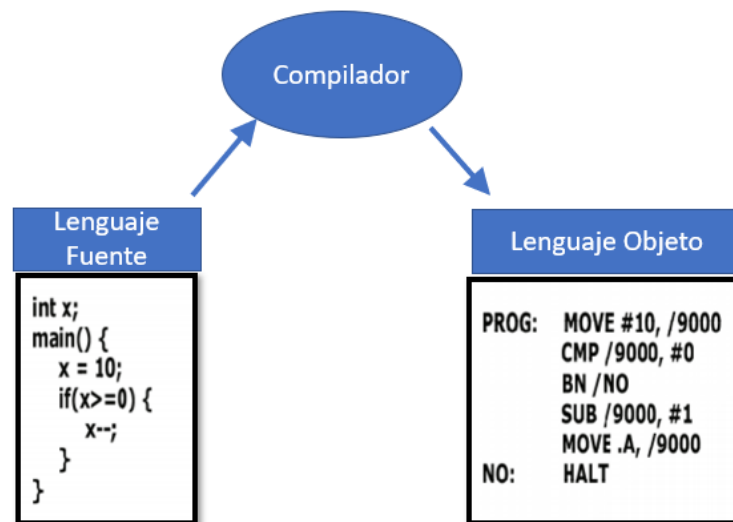


Compilador GCC

Introducción

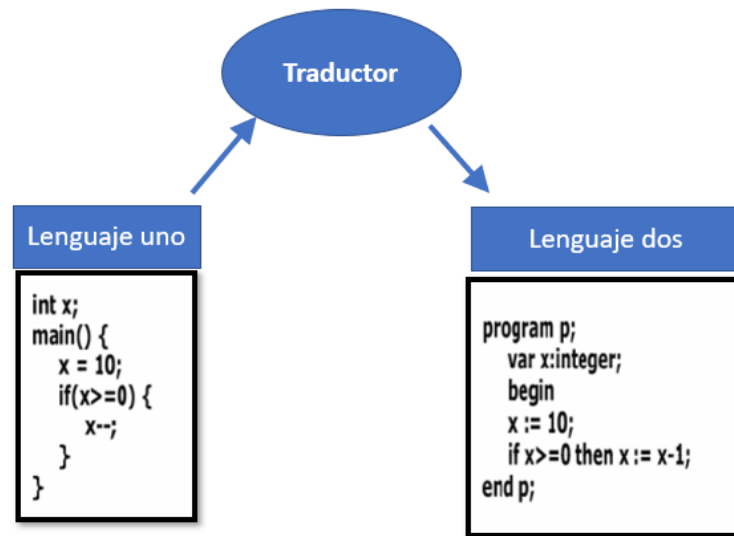
Un compilador es un tipo especial de traductor en el que el programa de origen o fuente es un lenguaje de alto nivel, y el código máquina o lenguaje objeto es de bajo nivel.

Este código máquina es almacenado en un fichero conocido como fichero ejecutable, a veces también llamado fichero binario.



Esquema de un compilador.

Un traductor es un programa que convierte el texto escrito en un lenguaje en texto escrito en otro lenguaje.



Esquema de un traductor.

La diferencia entre compilador e intérprete es que el compilador analiza todo el código fuente, crea el programa objeto o programa ejecutable. El intérprete lee sentencia por sentencia el código fuente, lo convierte en código objeto y lo ejecuta. Por lo tanto, es fácil entender que, tras compilar un programa, su ejecución es más rápida que la ejecución de un programa interpretado.

Ejemplos:

- Lenguaje interpretado: Python
- Lenguaje compilado: C ANSI



Después de leer este apunte se comprenderá cómo compilar un programa en lenguaje C y, cómo usar las opciones básicas del compilador para su optimización.

Características de GCC

Lo primero de todo: GCC es un compilador portable y puede producir salidas para muchos tipos de procesadores. También soporta microcontroladores. GCC no es sólo un compilador nativo produciendo ficheros ejecutables; está escrito en C con un fuerte enfoque hacia la portabilidad, y puede compilarse así mismo, pudiendo ser adaptado fácilmente a nuevos sistemas.

GCC tiene un diseño modular, permitiendo el soporte para nuevos lenguajes y arquitecturas.

Finalmente, y de manera más importante, GCC es software libre, distribuido bajo la GNU GPL (*General Public License*).

