

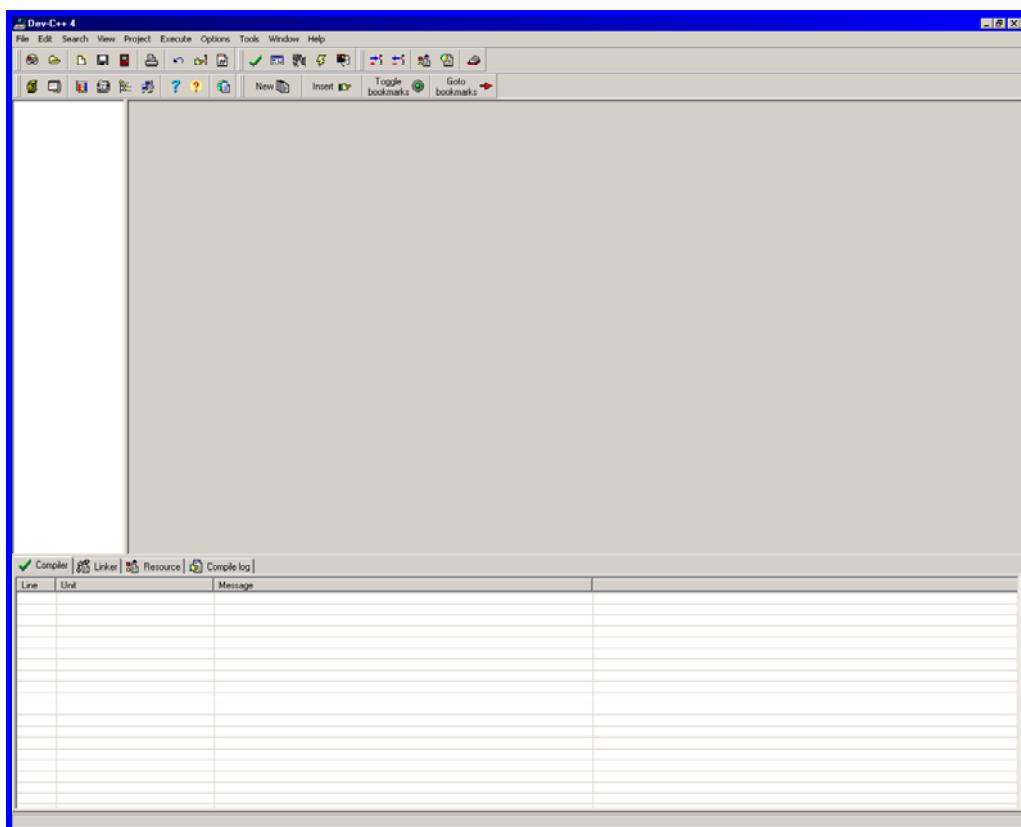
Introducción

Dev-C++ es una IDE (entorno de desarrollo integrado) que facilita herramientas para la creación y depuración de programas en C y en C++. Además, la instalación estándar cuenta también con un compilador de C y C++ (el Mingw) que se instala conjuntamente y se puede utilizar de una forma transparente para el usuario. Dev-C++ (y Mingw) son distribuidos gratuitamente y se pueden conseguir en la página web <http://www.bloodshed.net>. Entre sus principales características está que permite crear fácilmente distintos tipos de aplicaciones (de consola, DLL's, ...) así como proyectos de C y C++ indistintamente, permite editar los archivos de la compilación, añadir librerías a proyectos concretos,

La última versión disponible cuando se redactó este manual era la 5 Beta (versión 4.9.9) pero por razones de mayor seguridad y estabilidad, nosotros realizaremos las prácticas con la última versión estable, que es la 4. Por lo tanto, todas las instrucciones e imágenes que se pueden encontrar aquí corresponden a esta versión. Su adaptación a las nuevas versiones no debería ser muy costosa.

