

Programación 1110 - 0612

Ingeniería Informática



UNLaM
Universidad
Nacional de
La Matanza

Instalación Codeblocks

2020

Renata S. Guatelli

Clase 1-2

Instalación de Codeblocks

- ¿Qué es un IDE?
- ¿Qué es Codeblocks?
 - Instalación
 - Elementos
 - Crear un Proyecto
 - Ejecutar un Proyecto
 - Agregar archivos a nuestro Proyectos
 - Actividad Propuesta

¿Qué es un IDE?

IDE por sus siglas en inglés ***I**ntegrated **D**evelopment **E**nvironment*, es un **Entorno de Desarrollo Integrado**.

Es un entorno de desarrollo de software que ha sido empaquetado como un programa de aplicación.

Proporciona servicios integrales para facilitar la tarea de desarrollo de software. Maneja proyectos completos (varios archivos).

¿Por qué utilizar un IDE?

Permiten iniciar rápidamente el proceso de trabajo, gracias a que no es necesario ni integrar manualmente ni aprender varias herramientas por separado ya que todas están agrupadas bajo una misma interfaz de usuario.

Al aprender a utilizar un único producto, aprendemos a utilizar varios elementos necesarios para crear el programa final.

IDE – Elementos

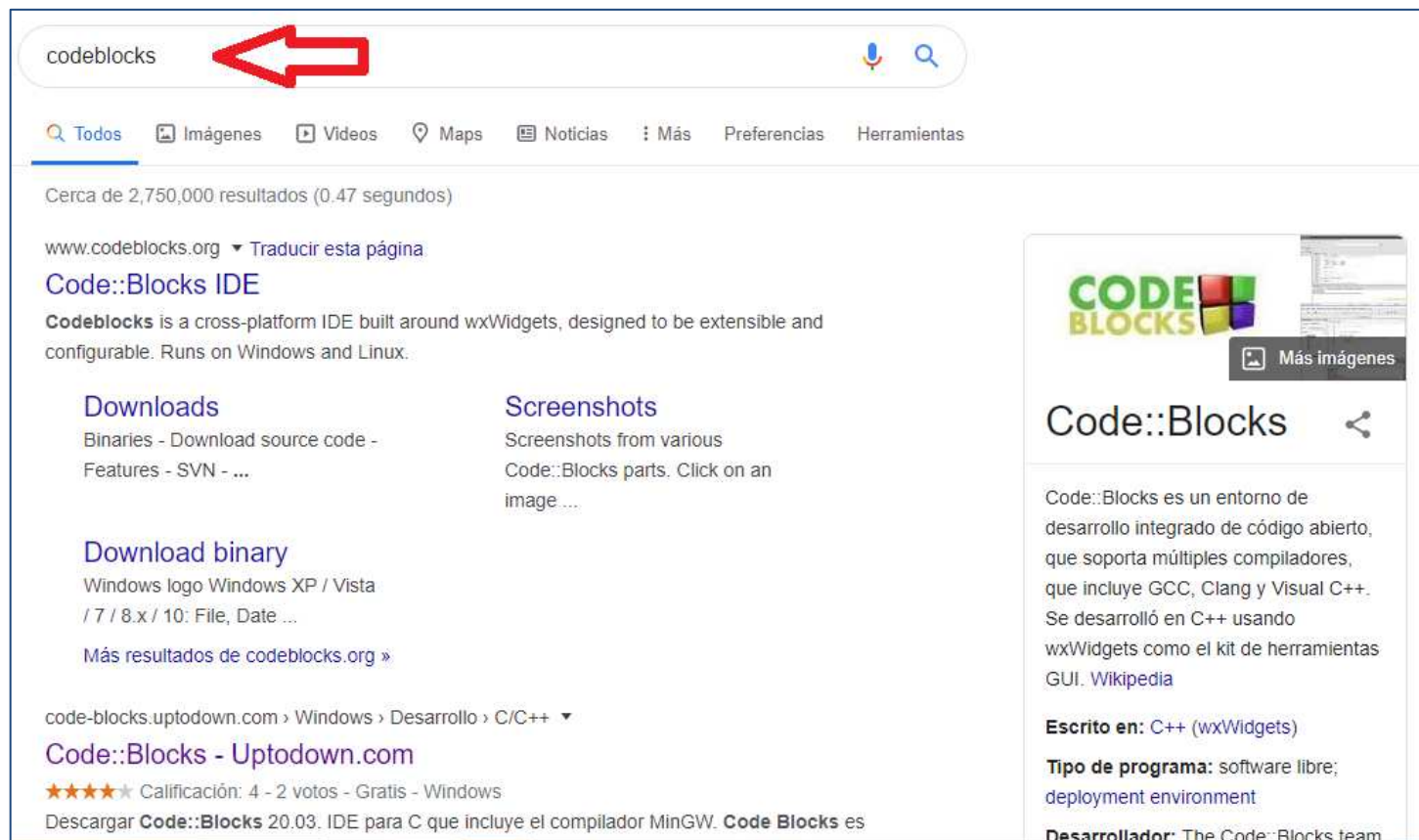
- editor de código fuente
- depurador.
- La mayoría de los IDE tienen auto - completado inteligente de código (IntelliSense).
- Algunos IDE contienen un compilador, un intérprete, o ambos.

¿Qué es Codebloks?

- Es un entorno integrado de desarrollo (IDE)
- multiplataforma,
- de código abierto que
- soporta la utilización de múltiples compiladores, entre ellos: GCC (MingW / GNU GCC). El compilador que viene por defecto con este paquete de Code Blocks es **MinGW** (es el que vamos a utilizar nosotros).

Codeblocks - Instalación

Ingresar al navegador y en la barra de búsqueda escribir: **Codeblocks**



Codeblocks - Instalación

1. Bajar el instalador

Seleccionar www.codeblocks.org

codeblocks

Todos Imágenes Videos Maps Noticias Más Preferencias Herramientas

Cerca de 2.750.000 resultados (0.47 segundos)

www.codeblocks.org Traducir esta página

Code::Blocks IDE

Codeblocks is a cross-platform IDE built around wxWidgets, designed to be extensible and configurable. Runs on Windows and Linux.

Downloads
Binaries - Download source code -
Features - SVN - ...

Screenshots
Screenshots from various
Code::Blocks parts. Click on an
image ...

Download binary
Windows logo Windows XP / Vista
/ 7 / 8.x / 10: File, Date ...
Más resultados de codeblocks.org »

code-blocks.uptodown.com > Windows > Desarrollo > C/C++

Code::Blocks - Uptodown.com

★★★★★ Calificación: 4 - 2 votos - Gratis - Windows

Descargar **Code::Blocks** 20.03. IDE para C que incluye el compilador MinGW. **Code Blocks** es

CODE BLOCKS Más imágenes

Code::Blocks

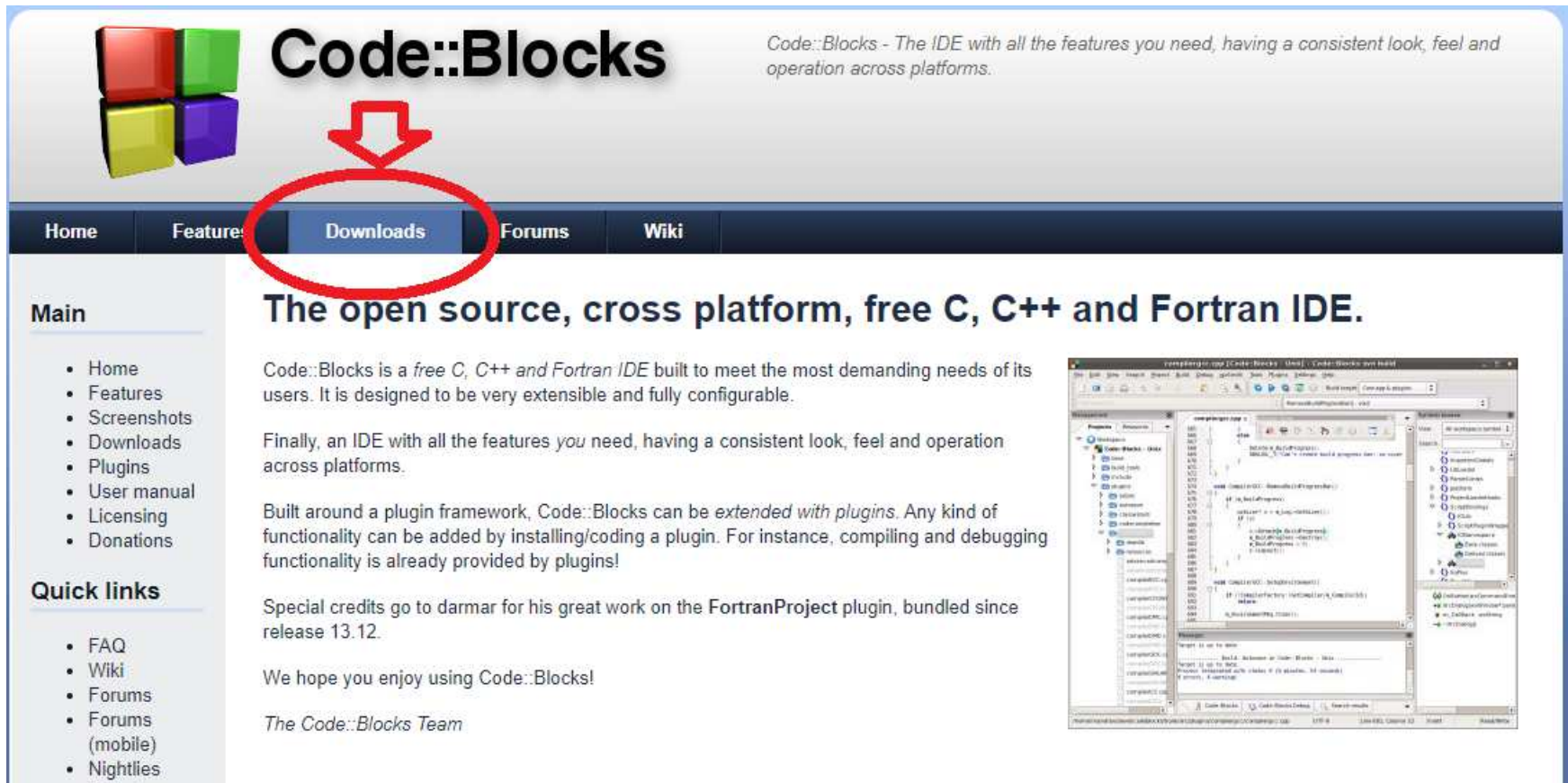
Code::Blocks es un entorno de desarrollo integrado de código abierto, que soporta múltiples compiladores, que incluye GCC, Clang y Visual C++. Se desarrolló en C++ usando wxWidgets como el kit de herramientas GUI. [Wikipedia](#)

Escrito en: C++ (wxWidgets)

Tipo de programa: software libre; deployment environment

Desarrollador: The Code::Blocks team

Estamos en el sitio oficial de Codeblocks.
Seleccionamos la opción “Downloads”



The screenshot shows the Code::Blocks website. The logo is on the left, followed by the text "Code::Blocks" and a tagline: "Code::Blocks - The IDE with all the features you need, having a consistent look, feel and operation across platforms." Below this is a navigation bar with links: Home, Features, Downloads, Forums, and Wiki. The "Downloads" link is circled in red, with a red arrow pointing to it from above. To the left of the main content is a sidebar with a "Main" section containing links to Home, Features, Screenshots, Downloads, Plugins, User manual, Licensing, and Donations. Below that is a "Quick links" section with links to FAQ, Wiki, Forums, Forums (mobile), and Nightlies. The main content area has the heading "The open source, cross platform, free C, C++ and Fortran IDE." followed by a paragraph: "Code::Blocks is a free C, C++ and Fortran IDE built to meet the most demanding needs of its users. It is designed to be very extensible and fully configurable." Another paragraph follows: "Finally, an IDE with all the features you need, having a consistent look, feel and operation across platforms." A third paragraph states: "Built around a plugin framework, Code::Blocks can be extended with plugins. Any kind of functionality can be added by installing/coding a plugin. For instance, compiling and debugging functionality is already provided by plugins!" A fourth paragraph mentions: "Special credits go to darmar for his great work on the FortranProject plugin, bundled since release 13.12." A fifth paragraph says: "We hope you enjoy using Code::Blocks!" and the text "The Code::Blocks Team" is at the bottom. On the right side of the main content area is a screenshot of the Code::Blocks IDE interface, showing a project explorer, a code editor with C++ code, and a console window.

