right. Below these buttons are two tabs: "compilation" and "execution". The "compilation" tab is selected, and a yellow arrow points to it from the left. The output area shows the text "Bienvenidos a la Universidad" and "Exit code: 0 (normal program termination)". At the bottom of the page, it says "C++ Shell, 2014".

# Un primer programa en C++

```
#include <iostream>

/*
 * Programa que escribe en la pantalla el mensaje
 * "Bienvenidos a La Universidad"
 */
int main() {
    // una única instrucción:
    std::cout << "Bienvenidos a la Universidad" << std::endl;
}
```

# Un primer programa en C++

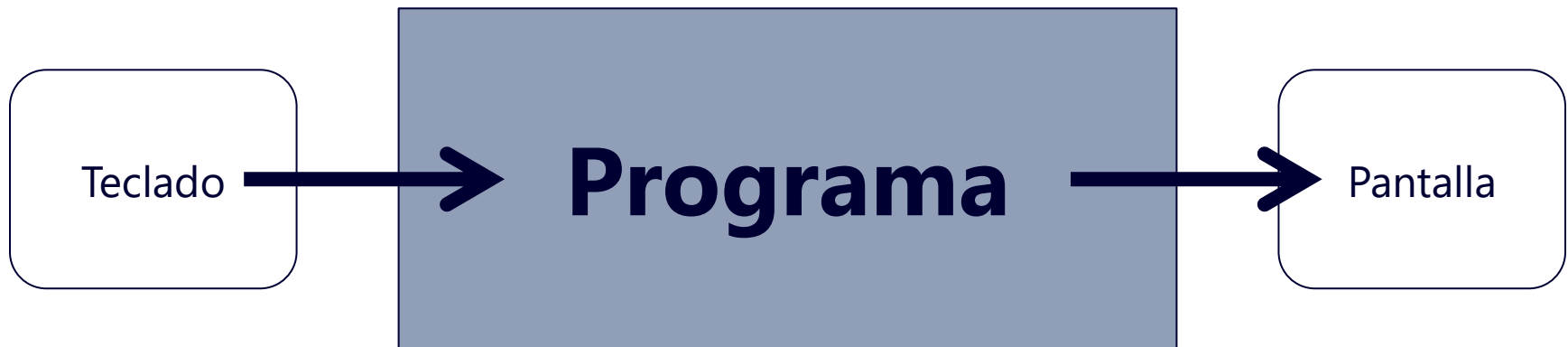
```
#include <iostream>
using namespace std;

/*
 * Programa que escribe en la pantalla el mensaje
 * "Bienvenidos a La Universidad"
 */
int main() {
    // una única instrucción:
    cout << "Bienvenidos a la Universidad" << endl;
}
```



# Un programa interactivo que lee del teclado

---





# Un programa interactivo que lee del teclado

```
#include <iostream>
using namespace std;

/*
 * Programa que solicita un número entero al usuario y lo
 * escribe en pantalla.
 */
int main() {
    cout << "Escriba un número entero: ";

    int numero;
    cin >> numero;

    cout << "El número escrito es el " << numero << endl;
}
```



# Un programa interactivo que lee del teclado. Ejecución

Escriba un número entero:





# Un programa interactivo que lee del teclado. Ejecución

Escriba un número entero: 2019



# Un programa interactivo que lee del teclado. Ejecución

Escriba un número entero: 2019↵



# Un programa interactivo que lee del teclado. Ejecución

Escriba un número entero: 2019

El número escrito es el 2019



# Programa interactivo que hace un cálculo

---

- Programa que solicita al usuario la longitud de un radio y escriba en la pantalla el área del círculo correspondiente.