Interfaz de Dev-C++

Cuando uno ejecuta Dev-C++ 4 por primera vez se encuentra en un entorno de trabajo parecido al siguiente:



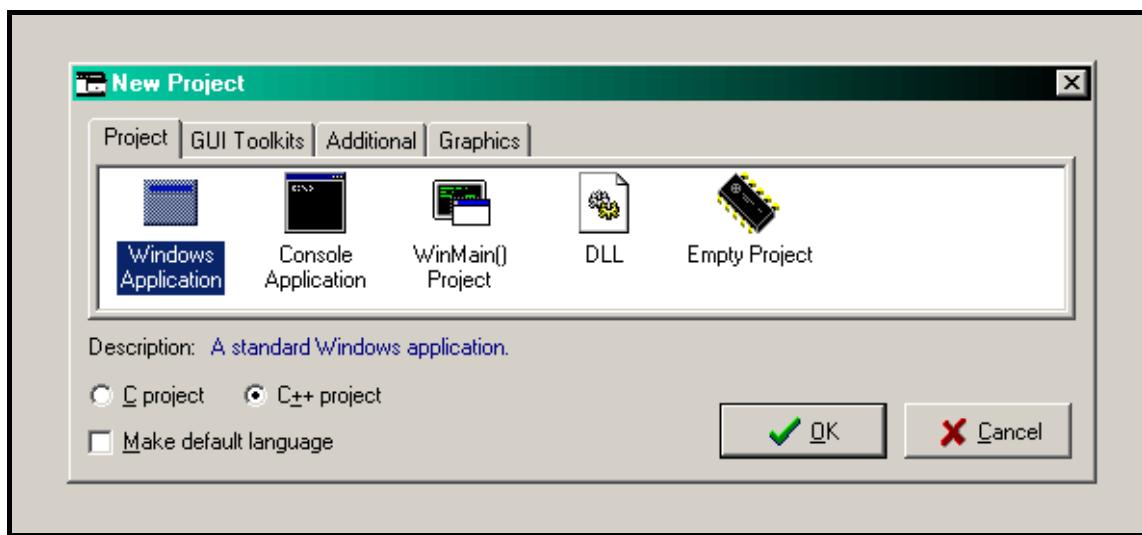
Dentro de ese entorno está permitido abrir, modificar y guardar tanto proyectos como archivos desarrollados en C y C++. La columna izquierda está reservada a la estructura del proyecto. En ella aparecerán tanto el proyecto principal como los ficheros del mismo que se encuentran abiertos en un momento determinado. El espacio central está reservado para los ficheros en C o C++ con los que nos encontramos trabajando. La parte inferior de la interfaz gráfica es la que utiliza el programa para devolver los resultados de la compilación y depuración de los proyectos. Veamos ahora cómo se puede crear un nuevo proyecto.

Creación de un nuevo proyecto

Una vez situados dentro de la aplicación, la forma de crear un proyecto es:

File -> New Project

Tras realizar esa operación, se obtiene la siguiente interfaz, en la cual deberemos elegir qué tipo de proyecto queremos crear, y si el mismo es de C o de C++:



Nos encontramos con 5 opciones principales:

Windows Application: para crear aplicaciones para Windows

Console Application: para crear aplicaciones cuyo interfaz con el usuario será la consola de MS-DOS

WinMain () Project: aplicaciones en las cuales la función main() es sustituida por WinMain ()