Code::Blocks

Code::Blocks - The IDE with all the features you need, having a consistent look, feel and operation across platforms.

Home Features **Downloads** Forums Wiki

Main

- Home
- Features
- Screenshots
- Downloads
- Plugins
- User manual
- Licensing
- Donations

Quick links

- FAQ
- Wiki
- Forums
- Forums (mobile)
- Nightlies

The open source, cross platform, free C, C++ and Fortran IDE.

Code::Blocks is a free C, C++ and Fortran IDE built to meet the most demanding needs of its users. It is designed to be very extensible and fully configurable.

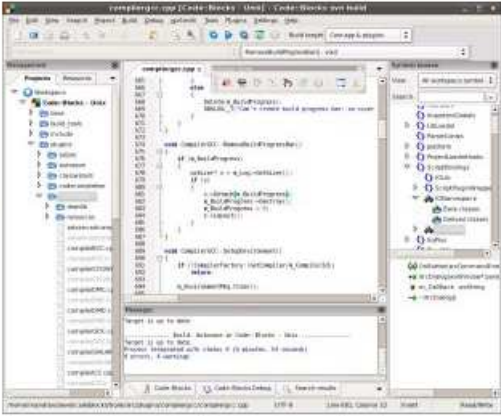
Finally, an IDE with all the features you need, having a consistent look, feel and operation across platforms.

Built around a plugin framework, Code::Blocks can be extended with plugins. Any kind of functionality can be added by installing/coding a plugin. For instance, compiling and debugging functionality is already provided by plugins!

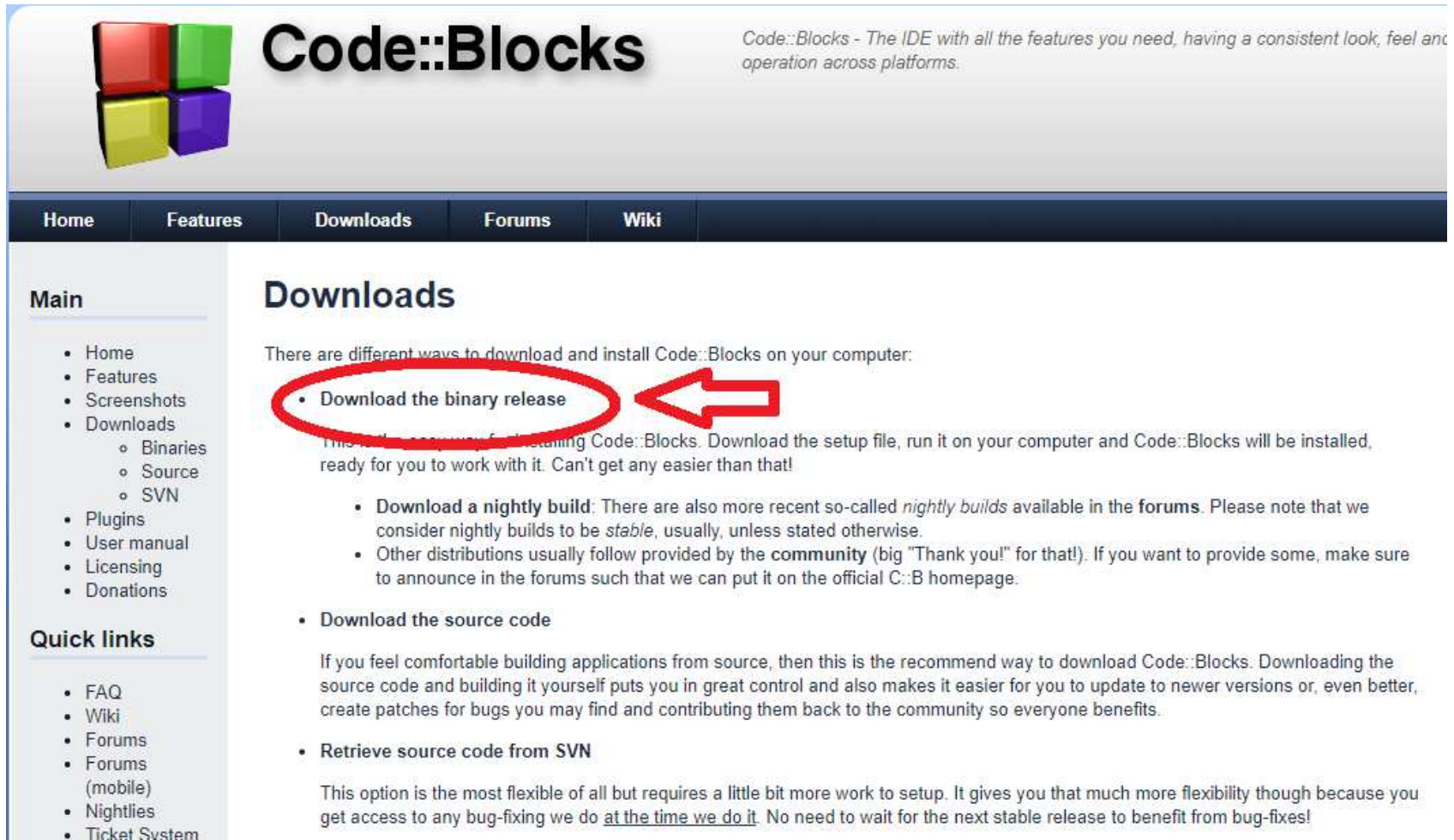
Special credits go to darmar for his great work on the FortranProject plugin, bundled since release 13.12.

We hope you enjoy using Code::Blocks!

The Code::Blocks Team



Dentro de la solapa **Downloads**, elegir la opción:
“Download the binary release”



Code::Blocks *Code::Blocks - The IDE with all the features you need, having a consistent look, feel and operation across platforms.*

Home Features **Downloads** Forums Wiki

Main

- Home
- Features
- Screenshots
- Downloads
 - Binaries
 - Source
 - SVN
- Plugins
- User manual
- Licensing
- Donations

Quick links

- FAQ
- Wiki
- Forums
- Forums (mobile)
- Nightlies
- Ticket System

Downloads

There are different ways to download and install Code::Blocks on your computer:

- **Download the binary release**
This is the **easiest way** for obtaining Code::Blocks. Download the setup file, run it on your computer and Code::Blocks will be installed, ready for you to work with it. Can't get any easier than that!
 - **Download a nightly build:** There are also more recent so-called *nightly builds* available in the **forums**. Please note that we consider nightly builds to be *stable*, usually, unless stated otherwise.
 - Other distributions usually follow provided by the **community** (big "Thank you!" for that!). If you want to provide some, make sure to announce in the forums such that we can put it on the official C::B homepage.
- **Download the source code**
If you feel comfortable building applications from source, then this is the recommend way to download Code::Blocks. Downloading the source code and building it yourself puts you in great control and also makes it easier for you to update to newer versions or, even better, create patches for bugs you may find and contributing them back to the community so everyone benefits.
- **Retrieve source code from SVN**
This option is the most flexible of all but requires a little bit more work to setup. It gives you that much more flexibility though because you get access to any bug-fixing we do at the time we do it. No need to wait for the next stable release to benefit from bug-fixes!

Dentro de “**Download the binary release**”,
primero seleccionamos nuestro sistema operativo



The screenshot shows the Code::Blocks website. The header features the Code::Blocks logo and the tagline "Code::Blocks - The IDE with all the features you need, having a consistent look, feel and operation across platforms." Below the header is a navigation bar with links to Home, Features, Downloads, Forums, and Wiki. The main content area is titled "Main" and contains a list of links: Home, Features, Screenshots, Downloads (with sub-links for Binaries, Source, and SVN), Plugins, User manual, Licensing, and Donations. A "Quick links" section includes links to FAQ, Wiki, Forums, Forums (mobile), Nightlies, and Ticket System. The central text prompts users to "Please select a setup package depending on your platform:" and lists three options: "Windows XP / Vista / 7 / 8.x / 10", "Linux 32 and 64-bit", and "Mac OS X". A red circle highlights the Windows option, and a red arrow points to it. Below this, there are three notes: "NOTE: For older releases. There are releases for many OS version and platforms on the Sourceforge.net page.", "NOTE: There are also more recent nightly builds available in the forums or (for Ubuntu users) in the Ubuntu PPA repository. Please note that we consider nightly builds to be stable, usually.", and "NOTE: We have a Changelog for 20.03, that gives you an overview over the enhancements and fixes we have put in the new release." The section for "Windows XP / Vista / 7 / 8.x / 10:" includes a table of download links.

File	Date	Download from
codeblocks-20.03-setup.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03-setup-nonadmin.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03-nosetup.zip	29 Mar 2020	FossHUB or Sourceforge.net

Si el sistema operativo seleccionado es Windows, Codeblocks permite seleccionar opciones que incluyen el compilador **mingw**.

Si selecciona otro sistema operativo deberá instalar usted un compilador.



Para Windows

Opción 1: incluye el compilador GCC / G ++ / GFortran y el depurador GDB del proyecto MinGW-W64 (versión ... 32/64 bit, SEH).



File	Date	Download from
codeblocks-20.03-setup.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03-setup-nonadmin.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03-nosetup.zip	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03mingw-setup.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03mingw-nosetup.zip	29 Mar 2020	FossHUB or Sourceforge.net

NOTE: The codeblocks-20.03-setup.exe file includes Code::Blocks with all plugins. The codeblocks-20.03-setup-nonadmin.exe file is provided for convenience to users that do not have administrator rights on their machine(s).

NOTE: The codeblocks-20.03mingw-setup.exe file includes *additionally* the GCC/G++/GFortran compiler and GDB debugger from **MinGW-W64 project** (version 8.1.0, 32/64 bit, SEH).

NOTE: The codeblocks-20.03(mingw)-nosetup.zip files are provided for convenience to users that are allergic against installers. However, it will not allow to select plugins / features to install (it includes everything) and not create any menu shortcuts. For the "installation" you are on your own.

If unsure, please use codeblocks-20.03mingw-setup.exe!

Para Windows



Windows XP / Vista / 7 / 8.x / 10:

Opción 2: se proporcionan para comodidad de los usuarios que son alérgicos a los instaladores. No permitirá seleccionar características para instalar (incluye todo) y no creará ningún acceso directo. Para la "instalación" usted está solo.



Windows XP / Vista / 7 / 8.x / 10:

File	Date	Download from
codeblocks-20.03-setup.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03-setup-nonadmin.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03-nosetup.zip	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03mingw-setup.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03mingw-nosetup.zip	29 Mar 2020	FossHUB or Sourceforge.net

NOTE: The codeblocks-20.03-setup.exe file includes Code::Blocks with all plugins. The codeblocks-20.03-setup-nonadmin.exe file is provided for convenience to users that do not have administrator rights on their machine(s).