# Programa interactivo que hace un cálculo

```
#include <iostream>
using namespace std;
const double PI = 3.14159265358979323846;
/*
 * Solicita al usuario la longitud de un radio y escribe en la
 * pantalla el área del círculo correspondiente.
 */
int main() {
    cout << "Escriba el radio de un círculo: ";

    double r;
    cin >> r;

    cout << "El área de un círculo de radio " << r << " es "
         << PI * r * r << endl;
}
```



# Programa interactivo que hace un cálculo. Ejecución

Escriba el radio de un círculo:



# Programa interactivo que hace un cálculo. Ejecución

Escriba el radio de un círculo: 2.5

El área de un círculo de radio 2.5 es 19.635

# Propiedades de un algoritmo

---

## □ **Imprescindibles**

- Corrección
- Legibilidad

## □ **Deseables**

- Generalidad
- Reusabilidad
- Eficiencia
- Independencia de la máquina y del lenguaje
- Simplicidad
- Robustez



# Programa sintácticamente incorrecto

```
#include <iostream>

/*
 * Programa que escribe en la pantalla el mensaje
 * "Bienvenidos a La Universidad"
 */
{
    // una única instrucción:
    cout << "Bienvenidos a La Universidad" << endl;
}
```

# Programa formalmente incorrecto

```
#include <iostream>

using namespace std;

/*
 * Programa que escribe en la pantalla la suma de los
 * números del 1 al 5.
 */
int main() {
    cout << 1 + 2 + 3 + 4 << endl;
}
```

# Propiedades de un algoritmo

---

## □ **Imprescindibles**

- Corrección
- **Legibilidad**

## □ **Deseables**

- Generalidad
- Reusabilidad
- Eficiencia
- Independencia de la máquina y del lenguaje
- Simplicidad
- Robustez

# Programa C++ ilegible

```
#include <iostream>
#include <iomanip>
using namespace std; void o(double oo){
const double ooo=3.14159265358979323846;
cout<<setw(7)<<oo<<setw(26)<<2.0*ooo*oo<<
endl;}int main(){cout<<setprecision(2);cout
<<fixed;cout<<setw(7)<<"radio"<<setw(20)<<
"Circunferencia"<<endl;cout<<setw(7)<<
"===="<<setw(20)<<"===== "<<endl;o
(1.234);o(5.0112);o(11.5178);cout<<endl;
return 0;}
```

# Propiedades de un algoritmo

---

## □ **Imprescindibles**

- Corrección
- Legibilidad

## □ **Deseables**

- **Generalidad**
- **Reusabilidad**
- Eficiencia
- Independencia de la máquina y del lenguaje
- Simplicidad
- Robustez

# Generalidad

```
/*  
 * Programa que escribe en la pantalla la suma de los  
 * enteros comprendidos en el intervalo [1, 100].  
 */  
int main() {  
    int resultado = 0;  
    for (int i = 1; i <= 100; i++) {  
        resultado = resultado + i;  
    }  
  
    cout << "La suma de los enteros comprendidos en el "  
        << "intervalo [1, 100] es: "  
        << resultado << endl;  
}
```

# Generalidad

```
/*  
 * Programa que escribe en la pantalla la suma de los enteros comprendidos en  
 * el intervalo un intervalo solicitado al usuario.  
 */  
int main() {  
    cout << "Escriba el valor inicial y final de un intervalo entero: ";  
    int inicio, fin;  
    cin >> inicio >> fin;  
  
    int resultado = 0;  
    for (int i = inicio; i <= fin; i++) {  
        resultado = resultado + i;  
    }  
  
    cout << "La suma de los enteros comprendidos en el intervalo [" << inicio  
        << ", " << fin << "] es: " << resultado << endl;  
}
```

# Propiedades de un algoritmo

---

## □ **Imprescindibles**

- Corrección
- Legibilidad

## □ **Deseables**

- Generalidad
- Reusabilidad
- **Eficiencia**
- Independencia de la máquina y del lenguaje
- Simplicidad
- Robustez



