

Manual de instalación y uso del entorno de desarrollo utilizado en las prácticas de x86

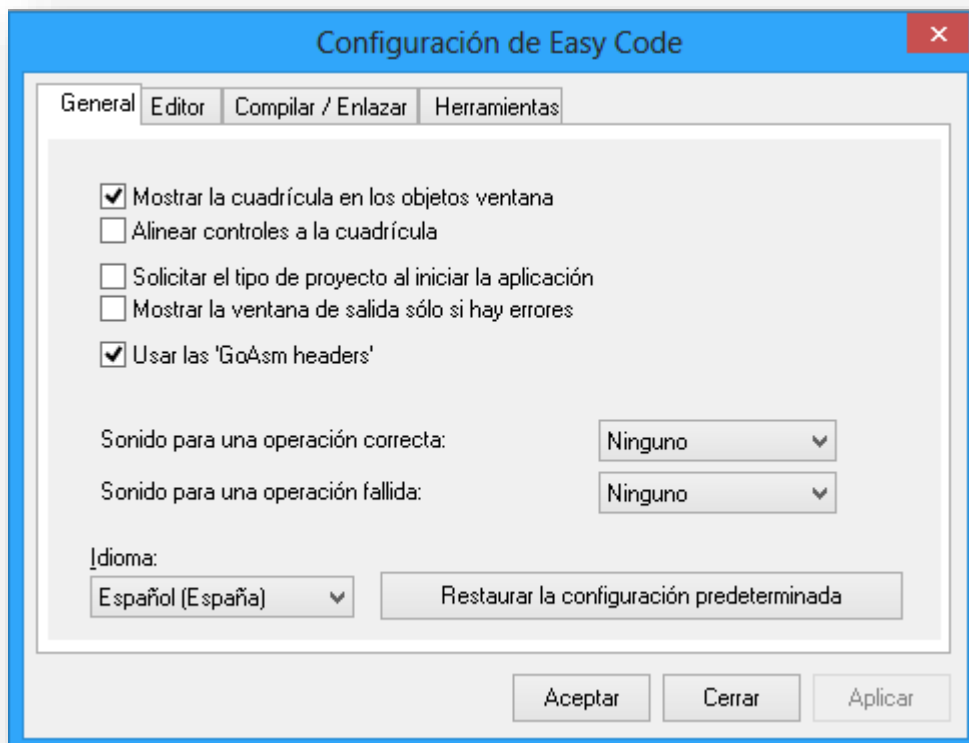
Instalación del entorno

En este manual se tratará la instalación y configuración de las herramientas necesarias para la realización de las prácticas de x86. Comience descargando el software necesario. Necesitaremos:

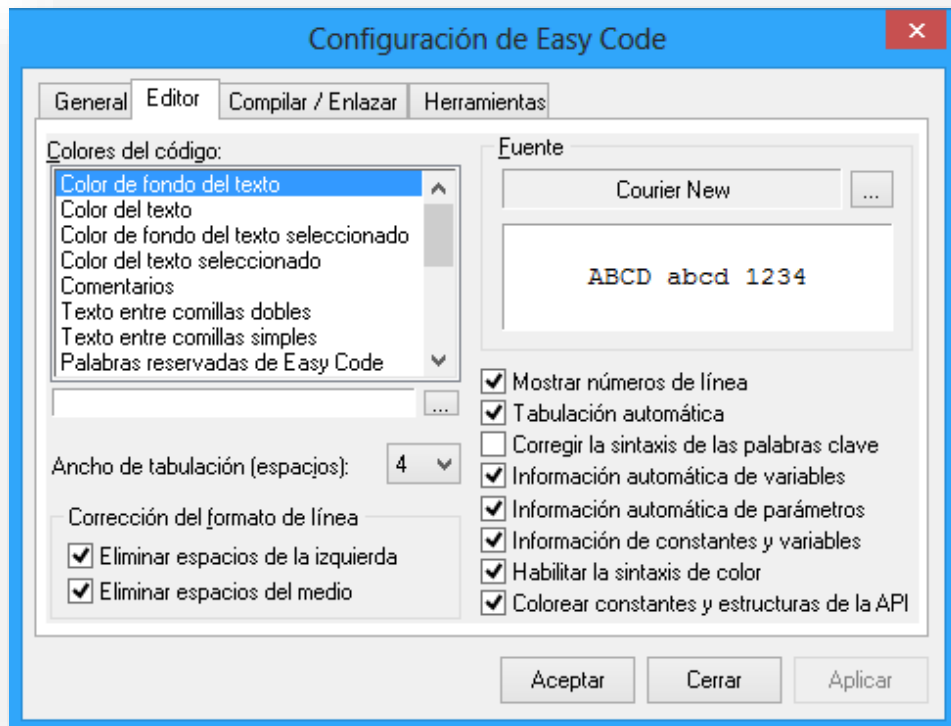
- IDE Easy Code: <http://www.easycode.cat/Spanish/Download.htm>
- GoDev Tools: <http://www.godevtool.com/>. Es preferible sin embargo que descargue las herramientas del campus, puesto que esta página puede dar errores. Además, en el archivo comprimido del campus se encuentran incluidas en la misma carpeta todas las herramientas, mientras que en la página debe descargar el depurador aparte.