NOTE: The codeblocks-20.03mingw-setup.exe file includes *additionally* the GCC/G++/GFortran compiler and GDB debugger from **MinGW-W64** project (version 8.1.0, 32/64 bit, SEH).

NOTE: The codeblocks-20.03(mingw)-nosetup.zip files are provided for convenience to users that are allergic against installers. However, it will not allow to select plugins / features to install (it includes everything) and not create any menu shortcuts. For the "installation" you are on your own.

If unsure, please use codeblocks-20.03mingw-setup.exe!

Para Windows



Windows XP / Vista / 7 / 8.x / 10:

Opción 3 – **RECOMENDADA POR Codeblock**: Si no está seguro, utilice codeblocks-20.03mingw-setup.exe.



Windows XP / Vista / 7 / 8.x / 10:

File	Date	Download from
codeblocks-20.03-setup.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03-setup-nonadmin.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03-nosetup.zip	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03mingw-setup.exe	29 Mar 2020	FossHUB or Sourceforge.net
codeblocks-20.03mingw-nosetup.zip	29 Mar 2020	FossHUB or Sourceforge.net

NOTE: The codeblocks-20.03-setup.exe file includes Code::Blocks with all plugins. The codeblocks-20.03-setup-nonadmin.exe file is provided for convenience to users that do not have administrator rights on their machine(s).

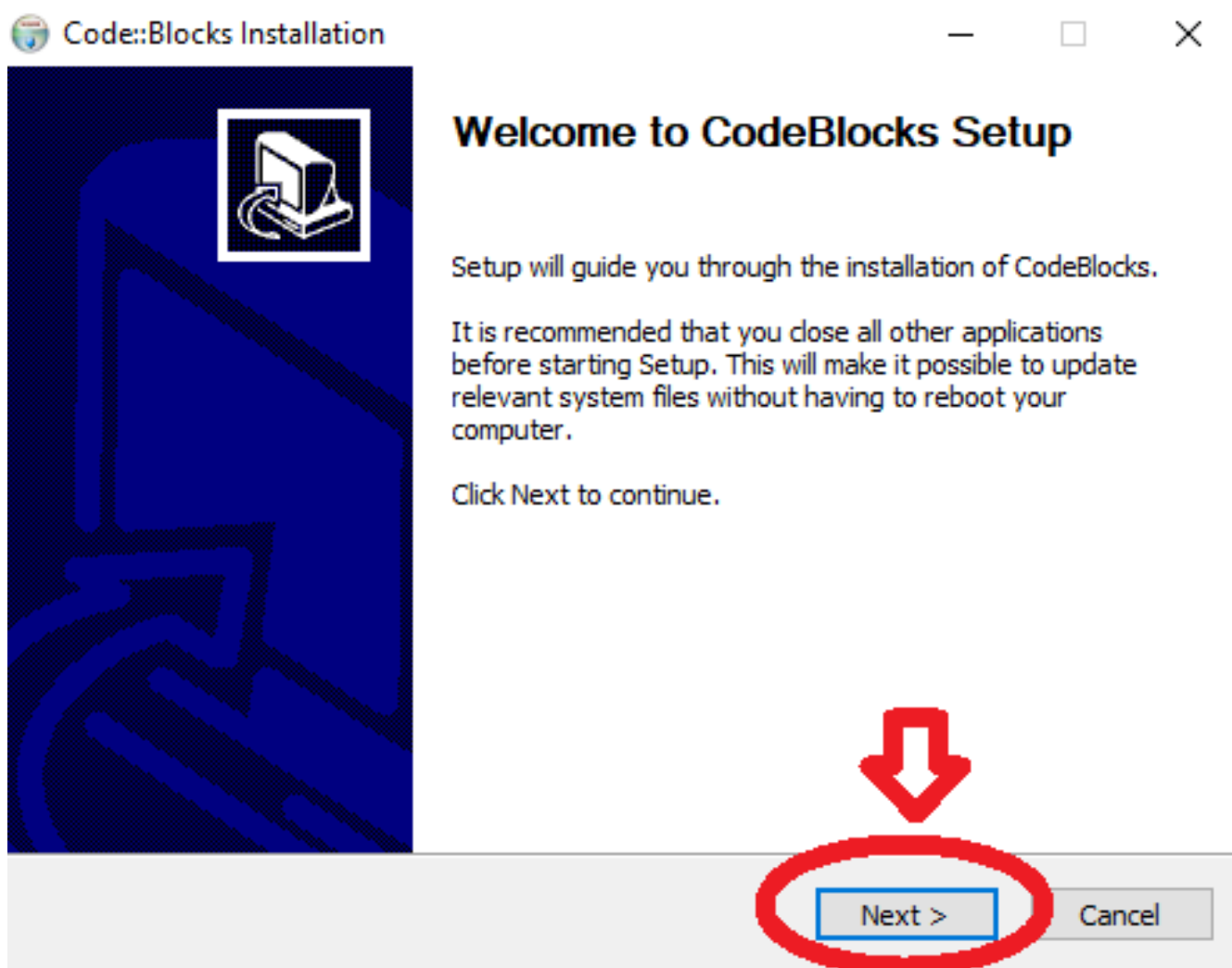
NOTE: The codeblocks-20.03mingw-setup.exe file includes *additionally* the GCC/G++/GFortran compiler and GDB debugger from MinGW-W64 project (version 8.1.0, 32/64 bit, SEH).

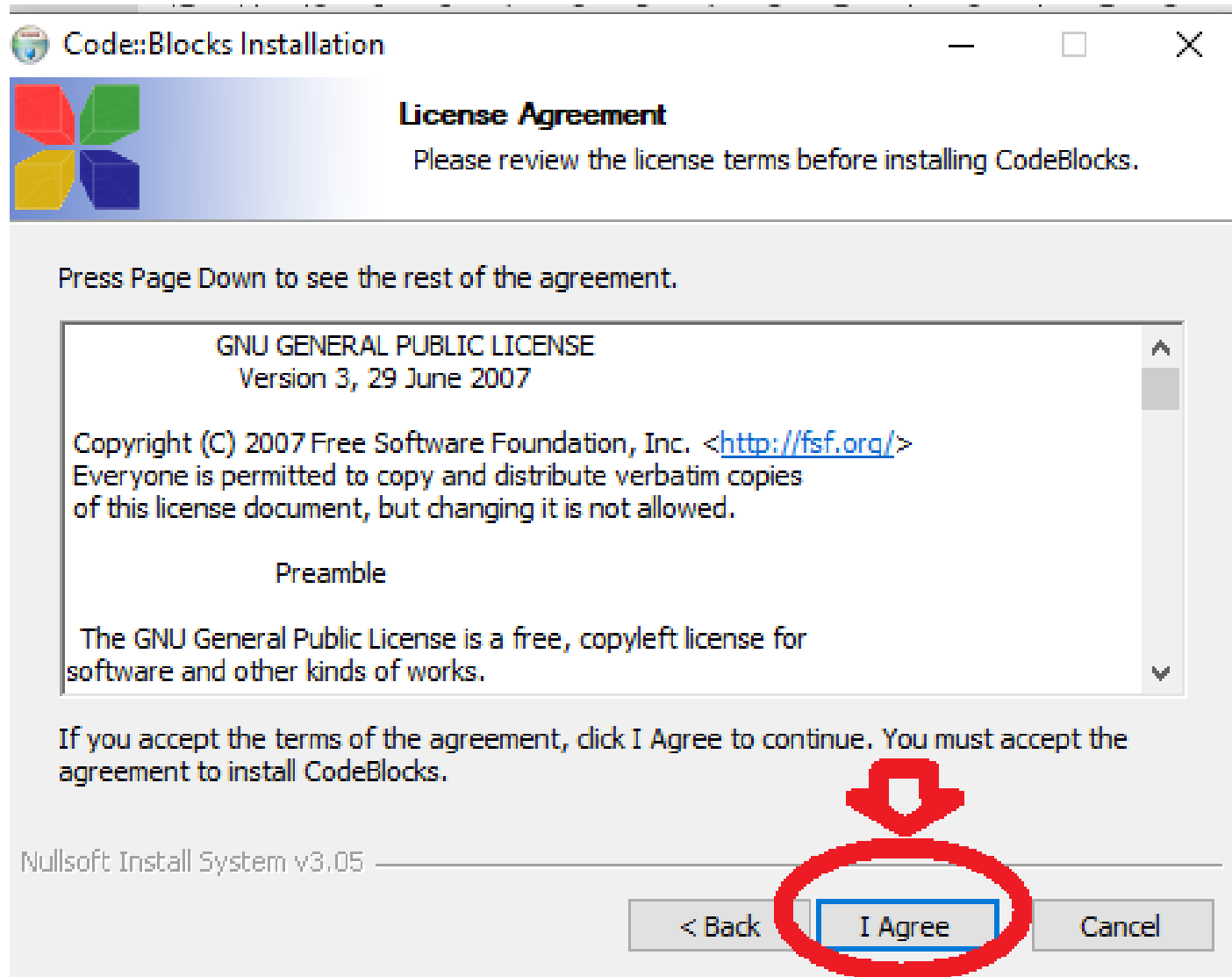
NOTE: The codeblocks-20.03(mingw)-nosetup.zip files are provided for convenience to users that are allergic against installers. However, it will not allow to select plugins / features to install (it includes everything) and not create any menu shortcuts. For the "installation" you are on your own.

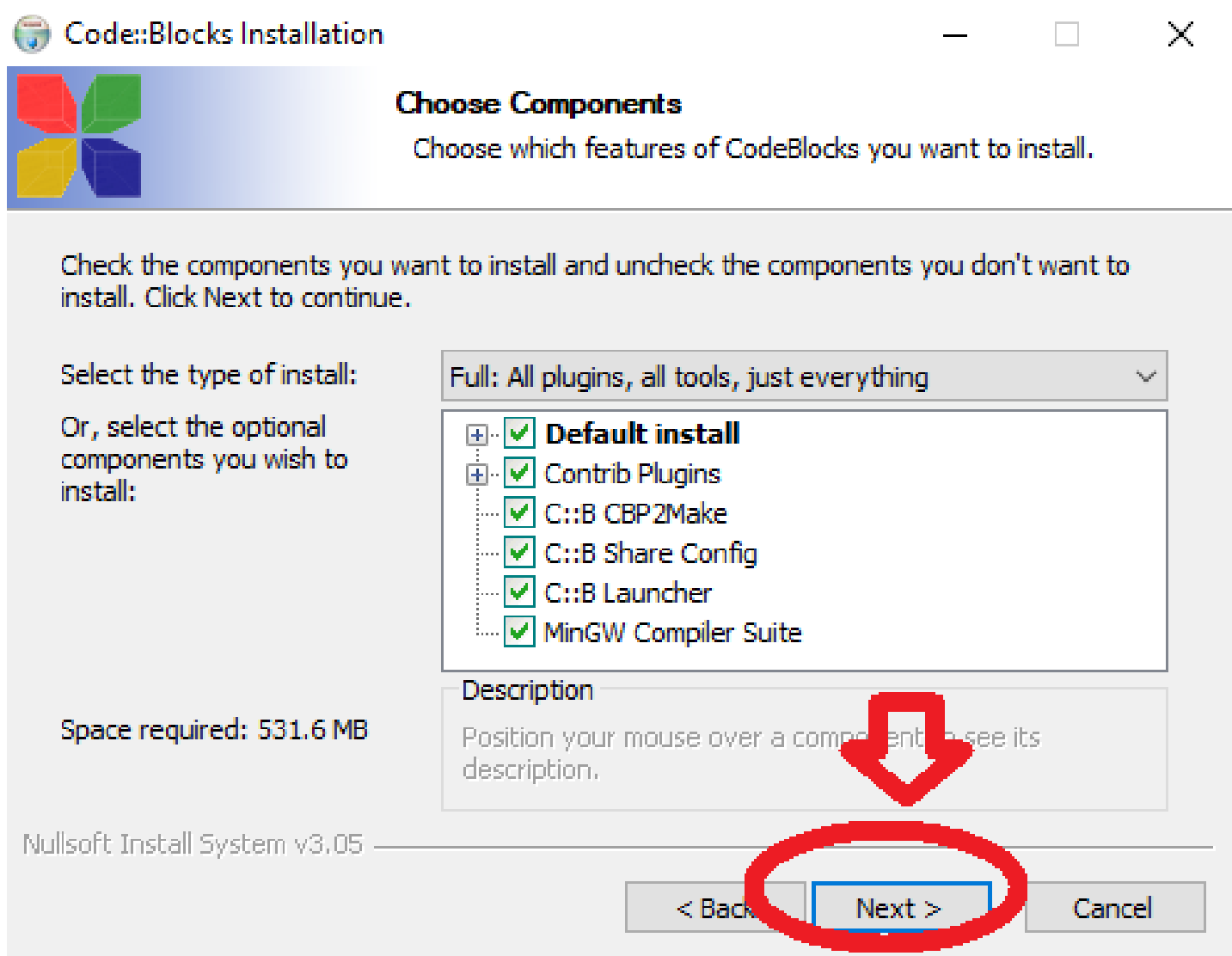
If unsure, please use codeblocks-20.03mingw-setup.exe!