# Eficiencia

```
/*  
 * Programa que escribe en la pantalla la suma de los enteros comprendidos en  
 * el intervalo un intervalo solicitado al usuario.  
 */  
int main() {  
    cout << "Escriba el valor inicial y final de un intervalo entero: ";  
    int inicio, fin;  
    cin >> inicio >> fin;  
  
    int resultado = (inicio + fin) * (fin - inicio + 1) / 2;  
  
    cout << "La suma de los enteros comprendidos en el intervalo [" << inicio  
        << ", " << fin << "] es: " << resultado << endl;  
}
```

# Propiedades de un algoritmo

---

## □ **Imprescindibles**

- Corrección
- Legibilidad

## □ **Deseables**

- Generalidad
- Reusabilidad
- Eficiencia
- **Independencia** de la máquina y del lenguaje
- **Simplicidad**
- Robustez

# Propiedades de un algoritmo

---

## □ **Imprescindibles**

- Corrección
- Legibilidad

## □ **Deseables**

- Generalidad
- Reusabilidad
- Eficiencia
- Independencia de la máquina y del lenguaje
- Simplicidad
- **Robustez**

# Robustez

---

- ❑ Lamentablemente, nuestros programas van a ser poco robustos:

Escriba un entero: un entero

El entero escrito es el 0

- ❑ En Programación II, se verán los mecanismos que ofrece C++ para aumentar la robustez cuando se lee de un flujo (como el teclado).

# Resumen

---

- ❑ Problemas de tratamiento de información
- ❑ Algoritmos y programas
- ❑ Ejemplos de programas C++
- ❑ Propiedades de un algoritmo

# ¿Cómo se estudia este tema?

---

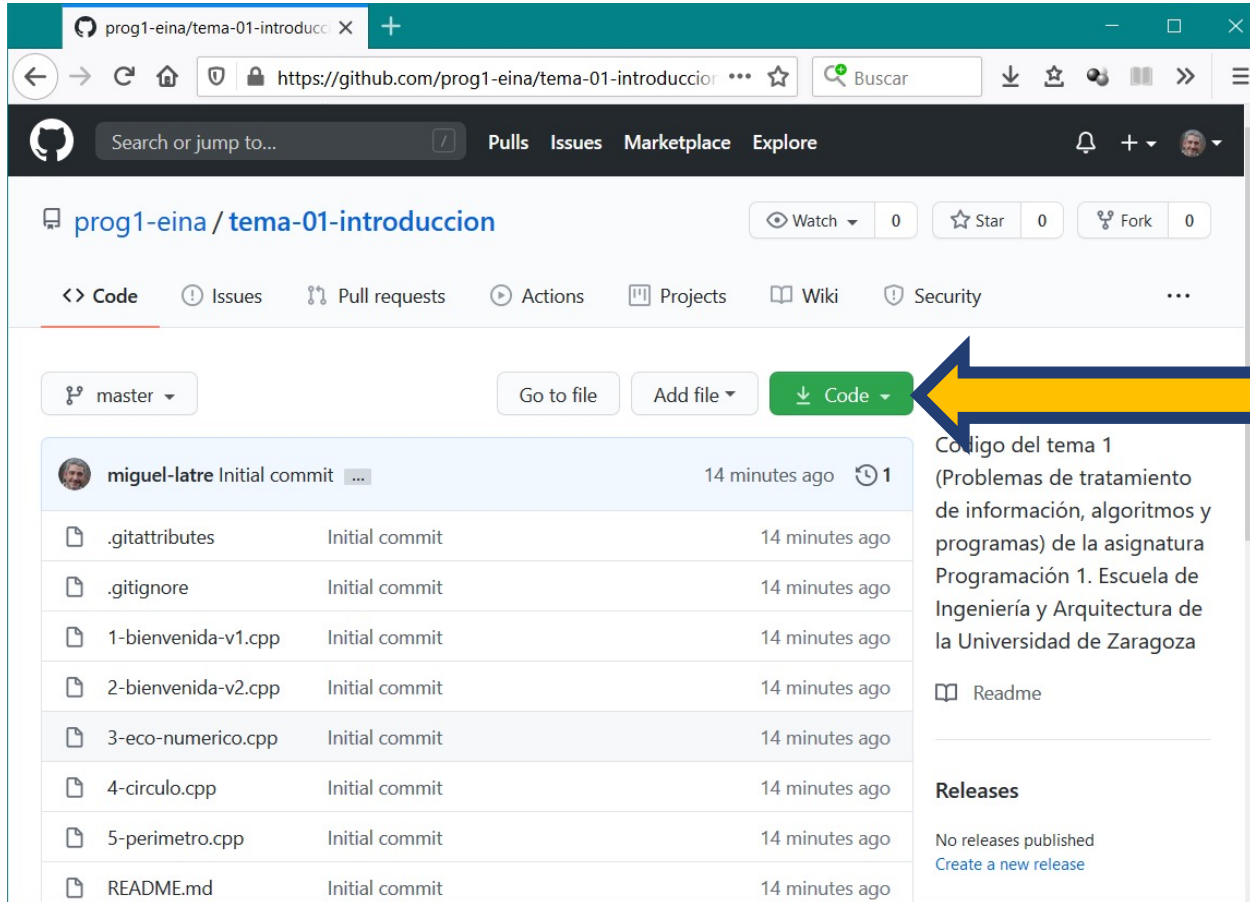
- Repasando las transparencias
- Leyendo las secciones 1.1, 1.2 y 1.4 del capítulo 1 de los apuntes del profesor Martínez, en Moodle
- Ejecutando los programas presentados en un entorno de ejecución en línea, como <http://cpp.sh/>
- Ejecutando los programas en Visual Studio Code:
  - Instalándolo según las instrucciones de «Tutorial para la instalación de Visual Studio Code» en Moodle
  - Leyendo la descripción del entorno Visual Studio Code de la 1.ª práctica de la asignatura.
    - Disponible el viernes en Moodle