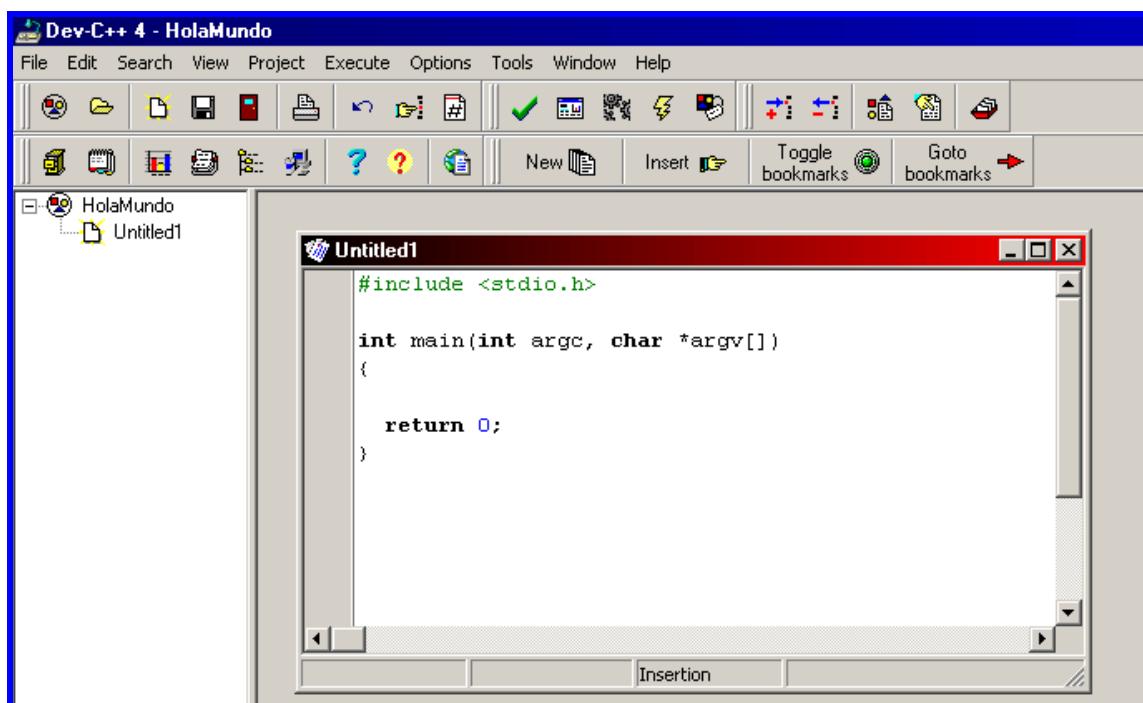
DLL: para crear DLL's (Dynamic Link Library, "Biblioteca de vínculos dinámicos", es un archivo que contiene funciones que se pueden llamar desde aplicaciones u otras DLL) en C o C++.

Empty Project: para proyectos que el usuario va a definir completamente.

Además hay algunas opciones adicionales en las solapas de la ventana que no trataremos ahora.

Se puede elegir también si queremos hacer un proyecto en C o en C++. Obviamente esta decisión es importante, pues de ella depende cómo se compilará el proyecto creado. La opción por defecto en Dev-C++ 4 es C++.

Con todas estas decisiones se crea un nuevo proyecto (al cual le tenemos que asignar un nombre que no tiene por qué tener ninguna relación con el de los ficheros que va a contener), y que inicialmente estará formado por las herramientas que considera Dev-C++ básicas para crear el tipo de proyecto que nosotros hemos elegido. Por ejemplo, en el caso de que nosotros creemos un proyecto llamado HolaMundo que sea una aplicación de consola MS-DOS en lenguaje C++ que saque por pantalla el diálogo "Hola Mundo", el proyecto que generará automáticamente Dev-C++ tendrá el siguiente aspecto:



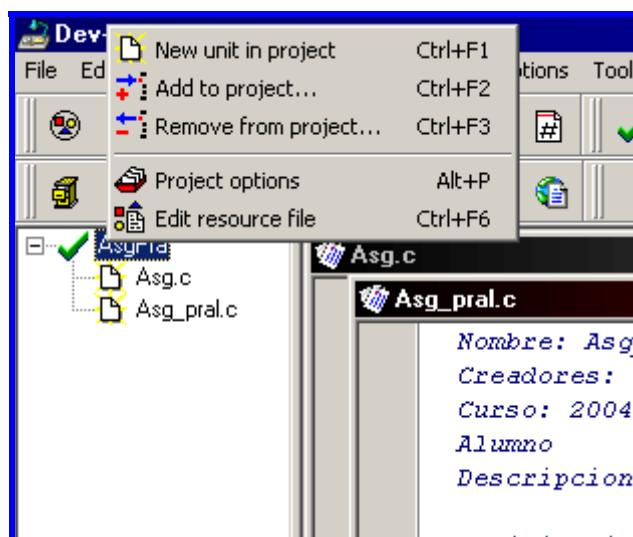
Vemos que se ha creado, en primer lugar, un proyecto llamado HolaMundo, y que por defecto, Dev-C++ le ha añadido un archivo con la información más básica para crear una aplicación que trabaje a través de la ventana de comandos de MS-DOS.