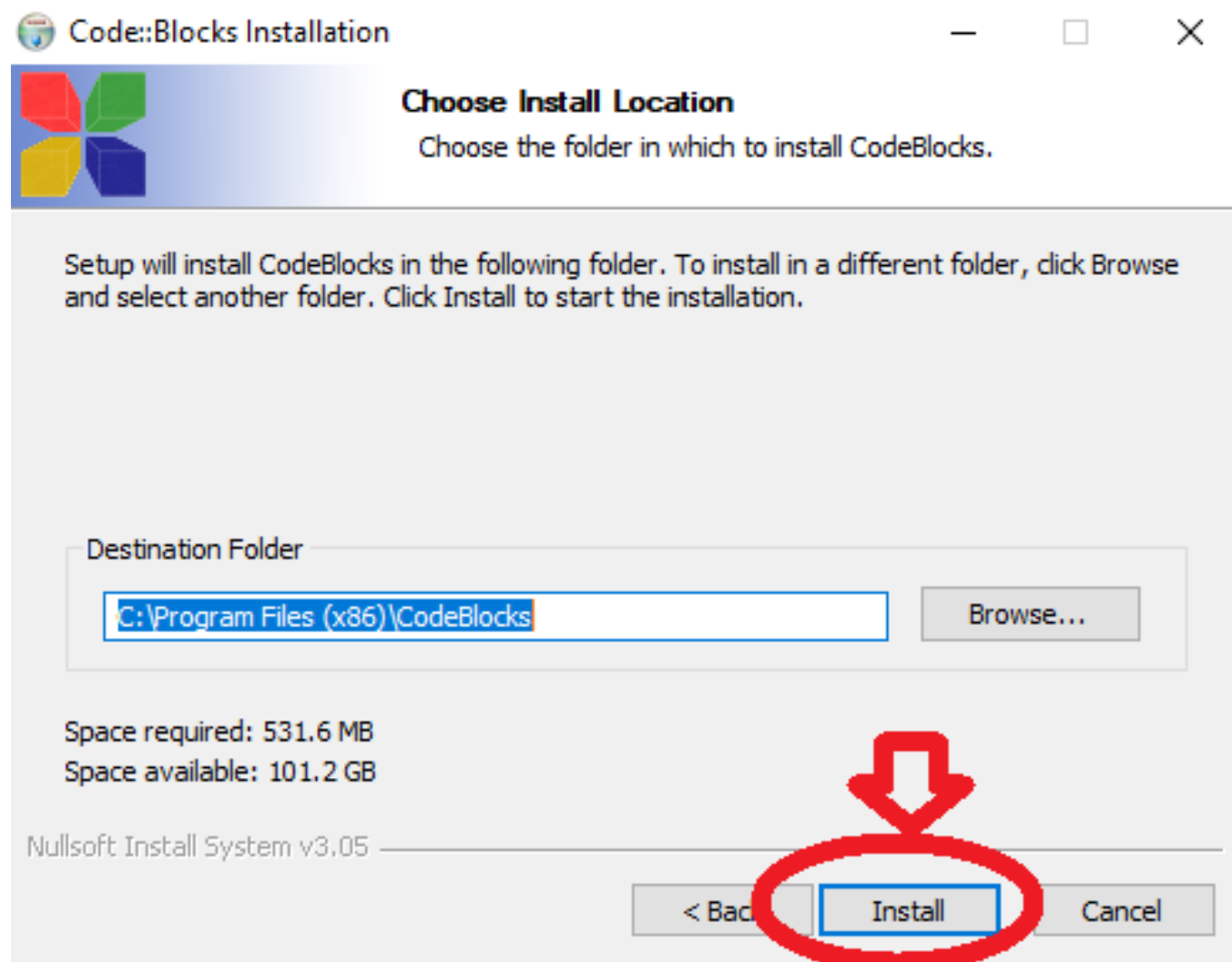
Codeblocks - Instalación

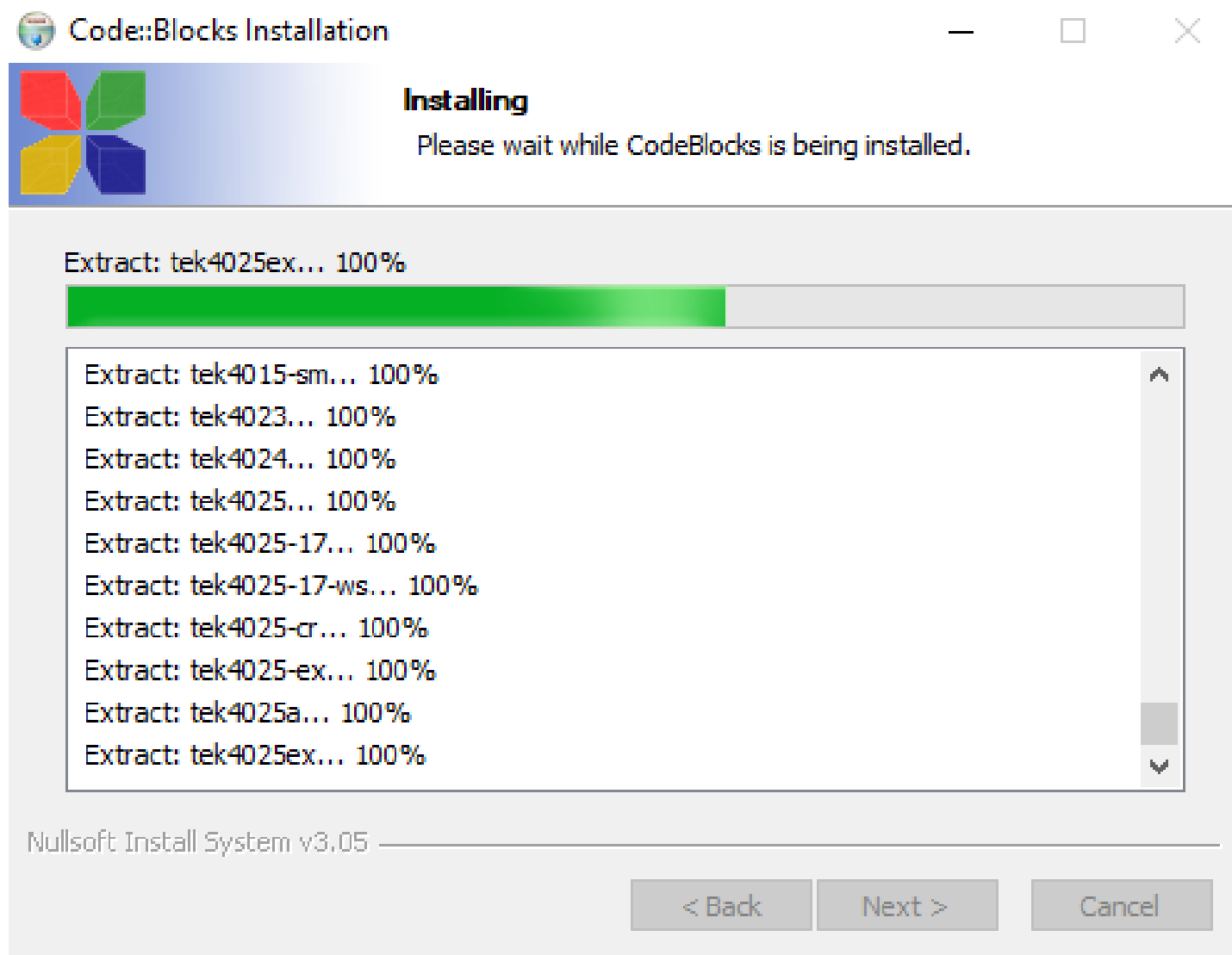
2. Instalar el programa descargado

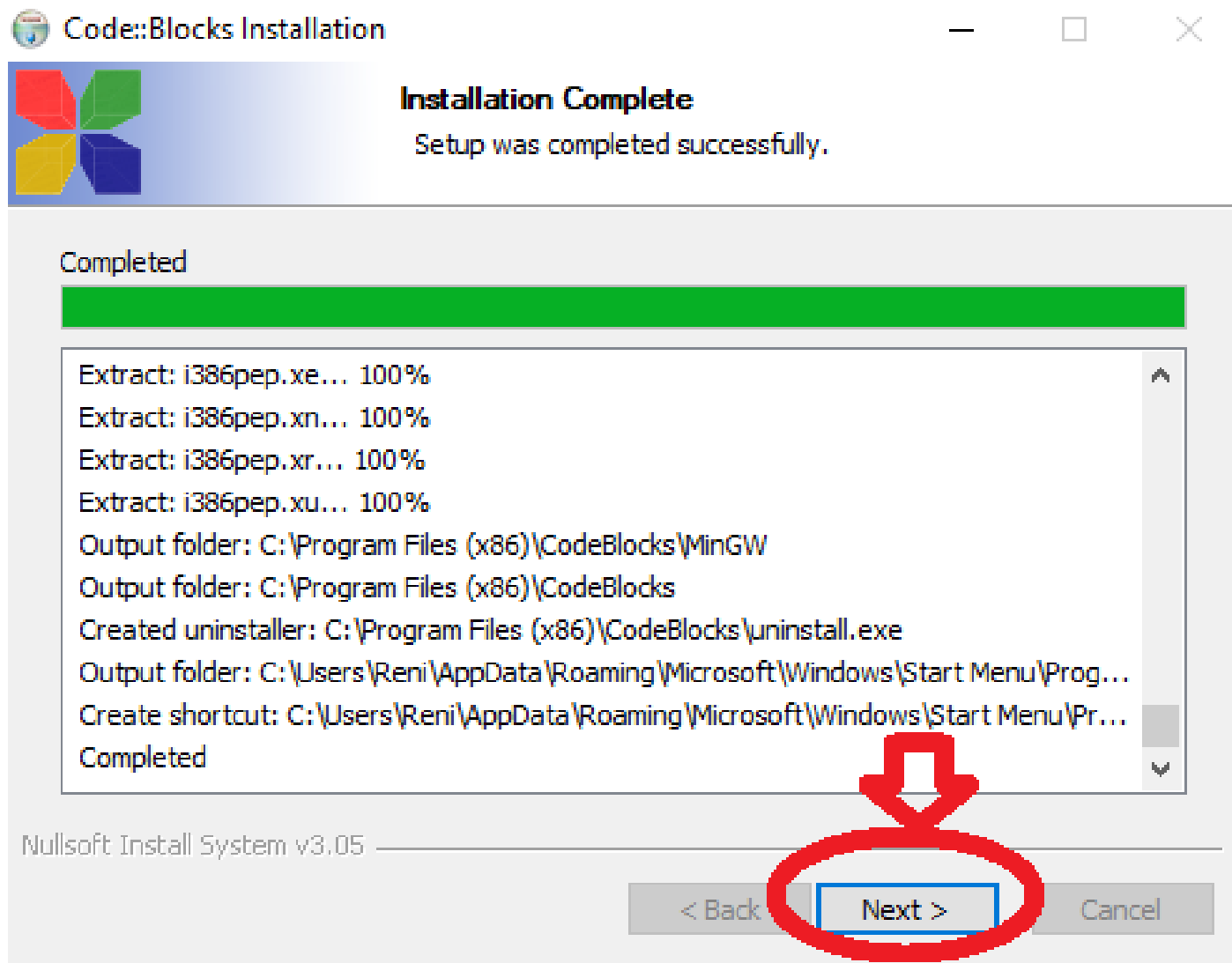


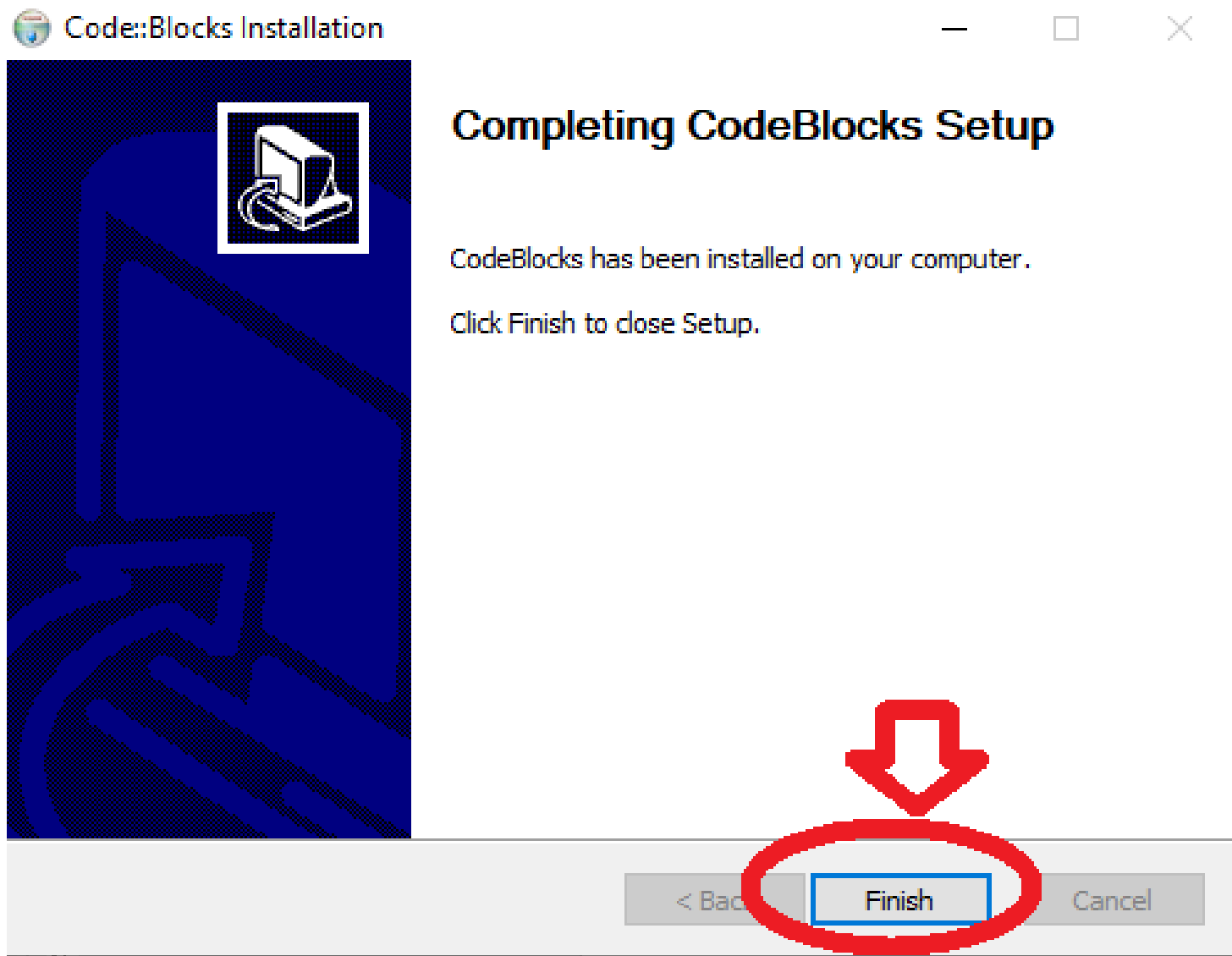


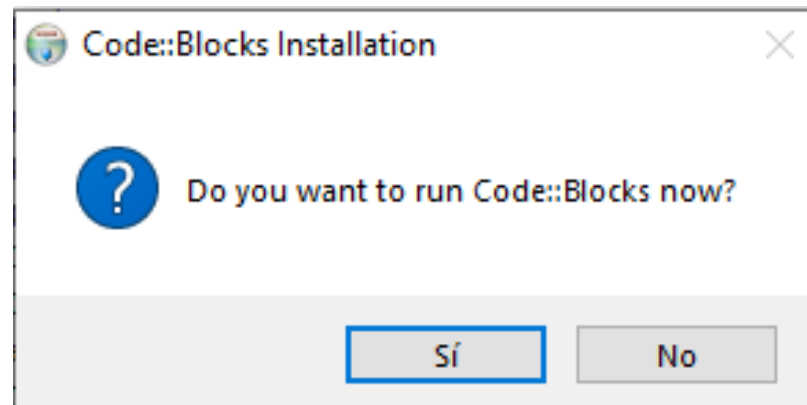












Codeblocks – Usar IDE

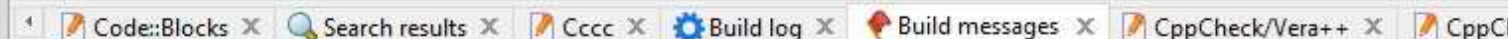
3. Ejecutamos Codeblocks



Release 20.03 rev 11983 (2020-03-12 18:24:30) gcc 8.1.0 Windows/unicode - 64 bit



Logs & others



File	Line	Message
------	------	---------

Aquí estará
nuestro
código



Code::Blocks

The open source, cross-platform IDE

[Release 20.03 rev 11983 \(2020-03-12 18:24:30\) gcc 8.1.0 Windows/unicode - 64 bit](#)



[Create a new project](#)



[Open an existing project](#)



[Tip of the Day](#)



[Visit the Code::Blocks forums](#)

[Report a bug or request a new feature](#)

Recent projects

Logs & messages

Code::Blocks x Search results x Cccc x Build log x Build messages x CppCheck/Vera++ x CppC

File	Line	Message
------	------	---------

Estructura
de nuestro
proyecto.

Aquí
veremos las
