



# El Compilador GNU para C: GCC

PROGRAMACIÓN I

OTOÑO DE 2022

NRC: 10844

DR. MARIO ROSSAINZ LÓPEZ

# El Compilador GCC

- ▶ **GCC** es un compilador originario del proyecto **GNU** que se utiliza para compilar programas escritos en C, C++, Objective C y Fortran.
- ▶ Es capaz de recibir un programa fuente en cualquiera de estos lenguajes y generar un programa ejecutable (binario) en el lenguaje de la máquina donde ha de ejecutarse.
- ▶ GNU = GNU's Not Unix (GNU no es UNIX)
- ▶ GCC = GNU Compiler Collection
- ▶ Una variante de GCC es G++ utilizado para compilar programas escritos en C++



# Sintaxis de uso de GCC en línea de comandos (Consola)

Para programas en lenguaje C:

```
gcc [ -opción [argumento(s)_opción]] nombre_archivo
```

Para programas en lenguaje C++:

```
g++ [ -opción [argumento(s)_opción]] nombre_archivo
```

Las opciones en la sintaxis van precedidas de un guión, como es habitual en UNIX y se conforman de una o más letras, sin embargo, no pueden agruparse varias opciones tras un mismo guión. Algunas opciones requieren de argumentos (que en su mayoría son nombres de archivos o directorios) y otras no.

Al final de la sintaxis, se pueden dar varios nombres de archivo a incluir en el proceso de compilación.

# Opciones más importantes del compilador GCC en su uso

## ▶ **-c**

Realiza solamente el preprocesamiento y la compilación de los ficheros fuentes. No se lleva a cabo la etapa de enlazado.

## ▶ **-o *archivo\_salida***

especifica el nombre del archivo de salida, resultado de la tarea solicitada al compilador.

## ▶ **-I *path***

Especifica el directorio donde se encuentran los archivos a incluir por la directiva **#include**. Se puede utilizar esta opción varias veces para especificar distintos directorios (carpetas).

## ▶ **-L *path***

Especifica al enlazador el directorio donde se encuentran los archivos de biblioteca que se habrán de incluir en la compilación. Se puede utilizar la opción -L varias veces para especificar distintos directorios de biblioteca.

# Opciones más importantes del compilador GCC en su uso

- ▶ **-W all**

Muestra todos los mensajes de advertencia del compilador

- ▶ **-g**

Incluye en el ejecutable la información necesaria para poder trazarlo empleando un depurador.

- ▶ **-v**

Muestra con detalle en stderr las órdenes ejecutadas por gcc.

# Extensiones más importantes en archivos de programas para gcc

|                           |  |
|---------------------------|--|
| .c                        | fuentes en C                             |
| .C .cc .cpp .c++ .cp .cxx | fuentes en C++; se recomienda .cpp       |
| .m                        | fuentes en Objective-C                   |
| .i                        | C preprocesado                           |
| .ii                       | C++ preprocesado                         |
| .s                        | fuentes en lenguaje ensamblador          |
| .o                        | código objeto                            |
| .h                        | archivo para preprocesador (encabezados) |

# Etapas de Compilación en GCC

- ▶ **1. PRE-PROCESADO:** En esta etapa directivas son interpretadas por el preprocesador. Las variables inicializadas con `#define` se sustituyen en el código por su valor en todos los lugares donde aparece su nombre.
- ▶ **2. COMPILACIÓN:** La compilación transforma el código C o C++ en el lenguaje ensamblador propio del procesador de nuestra máquina.
- ▶ **3. ENSAMBLADO:** El ensamblado transforma el programa escrito en lenguaje ensamblador a código objeto, que es un archivo binario en lenguaje de máquina ejecutable por el procesador.
- ▶ **4. ENLAZADO:** Las funciones de C/C++ incluidas en el código fuente, tal como `printf()`, se encuentran ya compiladas y ensambladas en bibliotecas existentes en el sistema. Es preciso incorporar de algún modo el código binario de estas funciones a nuestro ejecutable. En esto consiste la etapa de enlace o ligado, donde se reúnen uno o más módulos en código objeto con el código existente en las bibliotecas, en este caso del lenguaje C/C++.

# Todo en un solo paso

Los cuatro pasos definidos de la etapa de compilación se pueden llevar a cabo en un solo paso con el uso del gcc o g++ según corresponda.

- ▶ **gcc hola.c**

compila el programa en C hola.c, generando un archivo ejecutable a.out (a.exe)

- ▶ **gcc -o hola hola.c**

compila el programa en C hola.c, generando un archivo ejecutable hola.exe

- ▶ **gcc -c hola.c**

no genera el ejecutable, sino el código objeto, en el archivo hola.o. Si no se indica un nombre para el archivo objeto, usa el nombre del archivo en C y le cambia la extensión por .o

- ▶ **gcc -c -o objeto.o hola.c**

genera el código objeto indicando el nombre de archivo.