

Compilando con el GCC/DJGPP

Introducción:

Este pequeño instructivo es solamente a título informativo como para poder empezar a compilar con el GCC. Sólo abarcará lo básico, se recomienda leer la (abundante) documentación que acompaña a la aplicación para conocer los demás comandos y opciones que pueden usarse.

En primer lugar se presupone que el DJGPP está correctamente instalado según las instrucciones que figuran en `readme.1st...`

Compilando:

El programa que nos servirá para compilar se llama `gcc.exe` y se encuentra en el directorio `\DJGPP\BIN...` como se supone que la instalación ha sido correcta, no es necesario estar dentro de ese directorio dado que DOS debería encontrar las aplicaciones del DJGPP desde cualquier otra carpeta.

Supongamos que queremos compilar el código fuente `prueba.c` que tenemos en la carpeta `C:\7502\...` lo compilaremos de la siguiente manera (es importante recordar que siendo el GCC una aplicación desarrollada para UNIX distingue entre minúsculas y mayúsculas en los argumentos):

```
C:\7502\>gcc -o prueba.exe prueba.c -Wall -pedantic -ansi
```

Esto compilará nuestro código fuente `prueba.c` creando el ejecutable `prueba.exe...` ahora bien, ¿qué significa cada parámetro?...

- `-o:` (output) Indica que el siguiente argumento será el nombre con el cual queremos que se compile nuestro programa... el `.exe` no es necesario.
- `-Wall:` (warning all) Indica que queremos que el compilador nos muestre todas las advertencias que considere necesarias respecto de nuestro fuente.
- `-ansi:` Indica (si estamos compilando un fuente C) que queremos que compile según el standard ANSI-C '89 (es lo que requerimos en este curso).
- `-pedantic:` Indica que queremos que el código se compile según el standard ISO-C.

Estos argumentos que se han presentado alcanzan como para poder comenzar a trabajar con el compilador... se presenta una lista de otros argumentos útiles (no salva de leer la documentación que acompaña al compilador ;) que acepta el programa:

- `-x c:` Obliga al compilador a compilar nuestro fuente como si fuera en C. El GCC compila muchos otros lenguajes y se vale de la extensión para saber en cuál compilar `.c` lo reconoce como C.
- `-Os:` Optimiza la compilación para que el ejecutable ocupe menos espacio.
- `-O3:` (excluyente con el anterior) Optimiza la compilación para velocidad.
- `-c:` Solamente genera el código objeto, no linkea.
- `-S:` Solamente genera el código assembler, no compila.
- `-g:` Compila a nuestro programa guardando información para debugging.
- `-lm:` Linkea la biblioteca matemática (esto no se hace por default para economizar espacio).
- `--help:` Esta opción nos muestra un pequeño panorama de los argumentos que toma el programa.

2003-09-09

Sebastián Santisi

santisis@7502.com.ar