carpetas y
archivos
que forman
nuestro
proyecto.

Con
SHIFT + F2
aparece /
desaparece



Code::Blocks

The open source, cross-platform IDE

[Release 20.03 rev 11983 \(2020-03-12 18:24:30\) gcc 8.1.0 Windows/unicode - 64 bit](#)



[Create a new project](#)



[Open an existing project](#)



[Tip of the Day](#)



[Visit the Code::Blocks forums](#)

[Report a bug or request a new feature](#)

Recent projects

Logs & others

Code::Blocks

Search results

Cccc

Build log

Build messages

CppCheck/Vera++

CppC

File

Line

Message



Code::Blocks

The open source, cross-platform IDE

[Release 20.03 rev 11983 \(2020-03-12 18:24:30\) gcc 8.1.0 Windows/unicode - 64 bit](#)



[Create a new project](#)



[Open an existing project](#)



[Tip of the Day](#)



[Visit the Code::Blocks forums](#)

[Report a bug or request a new feature](#)

Recent projects

Aquí veremos los mensajes del compilador.
Con **F2** aparece y desaparece

Codeblocks – Usar IDE

4. Creamos nuestro primer proyecto

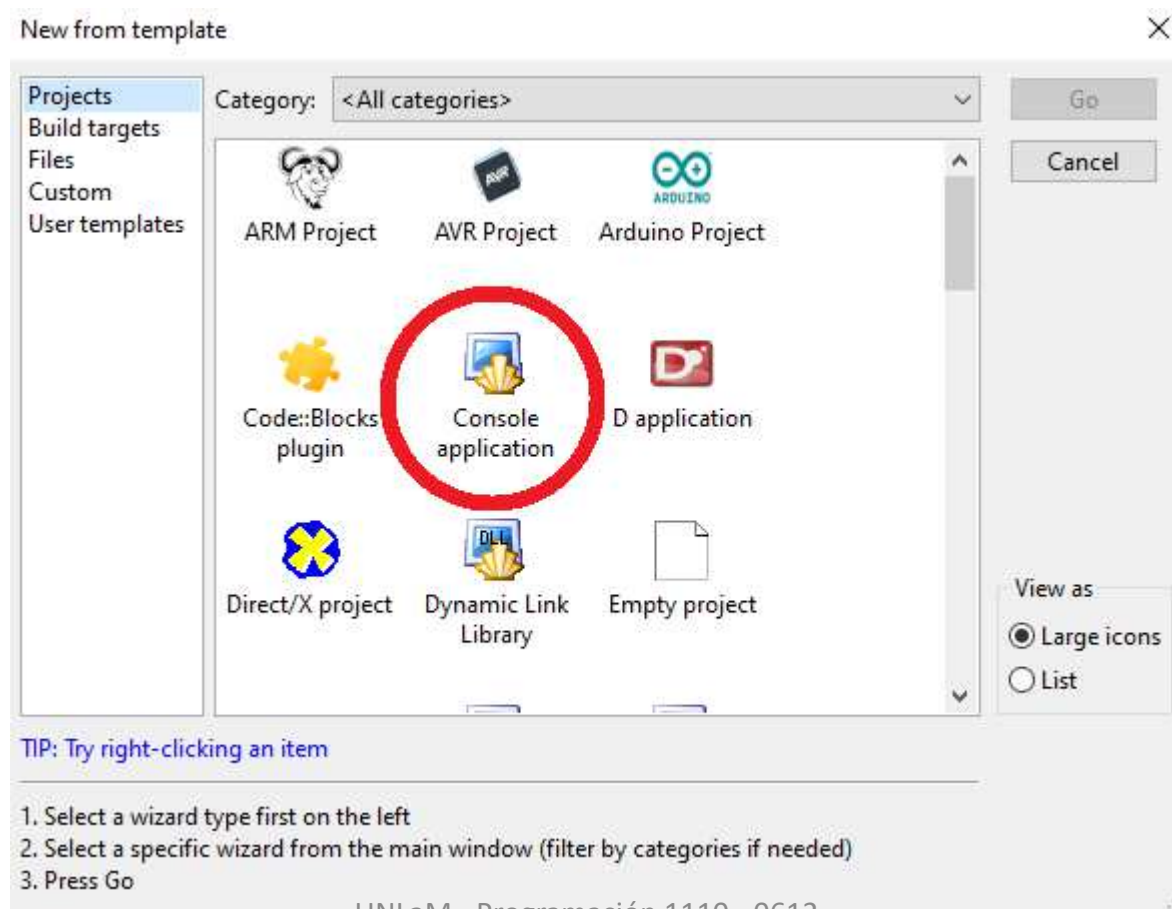
FILE → NEW → PROYECT



Para crear un proyecto:

