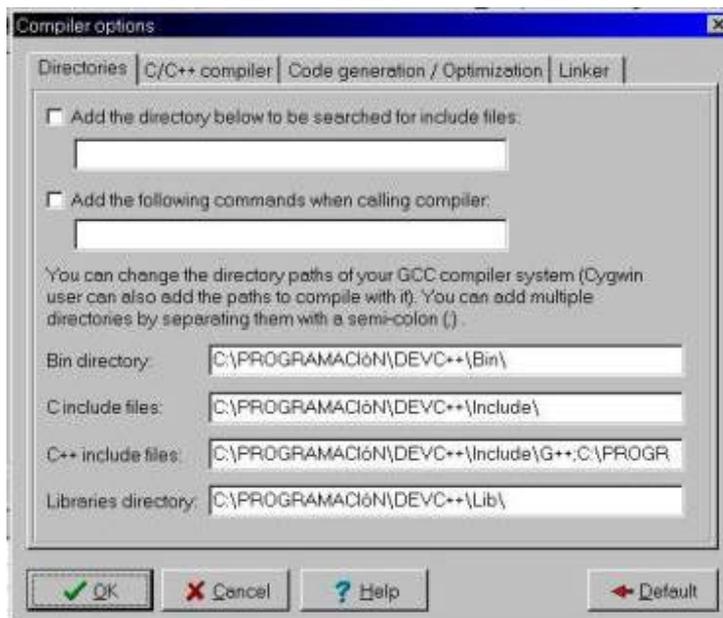


El compilador BloodShed DEV-C++ 4

El DEV C++ es un buen entorno de desarrollo integrado (IDE) para el compilador Mingw (Todo ello freeware). Permite crear programas en C/C++ para DOS o Windows. Se distribuye bajo licencia GPL y por supuesto incluye el código fuente (en Delphi). Ha sido distribuido recientemente por numerosas revistas; además se puede encontrar en <http://www.bloodshed.net/>.

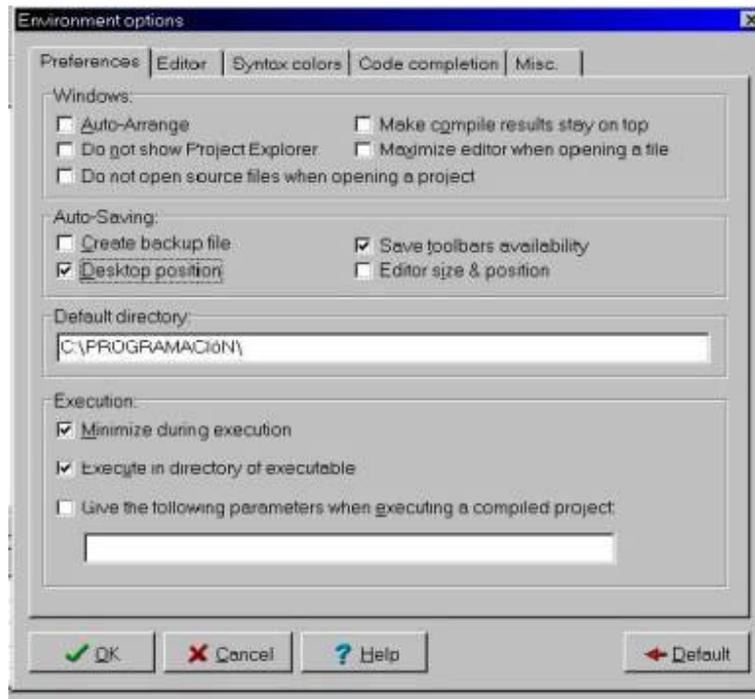
Su instalación se reduce a ejecutar el setup.exe y eligiendo la opción 'típica' estaremos preparados para usarlo en unos instantes.

Una vez abierto el programa vamos al menú Options. Primero pulsamos sobre Compiler Options:



Aquí debería estar todo autoconfigurado. Los directorios arriba señalados dependerán de dónde lo hayas instalado tú.