Si hubiésemos elegido una aplicación de Windows, el archivo Untitled añadido al proyecto tendría un aspecto diferente.

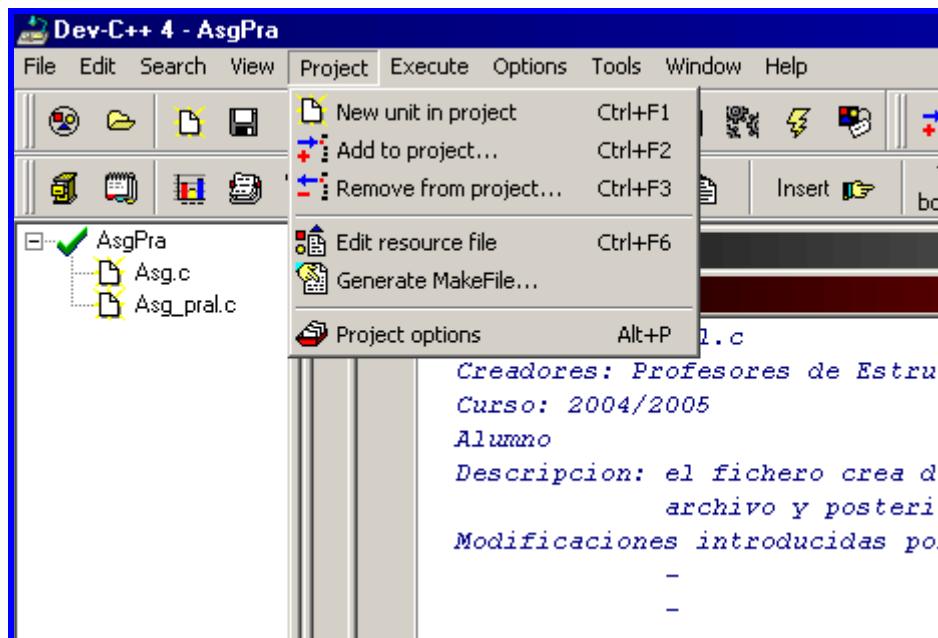
Las opciones elegidas para el proyecto no pueden ser modificadas posteriormente. La única modificación posible es si queremos compilar el proyecto como si fuese de C o de C++.

Modificación de proyectos

Una vez que hemos definido un proyecto, Dev-C++ posee multitud de herramientas que permiten modificar el mismo. Las más importantes son las que nos permiten añadir nuevos ficheros o ficheros ya existentes a nuestro proyecto, así como quitar ficheros al mismo, o modificar las propiedades del proyecto (por ejemplo, el nombre). Estas opciones pueden ser modificadas con el puntero del ratón situado encima del fichero del proyecto en la ventana del **"Gestor de proyectos"**:



o también directamente desde la solapa **"Project"** de la ventana del Dev-C++:



Los comandos que aparecen en las solapas nos permiten:

New unit in project: Añadir un fichero nuevo (en blanco) a nuestro proyecto.

Add to project: Añadir un fichero ya existente de C, C++, o de cabeceras (un .h) a nuestro proyecto.

Remove from project: Permite retirar un fichero de nuestro proyecto (pero no borra ese archivo).

