

# 01-transpas-contactoC.pdf



**Anónimo**



**Programación Para Sistemas**



**2º Grado en Ingeniería Informática**



**Escuela Técnica Superior de Ingenieros Informáticos  
Universidad Politécnica de Madrid**

# Sesión 01: Contacto C

Programación para Sistemas

---

Ángel Herranz

2021-2022

Universidad Politécnica de Madrid

- Ya estamos usando la línea de comandos (Bash)<sup>1</sup>
- *Transparencias con ejercicios* para hacer en clase
- *Hablad*<sup>2</sup> y compartir con vuestros compañeros. . .

---

<sup>1</sup>Si no tienes un Unix en casa puedes usar triqui.

<sup>2</sup>La crisis de la COVID19 ha abierto muchas opciones.

<sup>3</sup>Opcionales.

- Ya estamos usando la línea de comandos (Bash)<sup>1</sup>
- **Transparencias con ejercicios** para hacer en clase
- **Hablad**<sup>2</sup> y compartir con vuestros compañeros. . .  
... **excepto** cuando Herranz explica

---

<sup>1</sup>Si no tienes un Unix en casa puedes usar triqui.

<sup>2</sup>La crisis de la COVID19 ha abierto muchas opciones.

<sup>3</sup>Opcionales.

- Ya estamos usando la línea de comandos (Bash)<sup>1</sup>
  - **Transparencias con ejercicios** para hacer en clase
  - **Hablad**<sup>2</sup> y compartir con vuestros compañeros...
- ... **excepto** cuando Herranz explica
- Hojas de ejercicios para hacer durante la semana<sup>3</sup>...

---

<sup>1</sup>Si no tienes un Unix en casa puedes usar triqui.

<sup>2</sup>La crisis de la COVID19 ha abierto muchas opciones.

<sup>3</sup>Opcionales.

- Ya estamos usando la línea de comandos (Bash)<sup>1</sup>
  - **Transparencias con ejercicios** para hacer en clase
  - **Hablad**<sup>2</sup> y compartir con vuestros compañeros...
- ... **excepto** cuando Herranz explica
- Hojas de ejercicios para hacer durante la semana<sup>3</sup>...
  - o para empezarlas en clase si la clase va *demasiado lenta*

---

<sup>1</sup>Si no tienes un Unix en casa puedes usar triqui.

<sup>2</sup>La crisis de la COVID19 ha abierto muchas opciones.

<sup>3</sup>Opcionales.

# Hola mundo

- ¿Alguien ha escrito código C alguna vez?

---

<sup>4</sup>Un programa que escribe "Hola mundo" en la pantalla (en la salida estándar, para ser más precisos).

# Hola mundo

- ¿Alguien ha escrito código C alguna vez?
- ¿Qué haríais para escribir un “hola mundo”?<sup>4</sup>

---

<sup>4</sup>Un programa que escribe "Hola mundo" en la pantalla (en la salida estándar, para ser más precisos).





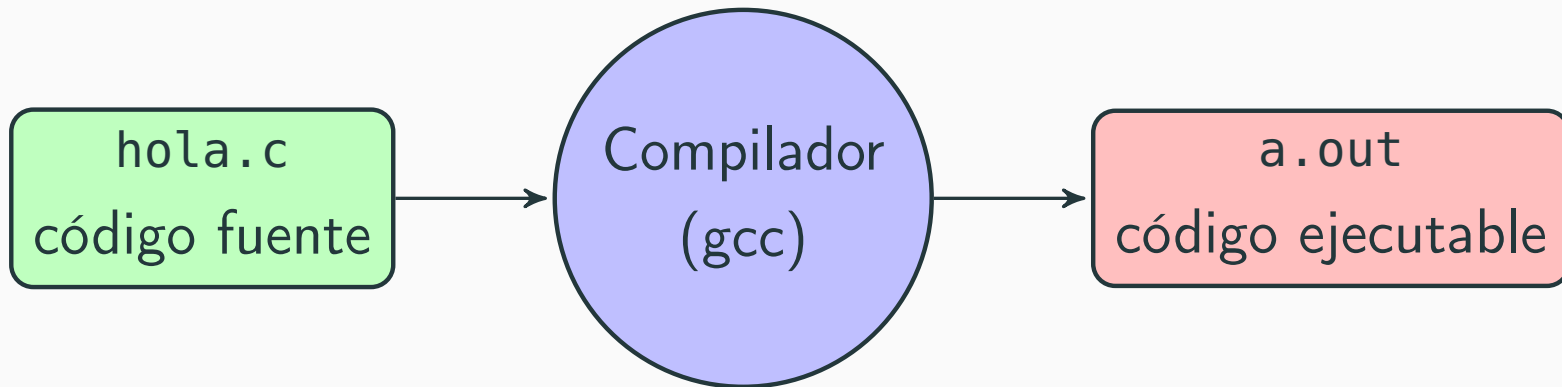
- ¿Alguien ha escrito código C alguna vez?
- ¿Qué haríais para escribir un “hola mundo”?<sup>4</sup>