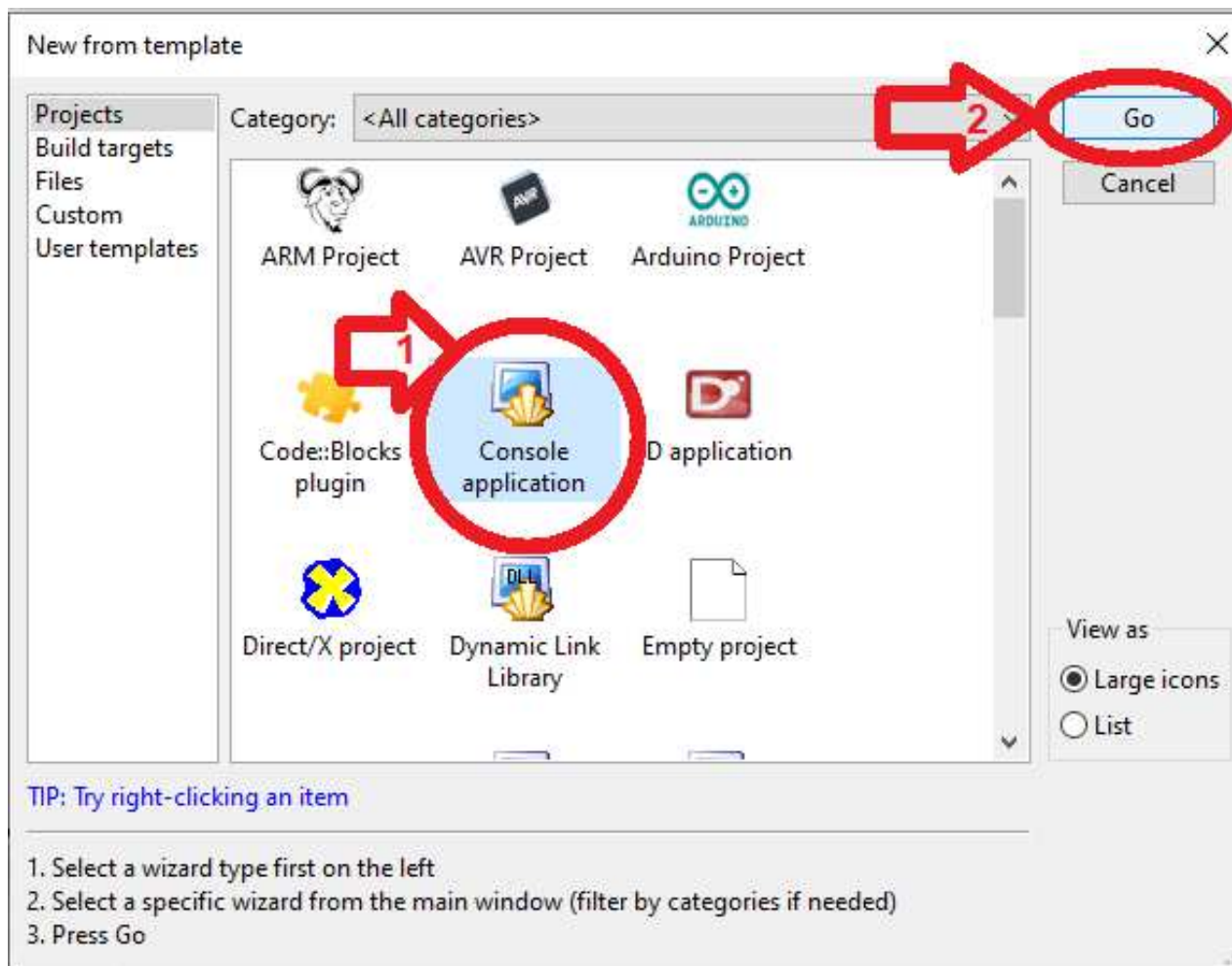
En la ventana que se abre, seleccionamos:

“Aplicación de consola”



En la ventana que se abre, seleccionamos:

1. **“Aplicación de consola”**
2. **Aceptar (Go)**



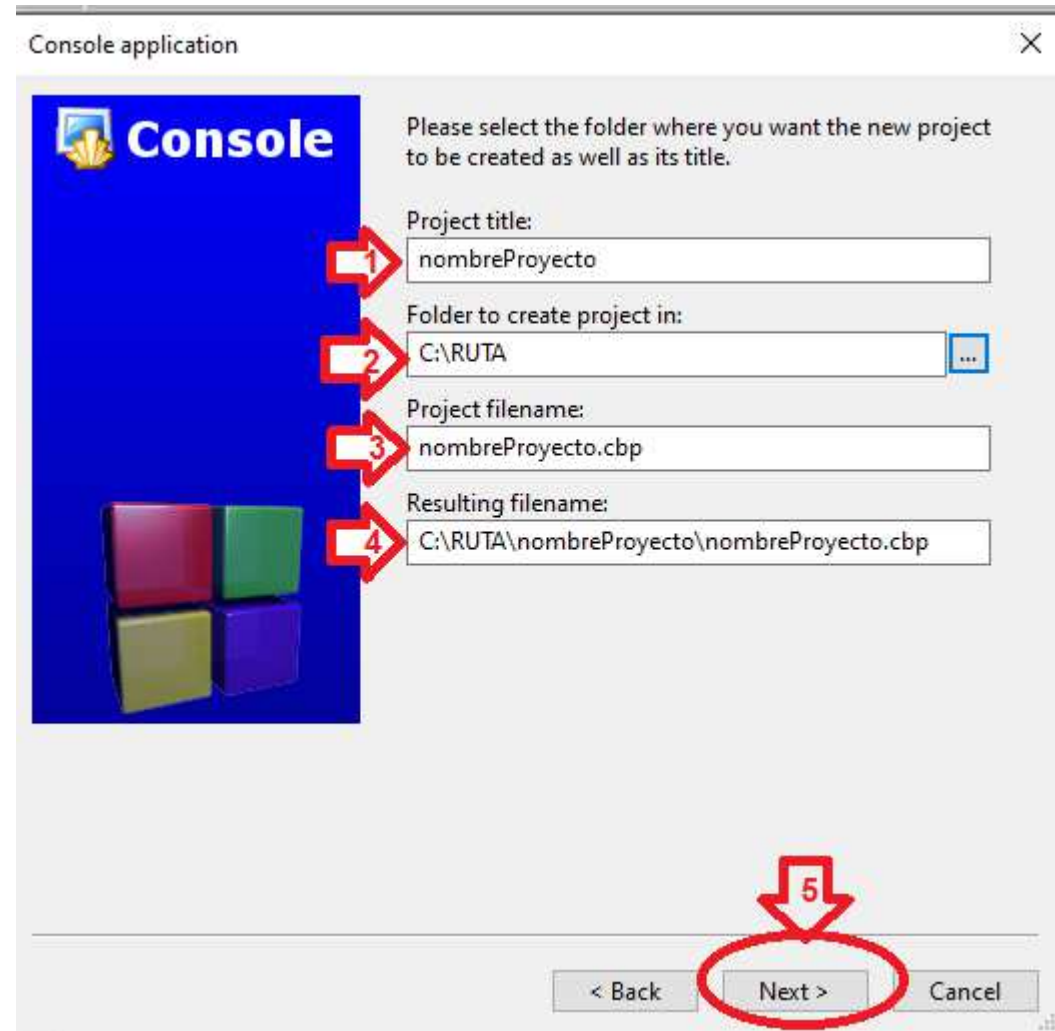
En la ventana que se abre, seleccionamos:

1. Next (Siguiente)



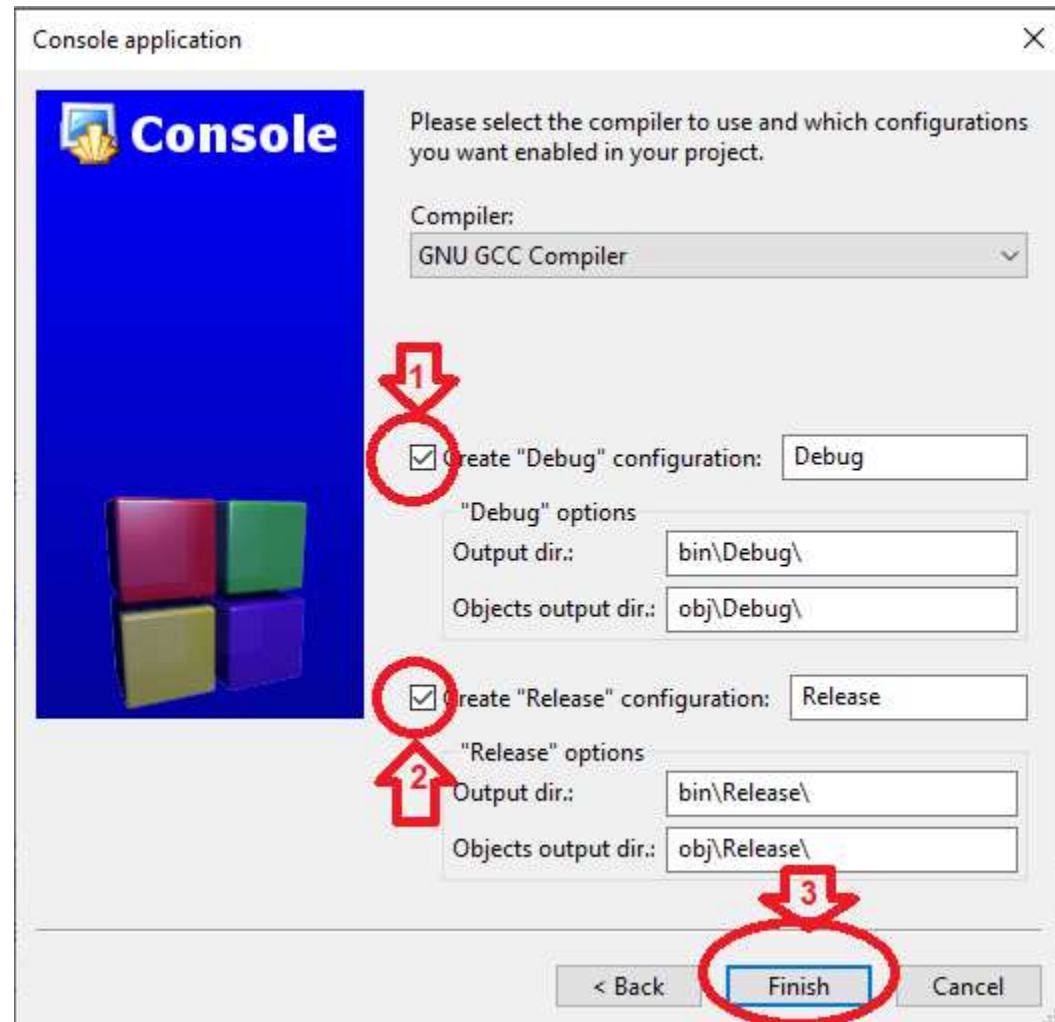
En la ventana que se abre completamos los datos del proyecto:

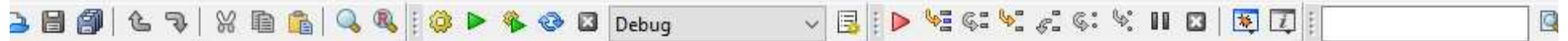
1. Carpeta del Proyecto
2. Seleccionamos la RUTA donde se guardará el proyecto.
3. Nombre del proyecto. Se completa automáticamente son el nombre de la carpeta (podemos modificarlo)
4. Se completa automáticamente (NO modificar)
5. Next (siguiente)