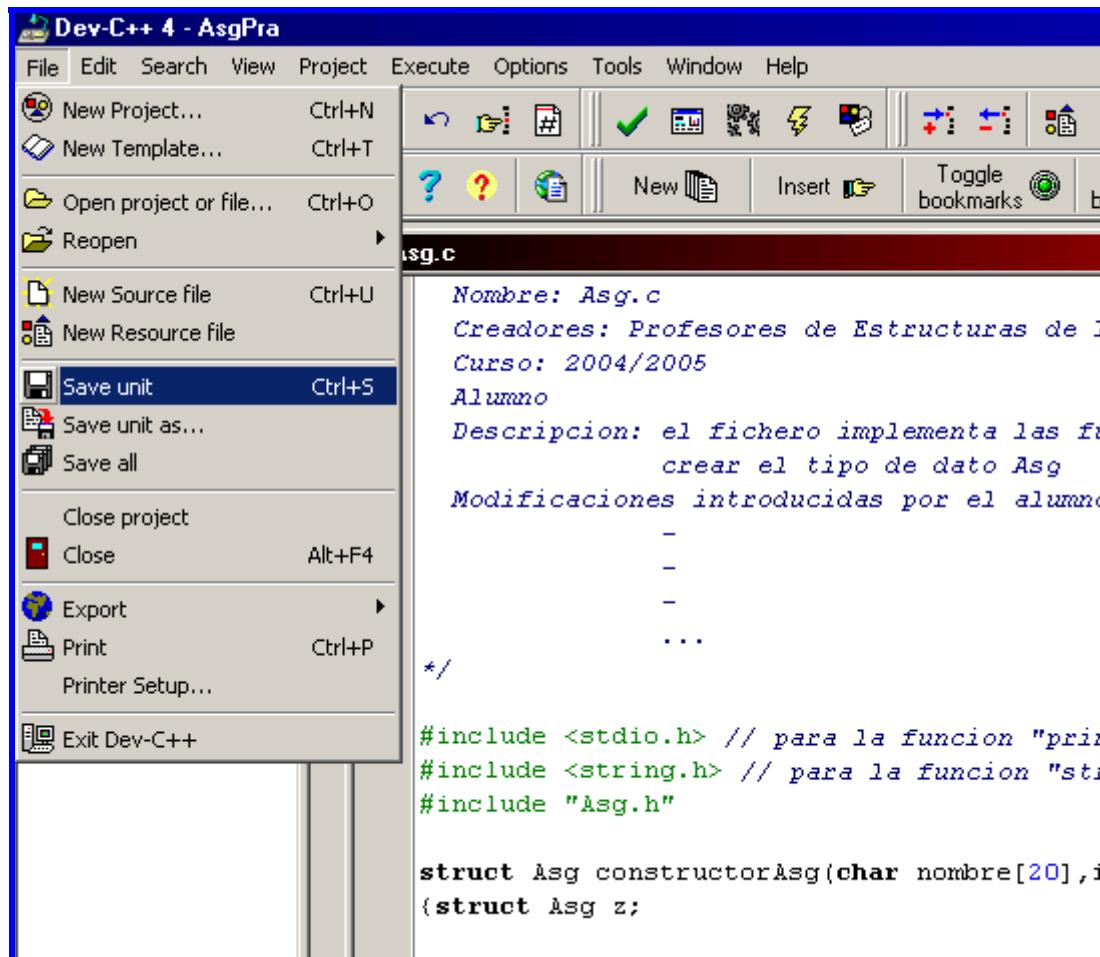
Project options: aquí se pueden modificar múltiples opciones, como el nombre del proyecto, añadir nuevos directorios al mismo, instrucciones al compilador, y hasta modificar el icono del mismo.

Grabar un proyecto

Aunque pueda parecer extraño y en muchas ocasiones se eche en falta, no hay ninguna orden que permita salvar un proyecto. Por tanto, la única herramienta disponible para grabar un proyecto es que éste tenga la forma que nosotros deseamos cuando el mismo se cierra. Por la misma razón, tampoco hay una facilidad que permita grabar un proyecto con otro nombre. Si se desea modificar el nombre de un proyecto se ha de hacer como hemos explicado en la sección anterior, cambiando el nombre del proyecto en las propiedades del mismo. Debido en parte a esto, no es posible gestionar varios proyectos al mismo tiempo con una sola ventana de Dev-C++. Cada vez que queremos tener abiertos varios proyectos al mismo tiempo, deberemos abrir Dev-C++ el mismo número de veces.