---

<sup>4</sup>Un programa que escribe "Hola mundo" en la pantalla (en la salida estándar, para ser más precisos).

# Hola mundo

- ¿Alguien ha escrito código C alguna vez?
- ¿Qué haríais para escribir un “hola mundo”?<sup>4</sup>
- C es un lenguaje **compilado** (necesitamos un compilador)



---

<sup>4</sup>Un programa que escribe "Hola mundo" en la pantalla (en la salida estándar, para ser más precisos).

# Leyenda (*be formal my friend*)

*artefacto*  
(fichero)  
desarrollado  
por nosotros

*artefacto*  
(fichero/memoria)  
temporal

producto  
(fichero)  
final

*artefacto*  
(fichero/biblioteca)  
disponible  
en el sistema

proceso  
(programa en  
ejecución)

# ¿Compilado? ¿Y cómo es...

- ... Java?
- ... Bash?
- ... Python?
- ... Haskell?
- ... Scala?
- ... Javascript?
- ... C#?

# Necesitas un compilador de C

- Asumimos que vais a tener disponible una máquina Unix:
  1. Instala Linux en tu máquina
  2. Usa triqui (necesitas VPN y un cliente de ssh<sup>5</sup> para conectar)
- Compilador de C:

GCC, the GNU Compiler Collection

- Tendrás que instalarlo en tu máquina (en Ubuntu lo mejor es instalar el paquete `build-essential`)
- Una vez instalado el compilador es `gcc` (normalmente programa `/usr/bin/gcc`)

---


<sup>5</sup>openssh disponible en Unix o PuTTY en Windows

# *Install party*




# hola.c

```
#include <stdio.h>
int main()
{
    printf("Hola mundo
");
    return 0;
}
```

-  Crear un directorio `clases_pps`, en él un fichero `hola.c` con tu editor favorito, compilar y ejecutar.

# hola.c

```
#include <stdio.h>
int main()
{
    printf("Hola mundo
");
    return 0;
}
```

 Crear un directorio clases\_pps, en él un fichero hola.c con tu editor favorito, compilar y ejecutar.

```
$ mkdir clases_pps
$ cd clases_pps
$ nano hola.c
$ gcc hola.c
$ ls -l
$ ./a.out
```



# ¿a.out?

- El nombre del ejecutable `a.out` es un capricho de los desarrolladores de gcc
- Vamos a darle un nombre más digno: `hola`
- ¿Cómo?  


---

“Los programas tienen argumentos”    “gcc es un programa”  
“gcc tiene argumentos”