Descomprima el archivo descargado de la página del IDE y extraiga la carpeta EasyCode.Go (puede situarla donde guste, pero recuerde la ruta, pues le será útil más adelante). A continuación descargue del campus la carpeta de las GoDev Tools y sitúela en la carpeta EasyCode.Go.

Abra el programa EasyCode (para ello abra la carpeta EasyCode.Go/bin y ejecute el fichero ejecutable que hallará). Ahora debe configurar la ruta donde se ubican las GoDev Tools, para que el IDE pueda utilizarlas. Seleccione Herramientas > Configuración. Aparecerá una ventana como ésta:



Seleccione la casilla “Usar las GoAsm headers” si no está señalada. A continuación seleccione la pestaña “Editor”. Podrá ver una pantalla como la siguiente:



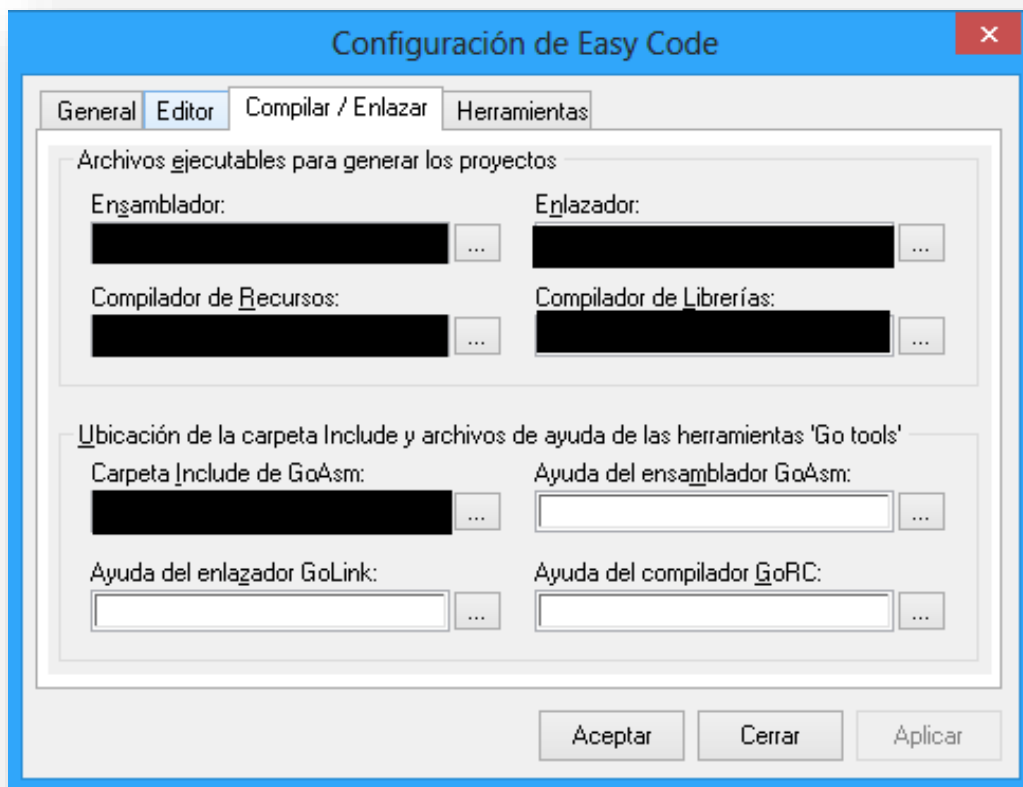
En esta pantalla puede escoger las opciones que desee.

Mientras trabaja en un proyecto, si introduce la siguiente línea:

```
Invoke system, "pause"      ;Invoca la función system("pause")
                             ;Forma parte del API de C
```

Es posible que el IDE cambie “system” por “System”, lo que provoca que el enlazador no encuentre la función. Si este es su caso, **NO** seleccione la opción “Corregir la sintaxis de las palabras clave”. Esta opción no se guarda entre proyectos, así que si crea otro proyecto nuevo, volverá a tener el mismo error. Para solucionarlo acceda a esta ventana, seleccione “Corregir la sintaxis de las palabras clave”, pinche “Aplicar”, vuelva a deseleccionar la opción y de nuevo pinche “Aplicar”. De esta forma, en el proyecto abierto en ese momento no corregirá la sintaxis de las palabras clave.