Grabar los ficheros de un proyecto

Lo que sí se puede grabar, individualmente o colectivamente, son los distintos ficheros que componen un proyecto. Como se puede observar en la imagen



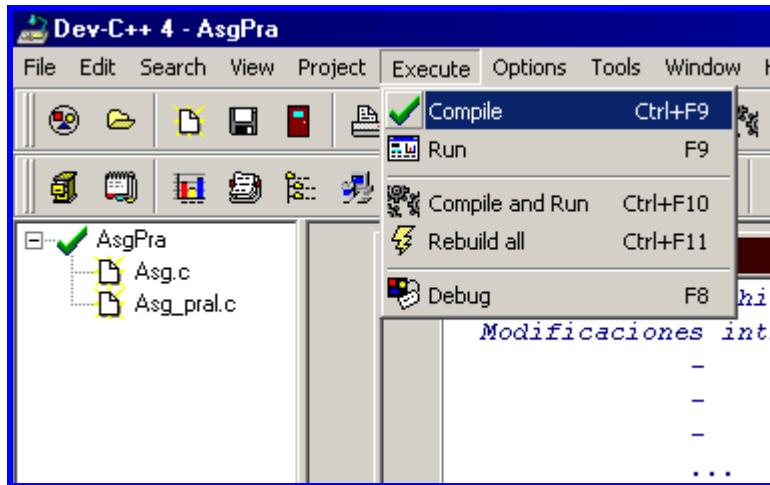
mediante los comandos **“Save unit”**, **“Save unit as...”** o **“Save all”** podemos grabar el fichero actual de nuestro proyecto, el fichero actual con un nuevo nombre, o simplemente todos los ficheros abiertos en nuestra ventana.

Compilación y ejecución de proyectos

Una vez que hemos creado un proyecto, el próximo paso será intentar compilarlo y ejecutarlo. El compilador que utiliza Dev-C++ (por defecto) es Mingw¹, aunque puede ser modificado si lo consideramos necesario.

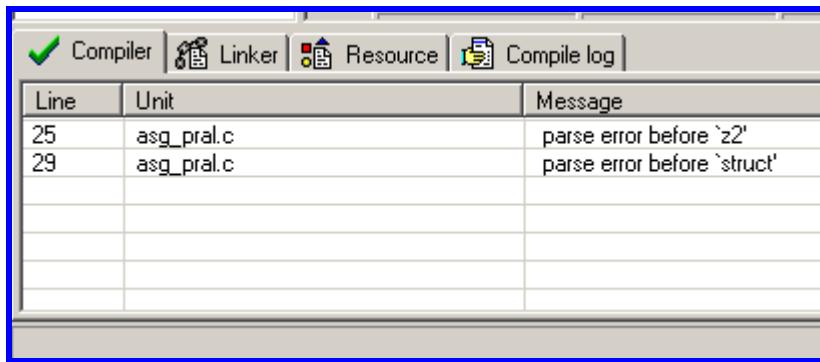
Cuando creamos un proyecto, o en las propiedades, debemos definir si ese proyecto es un proyecto desarrollado en C o en C++, y de esa forma el compilador de Dev-C++ ya sabe si debe aplicar los comandos de compilación propios de C o los de C++, por lo tanto, a la hora de ejecutar, nosotros como usuarios todo lo que debemos hacer es ir al menú **“Execute”** y pulsar **“Compile”**,

¹ Para más información sobre el proyecto Mingw visitar la página web <http://www.mingw.org>