# ¿a.out?

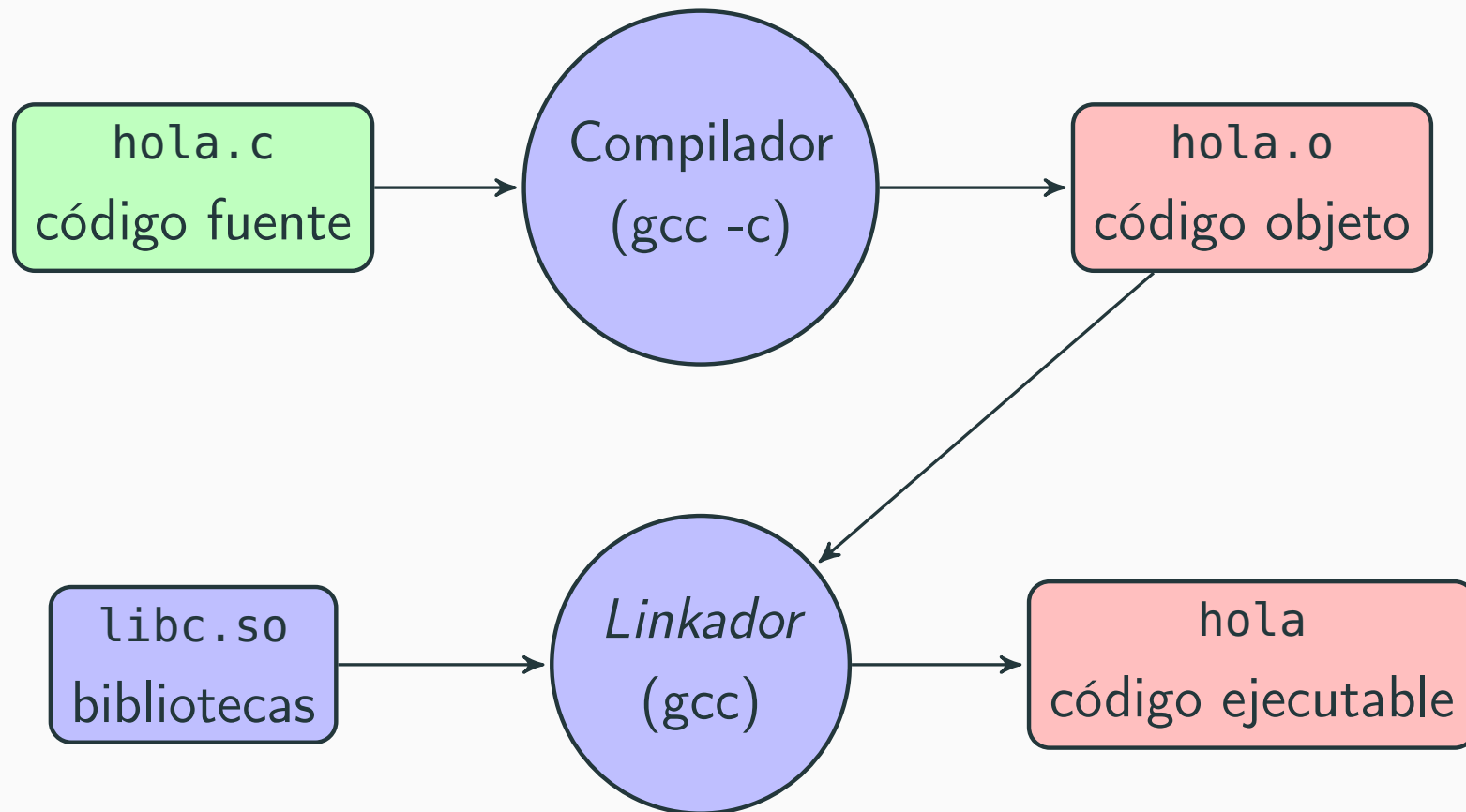
- El nombre del ejecutable `a.out` es un **capricho** de los desarrolladores de gcc
  - Vamos a darle un nombre más digno: `hola`
  - ¿Cómo?
- “Los programas tienen argumentos”    “gcc es un programa”
- 

“gcc tiene **argumentos**”