Programación en C

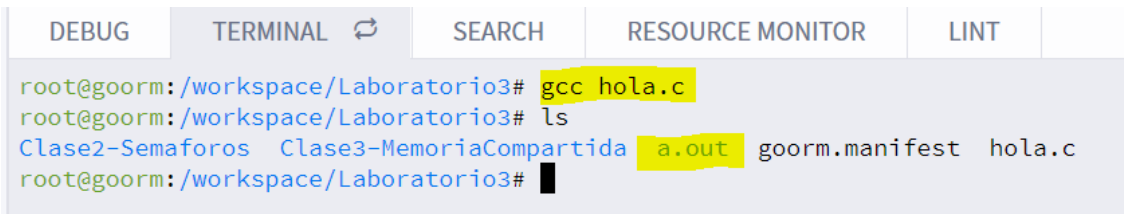
C es un lenguaje que permite el acceso directo a la memoria del ordenador. Históricamente, ha sido usado para escribir sistemas de software de bajo nivel, y aplicaciones donde el alto rendimiento o el control a través del uso de recursos es crítico. Sin embargo, se requiere de gran cuidado para asegurar que la memoria es accedida de manera correcta, para evitar inconvenientes en otras estructuras de datos en memoria.

Compilando un programa escrito en C

Para comenzar, un programa de ejemplo: el clásico “Hola Mundo”.

```
1  #include <stdio.h>
2
3  int main()
4  {
5      printf("Hola Mundo\n");
6
7      return 0;
8  }
```

Asumiendo que el nombre del archivo es “hola.c”, se escribe por consola ->gcc hola.c



The screenshot shows a terminal window with a dark background and light-colored text. At the top, there are tabs for 'DEBUG', 'TERMINAL', 'SEARCH', 'RESOURCE MONITOR', and 'LINT'. The 'TERMINAL' tab is active. The terminal output shows the following commands and results:

```
root@goorm:/workspace/Laboratorio3# gcc hola.c
root@goorm:/workspace/Laboratorio3# ls
Clase2-Semaforos  Clase3-MemoriaCompartida  a.out  goorm.manifest  hola.c
root@goorm:/workspace/Laboratorio3#
```

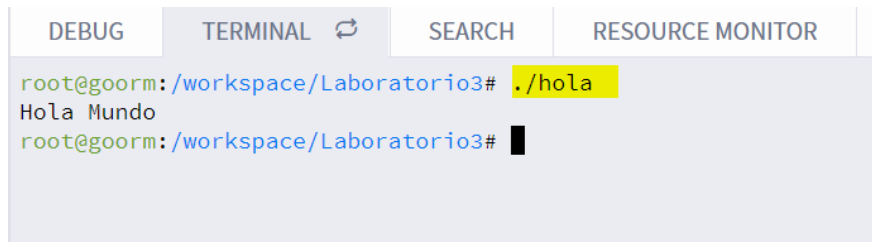
Esto genera un archivo de salida por defecto -> “a.out”. Para ejecutarlo, se escribe -> ./a.out

El “./” le indica al sistema que busque el archivo “a.out” en el directorio donde se está ubicado, si se quiere ejecutar un archivo que está en otro directorio, simplemente se debe reemplazar “./” por la ruta completa al directorio donde se encuentra el archivo.

Para indicarle al compilador que se desea renombrar el nombre del archivo de salida por un nombre personalizado, se escribe lo siguiente:

```
gcc -o hola hola.c
```

La opción “-o” indica que la palabra a continuación (en este caso, “hola”) va a ser el nombre de archivo de salida. Para ejecutar el archivo se escribe: “./hola”.

A screenshot of a terminal window with tabs for DEBUG, TERMINAL, SEARCH, and RESOURCE MONITOR. The terminal shows a prompt 'root@goorm:/workspace/Laboratorio3#' followed by the command './hola' which outputs 'Hola Mundo'.

```
root@goorm:/workspace/Laboratorio3# ./hola
Hola Mundo
root@goorm:/workspace/Laboratorio3#
```

Editores de código en C y SO (Sistemas Operativos Linux):

Editores de código

Hay una gran variedad de editores de código para el sistema operativo Linux, por ejemplo:

- *Visual Studio Code.*
- *Sublime text 3.*
- *Vi.*
- *Vim.*
- *Nano.*
- *Bluefish.*
- *GEdit.*
- Editor online: <https://www.onlinegdb.com/>

Independientemente del editor que se seleccione, la compilación de los códigos fuentes deberá realizarse desde la consola de Linux.

SO

Las distribuciones de Linux más utilizadas son:

- *Fedora.*
- *Ubuntu.*
- *Debian.*

En caso de no tener una PC con Sistema Operativo Linux, se puede utilizar una máquina virtual o algún Linux en la nube:

- *VirtualBox*.
- *VMware*.
- Linux en la nube: <https://ide.goorm.io/>