Seleccione ahora la etiqueta Compilar/Enlazar. Se le mostrará una ventana como la siguiente:



Deberá seleccionar la ruta de los ejecutables. Estos deberían estar en EasyCode.Go/GoAsm/bin. En ensamblador elija “GoAsm.exe”, en enlazador “GoLink.exe”, en compilador de recursos “GoRC.exe” y en compilador de librerías (si aparece) “GoAsm.exe”. En “Carpeta Include de GoAsm” debe poner el directorio Include de GoAsm. Éste debería encontrarse en EasyCode.Go/GoAsm/Include.

Por último seleccione “Aplicar” y “Aceptar” para guardar las configuraciones.

Ya tenemos nuestro IDE listo para empezar a trabajar. En cuanto al depurador, no puede ser integrado con el IDE, así que deberá ejecutarlo aparte (éste debería situarse en EasyCode.Go/GoAsm/GoBug.exe).

En esta práctica se utilizará el Entorno Integrado de Desarrollo (IDE) Easy Code creado por Ramón Sala, que se puede encontrar en la web: <http://www.easycode.cat/Spanish/Download.htm>. Se utilizará la versión para GoAsm.

- En el entorno se integra un editor junto con el ensamblador GoAsm y el enlazador GoLink. También puede usarse aparte el depurador GoBug. Todas estas herramientas, que han sido creadas por Jeremy Gordon, pueden hallarse, junto con la correspondiente documentación, en la web: <http://www.godevtool.com/>. También pueden ser descargadas del campus.
- Versión: 1.06
- Ejecutable en todas las versiones de 32 y 64 bits de Windows desde XP hasta Windows 8.

Uso del entorno y el depurador

Una vez que hemos configurado correctamente el entorno. Para un proyecto nuevo, seleccionaremos “Archivo” → “Proyecto nuevo” → “Aplicación de consola clásica (exe)”.

Se nos creará el proyecto con algunas líneas de código generado automáticamente. El entorno además habrá incluido la cabecera “windows.h” y la biblioteca a enlazar “Kernel32.dll”. La cabecera contiene la definición de las funciones del API de Windows, mientras que la biblioteca “Kernel32.dll” contiene la implementación de algunas de dichas funciones. Recuerde que las funciones que nos serán útiles en las prácticas son “ReadConsole” y “WriteConsole”, pero existen muchas más.

Si queremos hacer uso del API de C necesitaremos incluir entre las bibliotecas a enlazar la biblioteca de C que utiliza Windows. Para ello seleccionamos “Proyecto” → “Agregar archivos para enlazar” y buscamos en la carpeta C:\Windows\System32 la biblioteca msvcrt.dll. Al enlazarla podremos llamar sin problemas a las funciones del API de C.

Centrándonos de nuevo en el código, podemos prescindir del código que ha generado el IDE. Sin embargo recuerde que al principio de la sección de código debe haber una etiqueta “start:”. Además, al finalizar su programa deberá invocar a la función ExitProcess pasándole el valor 0 como parámetro. Si no invoca dicha función, Windows le informará de que el programa ha provocado un fallo, puesto que no ha sido finalizado correctamente. Por lo tanto, sería conveniente que conservara la línea que pone a 0 el registro EAX:

```
XOR EAX, EAX           ;Pone a 0 EAX
```

Así como la línea que invoca a ExitProcess, pasándole EAX como parámetro:

```
Invoke ExitProcess, EAX
```

Sería conveniente también que realizara una llamada a la función system(“pause”) del API de C, puesto que de lo contrario la consola se cerrará una vez que finalice el programa sin permitirle ver los resultados. Tenga en cuenta lo dicho anteriormente sobre la corrección de la sintaxis de las palabras clave y la invocación a la función “system”.

Una vez que ha realizado su código, deberá ensamblarlo para ser posteriormente enlazado y dar lugar así a un código ejecutable. Para generar el ejecutable, seleccione “Generar” → “Regenerar todo”. Se le mostrará en la parte de abajo del IDE si la generación del ejecutable fue satisfactoria o si por el contrario se produjo algún error. Deberá corregir los errores hasta que la generación del ejecutable se realice sin errores. Cuando se haya generado el ejecutable, podrá probarlo seleccionando “Generar” → “Ejecutar Nombre_Proyecto.exe”.

Aunque se muestra también la opción de depuración, algún fallo del IDE no permite depurar el programa directamente desde el IDE, aunque se haya configurado correctamente la ruta del depurador. Si necesita depurar su programa, abra el depurador (que se debería encontrar en EasyCode.Go/GoAsm/bin/GoBug.exe) concediéndole permisos de administrador. Una vez abierto, el depurador preguntará dónde está el ejecutable a depurar. Éste debería encontrarse en Carpeta_Proyecto/Debug/Nombre_proyecto.exe. En la documentación encontrará instrucciones más detalladas sobre el uso del depurador.