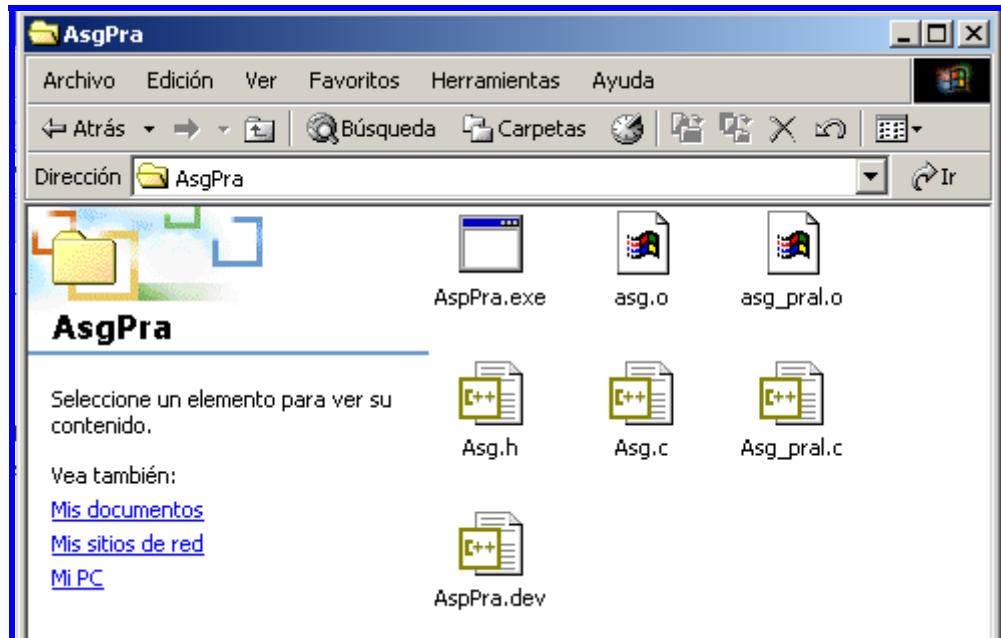


o directamente dirigirnos al ícono de acceso rápido  .

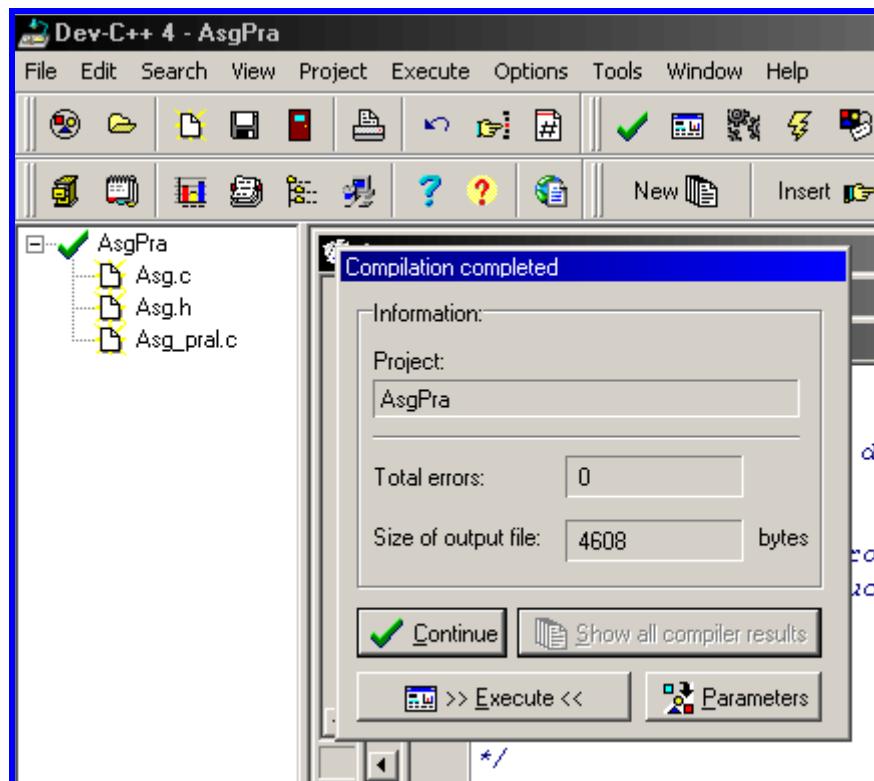
Tanto si el resultado de la compilación es satisfactorio como si no, la parte inferior del Dev-C++ nos dará los resultados de la compilación, con los errores existentes o con un comentario diciendo que el proyecto está bien compilado:

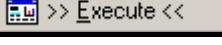


En caso de que esté bien compilado, se habrán generado varios ficheros en la misma carpeta que está el proyecto y sus ficheros. Por cada fichero C o C++ la compilación genera un fichero con el mismo nombre y extensión **".o"**, un fichero objeto, y luego los une todos (junto con las librerías necesarias) para crear el ejecutable. Los ficheros objeto pueden ser borrados, aunque eso hará que posteriores compilaciones del proyecto sean un poco más lentas.