# ¿Cómo se estudia este tema?

---

- Código fuente de este tema
  - Se puede copiar y pegar
  - También está disponible en el repositorio <https://github.com/prog1-eina/tema-01-introduccion>

# ¿Cómo se estudia este tema?



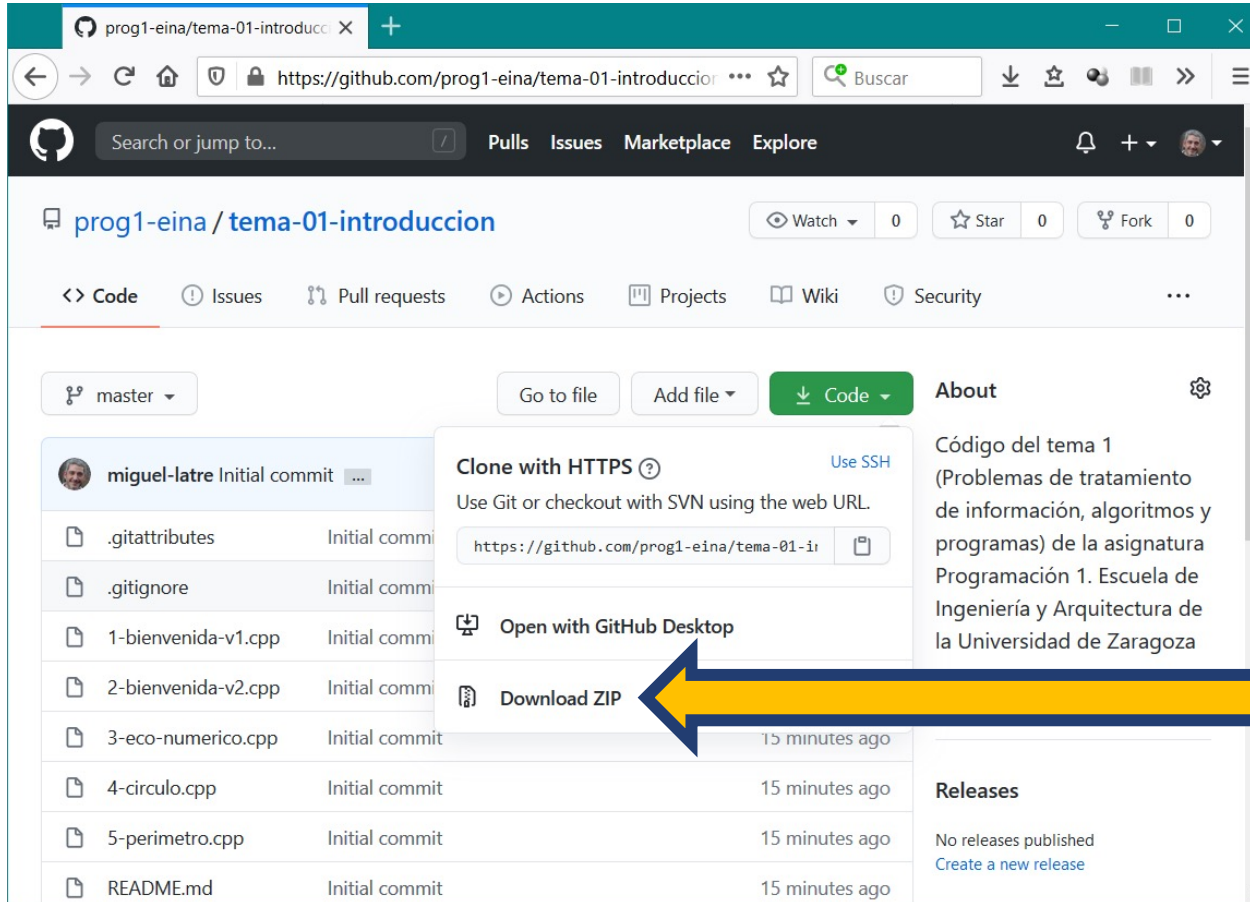
The screenshot shows a web browser displaying the GitHub repository page for 'prog1-eina/tema-01-introduccion'. The repository is owned by 'prog1-eina' and has 0 stars, 0 forks, and 0 watchers. The 'Code' button is highlighted with a large yellow arrow. Below the repository name, there is a table of files and their commit history.

File	Commit	Time
.gitattributes	Initial commit	14 minutes ago
.gitignore	Initial commit	14 minutes ago
1-bienvenida-v1.cpp	Initial commit	14 minutes ago
2-bienvenida-v2.cpp	Initial commit	14 minutes ago
3-eco-numerico.cpp	Initial commit	14 minutes ago
4-circulo.cpp	Initial commit	14 minutes ago
5-perimetro.cpp	Initial commit	14 minutes ago
README.md	Initial commit	14 minutes ago