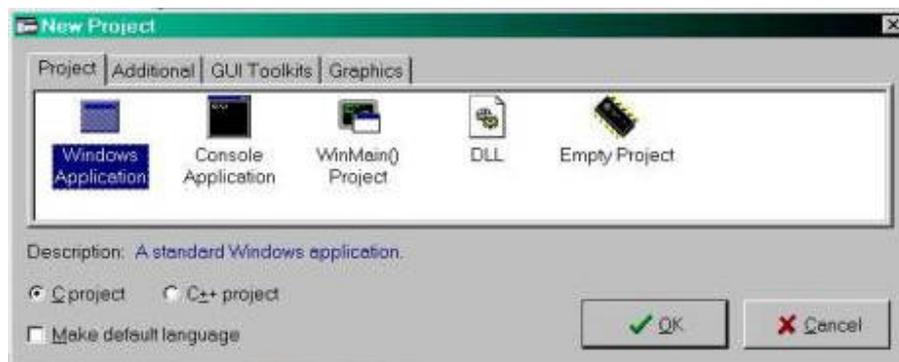
Luego pulsamos en Environment Options:



Pon el default directory que quieras. Las demás solapas sólo afectan a la apariencia del entorno.

Veamos como configurarlo para compilar el ejemplo tut_2_0.zip. Supongamos que hemos descomprimido dicho ejemplo en c:\programación\prog.

Crearemos un nuevo proyecto en File --> New Project (o icono correspondiente):

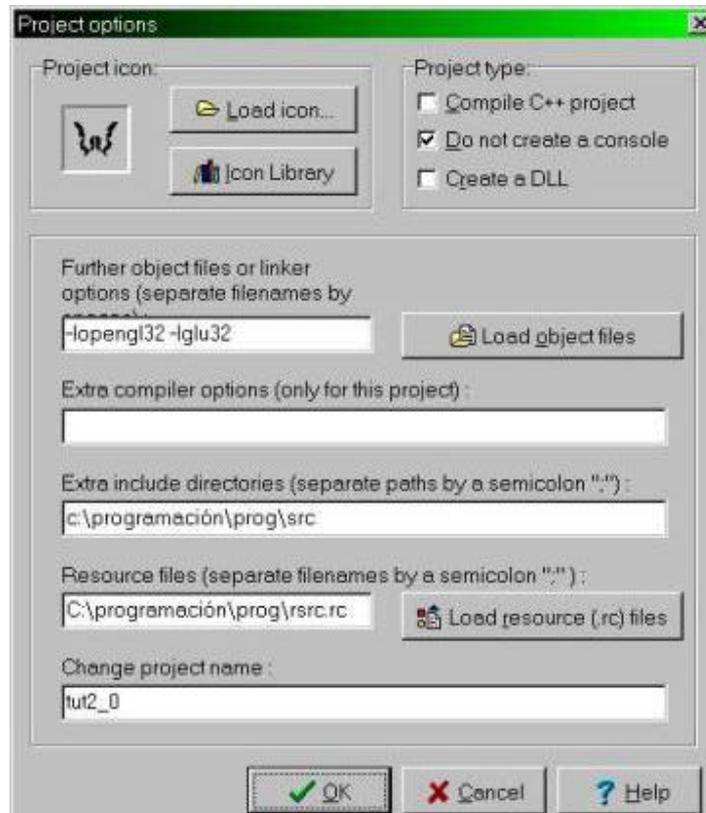


Seleccionamos C project y Windows Application. Las demás solapas las dejamos sin tocar. Luego nos pedirá un nombre para nuestro proyecto y dónde crearlo.

Abrirá una ventana con una plantilla de aplicación Windows. Como no la necesitamos, vamos a project > Remove from project > untitled y elegimos no salvar.

Ahora incluiremos en el proyecto los ficheros de nuestro ejemplo. Para ello vamos a Project  Add to project → elegimos el directorio donde tengamos los ficheros (c:\programación\prog) y manteniendo pulsada la tecla de control pinchamos sobre todos los '.c'.

Ahora sólo falta especificar las propiedades de nuestro proyecto: Project  Project Options.



Aquí podemos seleccionar el icono de nuestra aplicación y lo que es más importante: especificar el uso de las librerías opengl y glu. Para ello en 'further object files' escribimos "-lopengl32 -lglu32" (sin las comillas). Además en 'extra include directories' hemos de especificar el directorio de nuestro ejemplo para que el compilador pueda leer nuestros '.h'.

Bien, ahora ya estaríamos preparados para compilar: (Execute  Compile). Pero para que el ejemplo funcione correctamente en Dev C++ es necesario añadir a 'sys_win32.c' las siguientes líneas (al comienzo del fichero):

```
#ifndef CDS_FULLSCREEN
#define CDS_FULLSCREEN 4
#endif
```

Pues ya está ¡a compilar!