En la ventana que se abre nos aseguramos de colocar los tildes a:

1. Crear la versión
DEBUG
2. Crear la versión
RELEASE
3. Finish
(se crea el
proyecto)





<global>

ment

Files FSymbols Resources

orkspace

nombreProyecto

Sources

main.c

main.c

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      printf("Hello world!\n");
7      return 0;
8  }
9

```

Logs & others

Code::Blocks Search results Cccc Build log Build messages CppCheck/Vera++ CppCl

File Line Message

UNLaM - Programación 1110 - 0612 38

R. Guatelli

A\nombreProyect...

C/C++

Windows (CR+LF)

WINDOWS-1252

Line 1, Col 1, Pos 0

Insert

Read/Write

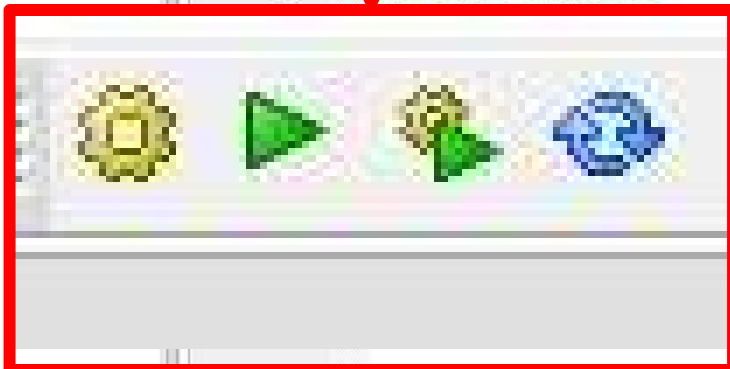
default

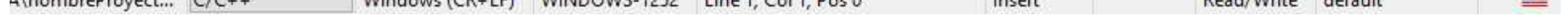
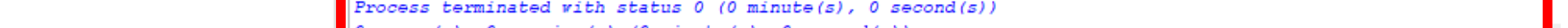
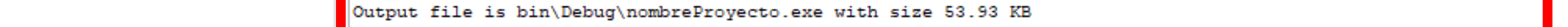
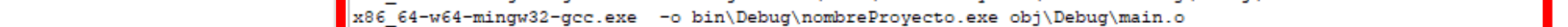
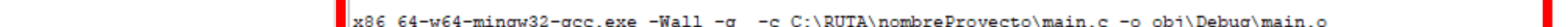
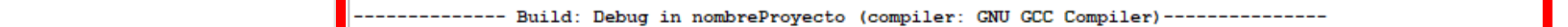
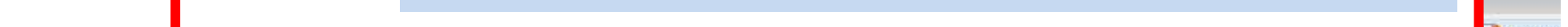
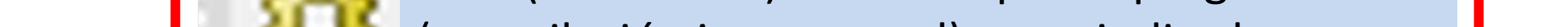
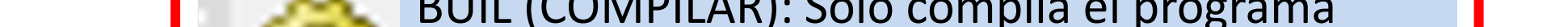
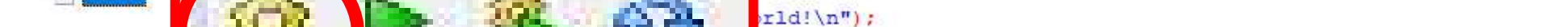
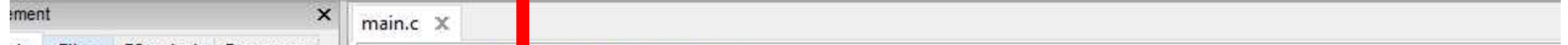


Codeblocks – Usar IDE

5. Compilamos nuestro proyecto, para asegurarnos que todo quedo bien instalado.

```
1 #include <stdio.h>
2 #include <stdlib.h>
```





BUIL (COMPILAR): Solo compila el programa (compilación incremental) y nos indica los errores. Debemos tratar siempre de obtener 0 errores y 0 advertencias.

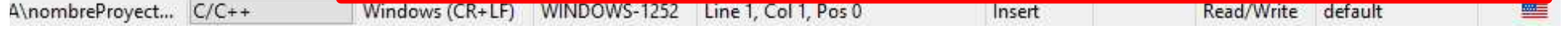
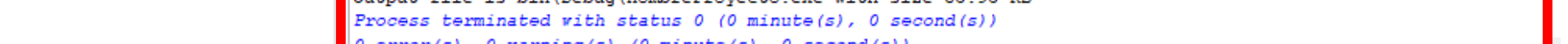
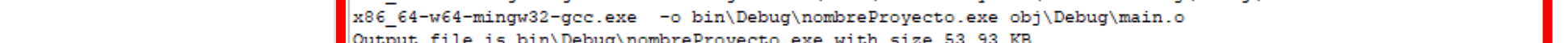
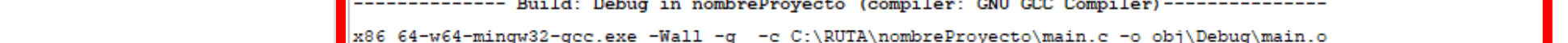
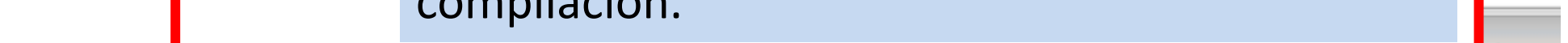
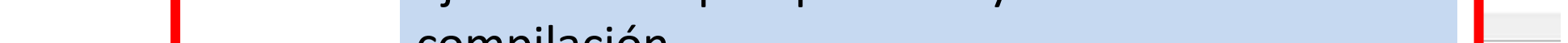
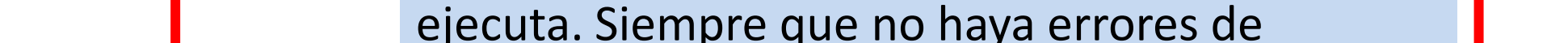
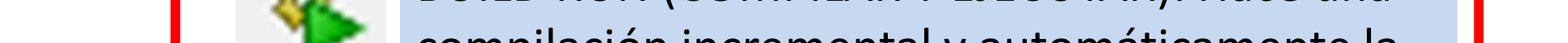
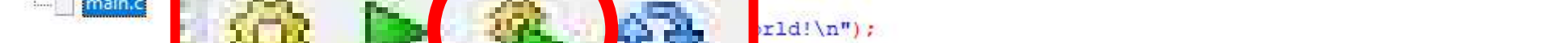
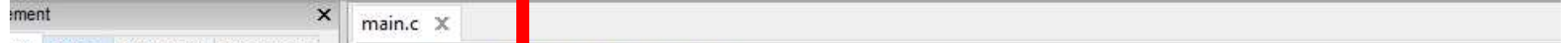
```
----- Build: Debug in nombreProyecto (compiler: GNU GCC Compiler)-----
x86_64-w64-mingw32-gcc.exe -Wall -g -c C:\RUTA\nombreProyecto\main.c -o obj\Debug\main.o
x86_64-w64-mingw32-gcc.exe -o bin\Debug\nombreProyecto.exe obj\Debug\main.o
Output file is bin\Debug\nombreProyecto.exe with size 53.93 KB
Process terminated with status 0 (0 minute(s), 0 second(s))
0 error(s), 0 warning(s) (0 minute(s), 0 second(s))
```

Process terminated with status 0 (0 minute(s), 0 second(s))
0 error(s), 0 warning(s) (0 minute(s), 0 second(s))



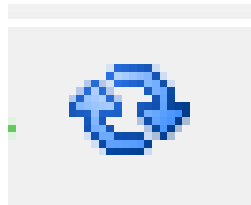
RUN (EJECUTAR): ejecuta la última versión compilada. Si se hicieron modificaciones y no se compiló, estas modificaciones no son tenidas en cuenta durante la ejecución. Para que se ejecuten DEBE compilar o reconstruir el proyecto.

```
----- Build: Debug in nombreProyecto (compiler: GNU GCC Compiler)-----
x86_64-w64-mingw32-gcc.exe -Wall -g -c C:\RUTA\nombreProyecto\main.c -o obj\Debug\main.o
x86_64-w64-mingw32-gcc.exe -o bin\Debug\nombreProyecto.exe obj\Debug\main.o
Output file is bin\Debug\nombreProyecto.exe with size 53.93 KB
Process terminated with status 0 (0 minute(s), 0 second(s))
0 error(s), 0 warning(s) (0 minute(s), 0 second(s))
```



BUILD RUN (COMPILAR Y EJECUTAR): Hace una compilación incremental y automáticamente la ejecuta. Siempre que no haya errores de compilación.

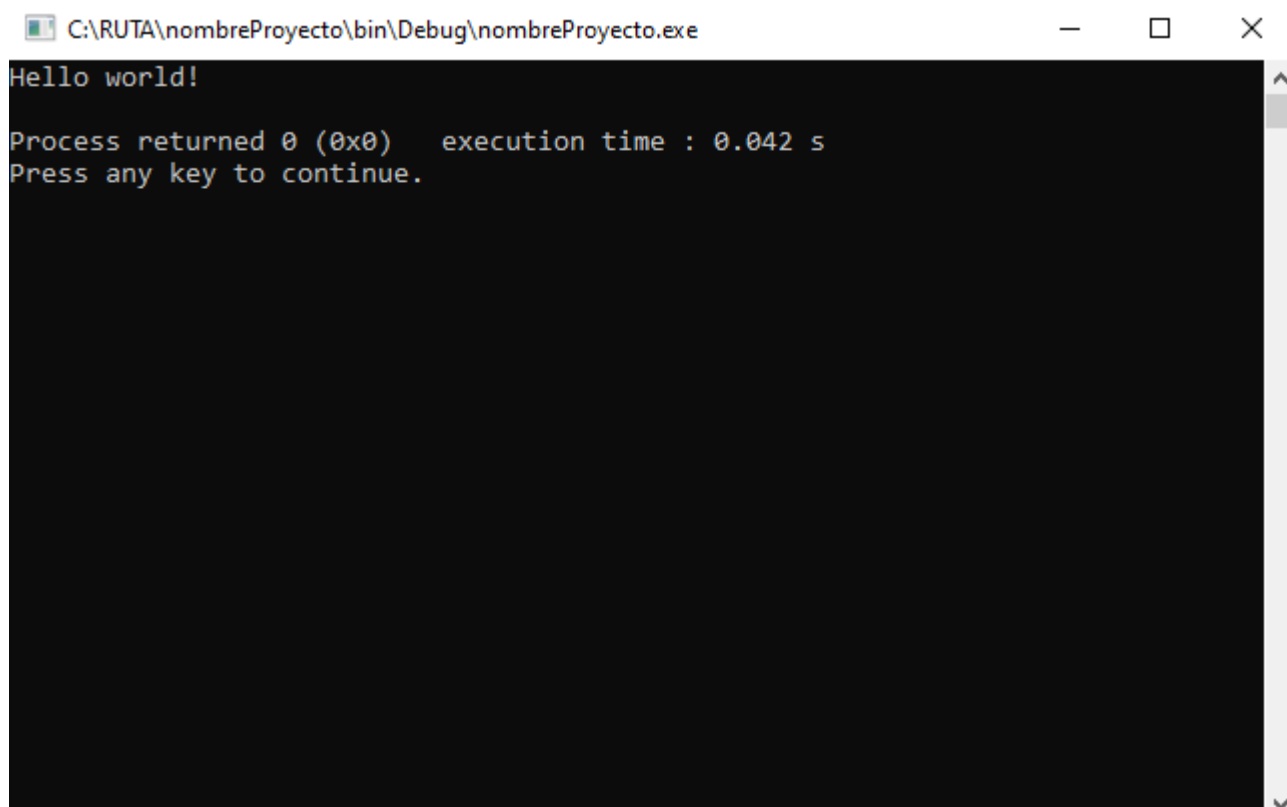
```
----- Build: Debug in nombreProyecto (compiler: GNU GCC Compiler)-----
x86_64-w64-mingw32-gcc.exe -Wall -g -c C:\RUTA\nombreProyecto\main.c -o obj\Debug\main.o
x86_64-w64-mingw32-gcc.exe -o bin\Debug\nombreProyecto.exe obj\Debug\main.o
Output file is bin\Debug\nombreProyecto.exe with size 53.93 KB
Process terminated with status 0 (0 minute(s), 0 second(s))
0 error(s), 0 warning(s) (0 minute(s), 0 second(s))
```



REBUILD (RECONSTRUIR): Elimina los elementos de compilaciones anteriores y vuelve a compilar la solución completa.