On the right side of the repository page, there is a description of the repository: 'Codigo del tema 1 (Problemas de tratamiento de información, algoritmos y programas) de la asignatura Programación 1. Escuela de Ingeniería y Arquitectura de la Universidad de Zaragoza'. Below this, there is a 'Readme' section and a 'Releases' section with the text 'No releases published' and a link to 'Create a new release'.



# ¿Cómo se estudia este tema?



The screenshot shows a web browser displaying the GitHub repository page for 'prog1-eina / tema-01-introduccion'. The repository is owned by 'miguel-latre' and contains several files, including .gitattributes, .gitignore, and several .cpp files (1-bienvenida-v1.cpp, 2-bienvenida-v2.cpp, 3-eco-numerico.cpp, 4-circulo.cpp, 5-perimetro.cpp, and README.md). The 'Code' button is highlighted with a yellow arrow pointing to the 'Download ZIP' option in the dropdown menu.

Clone with HTTPS ?  
Use Git or checkout with SVN using the web URL.  
<https://github.com/prog1-eina/tema-01-introduccion>

Open with GitHub Desktop

Download ZIP

# ¿Cómo se estudia este tema?

---

- ❑ Descomprimir
- ❑ Abrir Visual Studio Code
- ❑ Menú «File» > «Open folder...»
- ❑ Buscar y seleccionar el directorio en el que se ha descomprimido