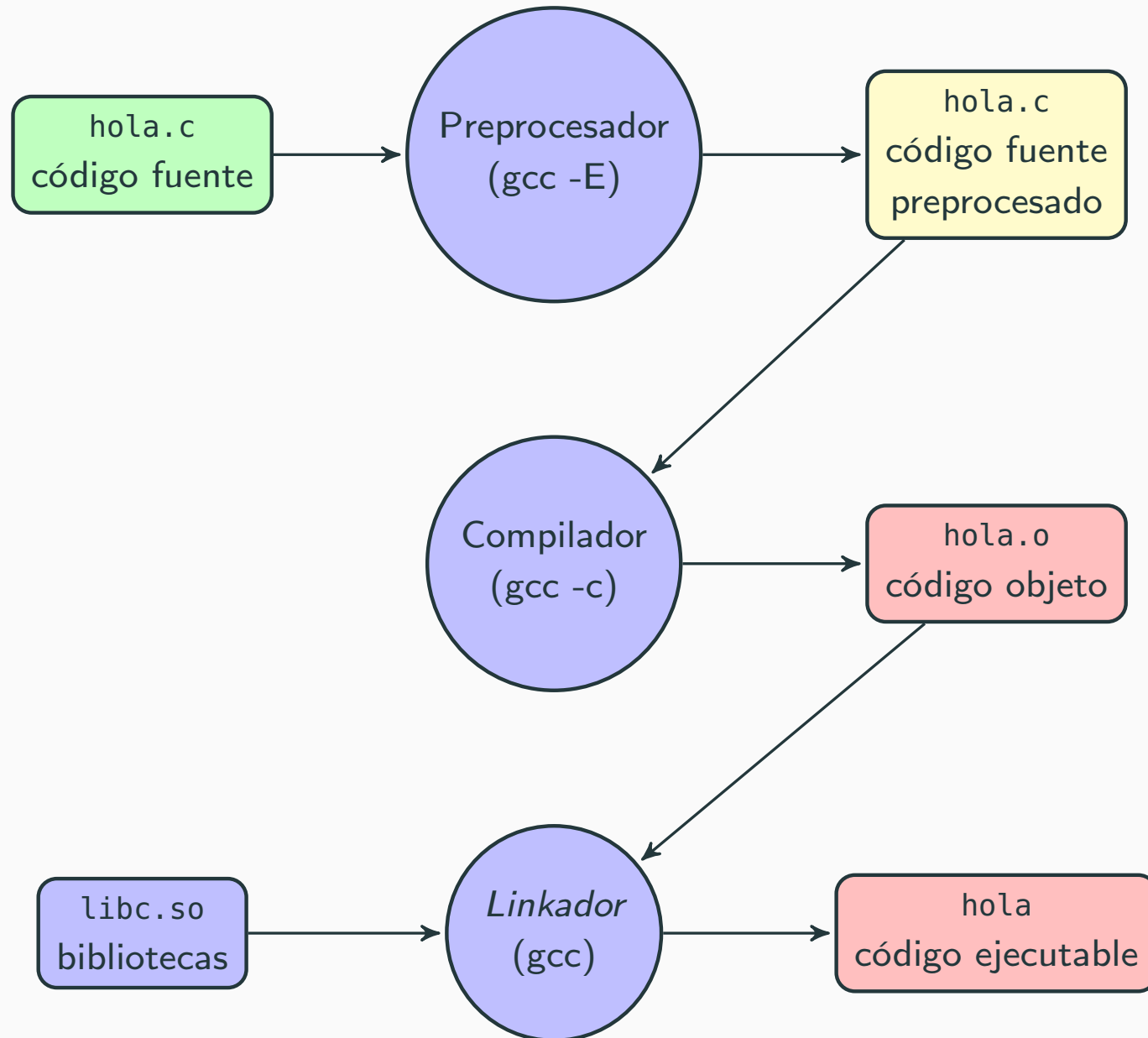
 gcc **-o hola** hola.c

**-o file**    *Place output in file file.  
This applies to whatever  
sort of output is being  
produced, [...] man gcc*

# Proceso: compilación + *linkado*



# Más detalle: preprocesado previo (**include**)



# Hola ángel

- Comunicándonos con nuestro programa: argumentos desde la línea de comandos

# Hola ángel

- Comunicándonos con nuestro programa: argumentos desde la línea de comandos
- 🏠 Escribir un programa en C (`hola.c`) que saque por la salida estándar "Hola NOMBRE", donde NOMBRE va a ser un argumento de la línea de comandos. Es decir, el programa tiene que hacer esto:

```
$ ./hola ángel
```

```
Hola ángel
```

Si el programa se invoca sin argumentos:

```
$ ./hola
```

```
Hola mundo
```

# Hola ángel (solución)

# Hola ángel (solución)

En la siguiente clase ; )



# Cómprate este libro