El ejecutable es completamente independiente de los ficheros del proyecto y puede ser trasladado con total facilidad. Para ejecutarlo se puede hacer desde la ventana de diálogo que aparece después de la compilación

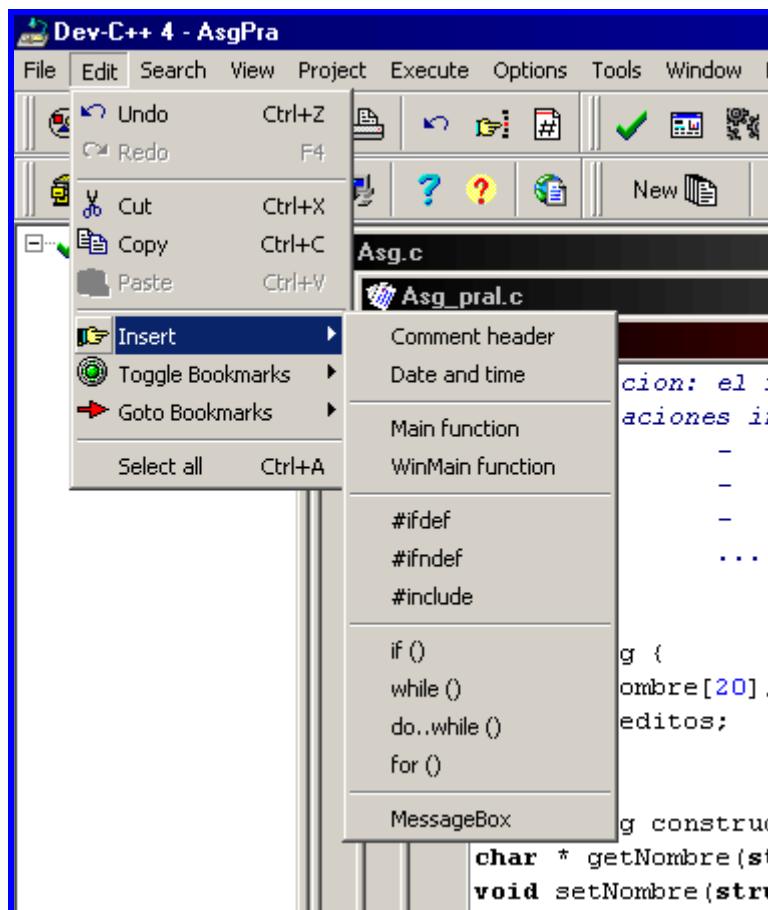


por medio del botón  , desde el menú **Execute** de la barra de tareas o desde el botón de acceso rápido  de la barra de herramientas.

En caso de que la compilación no haya sido satisfactoria, el compilador nos dará alguna información de los errores detectados (por lo menos dónde han sido encontrados) y se tendrá que empezar de nuevo el proceso de compilación.

Otras herramientas

Dev-C++ ofrece otras muchas opciones al usuario en los distintos menús, casi todas ellas similares a las de otras aplicaciones de Windows, como el menú **“Edit”**, que además de las opciones habituales de **“Cut”**, **“Copy”**, **“Select all”**, ..., en Dev-C++ posee otras dos utilidades



“Insert”, que nos permitirá introducir comandos comunes de C y C++, y las opciones **“Toggle Bookmark”** y **“Goto Bookmark”**, que permiten al usuario insertar señales en nuestros programas y luego desplazarnos directamente hasta ellas.

El menú **“Search”** permite hacer búsquedas dentro de los ficheros de nuestros proyectos. En el menú **“View”** se nos permite modificar las barras de herramientas visibles en el entorno de trabajo. En **“Options”** podemos modificar algunas de las opciones referentes al compilador (aunque generalmente dejaremos las que tiene por defecto) o a la interfaz con el usuario. En el menú **“Tools”** se nos permite ejecutar diversas aplicaciones

como el explorador de Windows o la ventana de comandos de MS-DOS. El menú **“Windows”** nos permite gestionar también el entorno de trabajo y seleccionar como activo alguno de los distintos ficheros que tenemos abiertos, y finalmente el programa cuenta con una ayuda en **“Help”** que cubre incluso la Standard Template Library de C++ y que puede ser de utilidad.