```
----- Build: Debug in nombreProyecto (compiler: GNU GCC Compiler)-----
x86_64-w64-mingw32-gcc.exe -Wall -g -c C:\RUTA\nombreProyecto\main.c -o obj\Debug\main.o
x86_64-w64-mingw32-gcc.exe -o bin\Debug\nombreProyecto.exe obj\Debug\main.o
Output file is bin\Debug\nombreProyecto.exe with size 53.93 KB
Process terminated with status 0 (0 minute(s), 0 second(s))
0 error(s), 0 warning(s) (0 minute(s), 0 second(s))
```

Si Codeblocks se instaló correctamente y compilamos y ejecutamos el main que generó por defecto, veremos la siguiente pantalla:



```
C:\RUTA\nombreProyecto\bin\Debug\nombreProyecto.exe
Hello world!
Process returned 0 (0x0) execution time : 0.042 s
Press any key to continue.
```

Para cerrar la ventana de la ejecución presionamos “**cualquier tecla**” y la ventana se cerrará.

Evitar cerrar la ventana haciendo click en “X (cerrar ventana)”

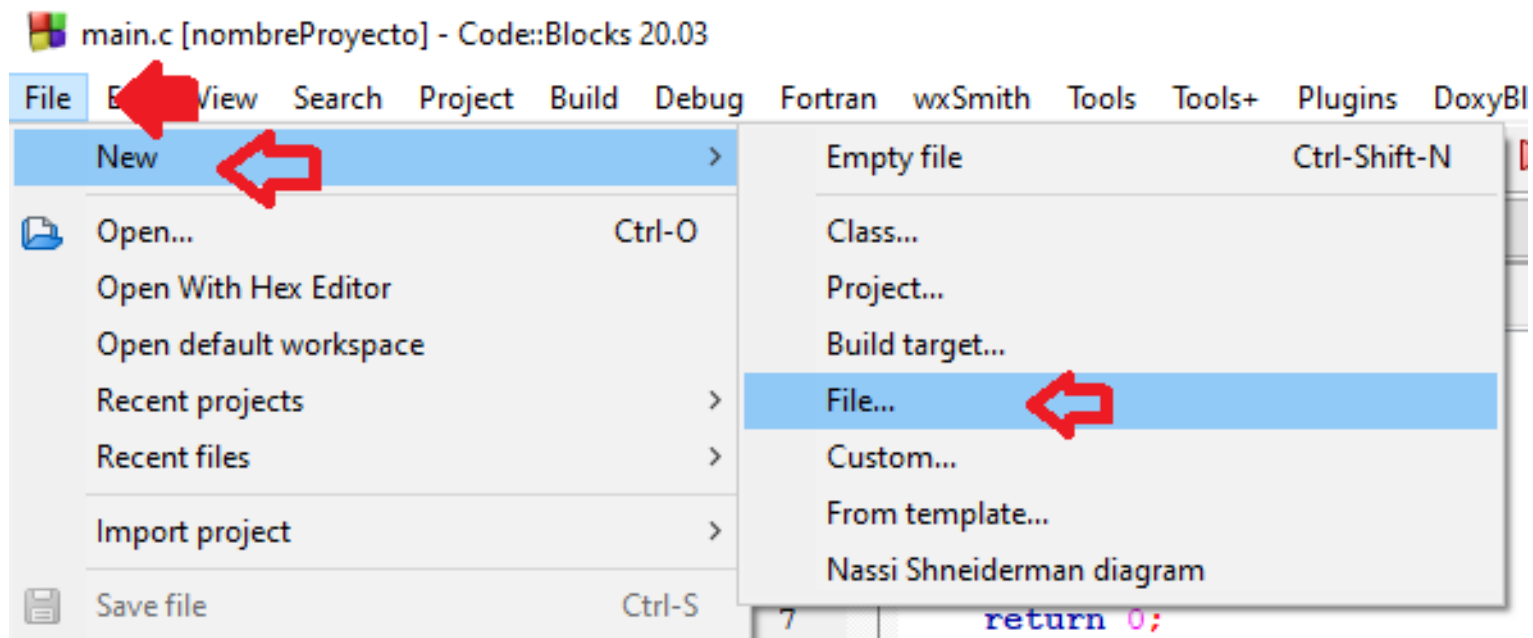


Codeblocks – Usar IDE

6. Ya estamos seguros que nuestro Codeblocks quedó bien instalado. Vamos a agregar los archivos de cabecera y de funciones a nuestro proyecto.

Para agregar un archivo de cabecera .h:

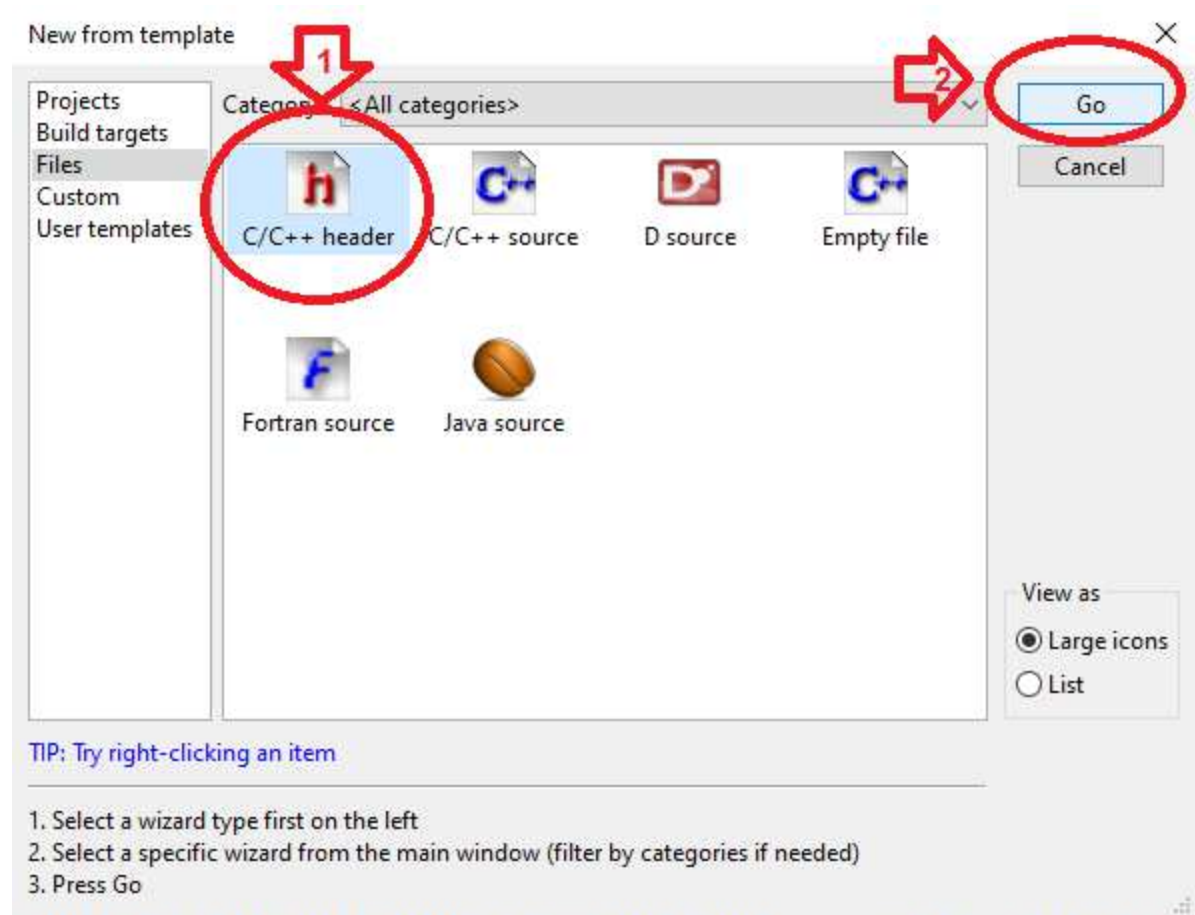
FILE → NEW → File



Para agregar un archivo de cabecera .h:

En la ventana que se abre, seleccionamos:

1. **C/C++ header**
2. Go (siguiente)



En la ventana que se abre, seleccionamos:

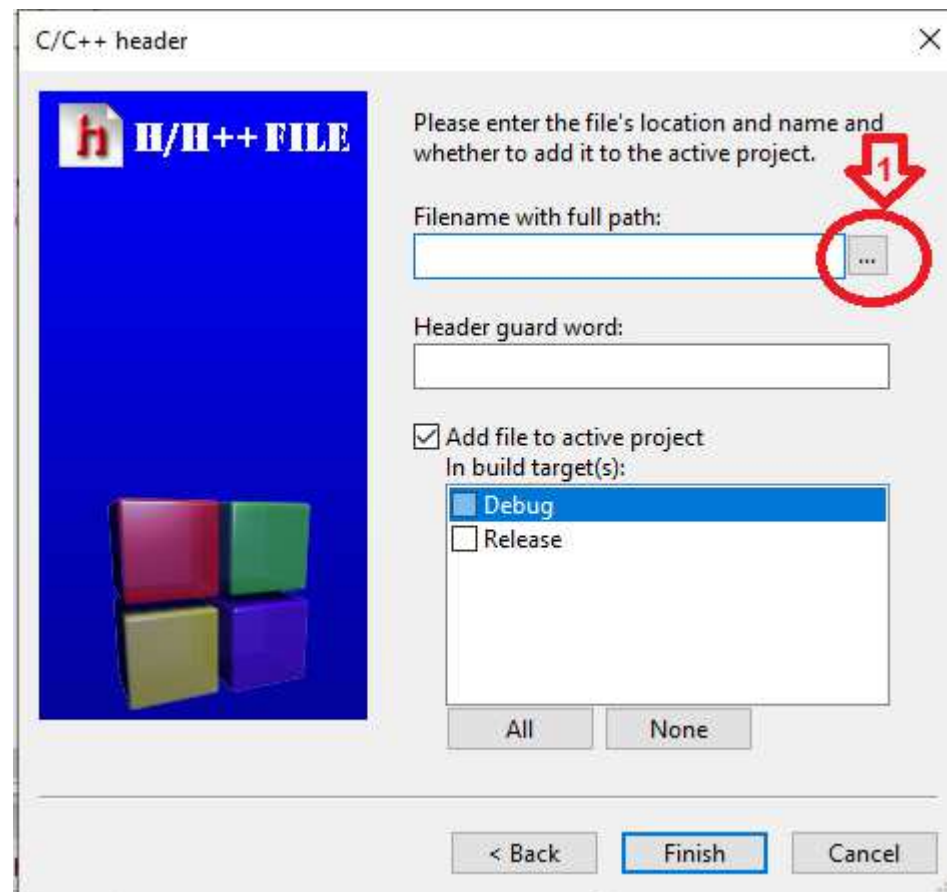
1. Next (Siguiente)



En la ventana que se abre, seleccionamos:

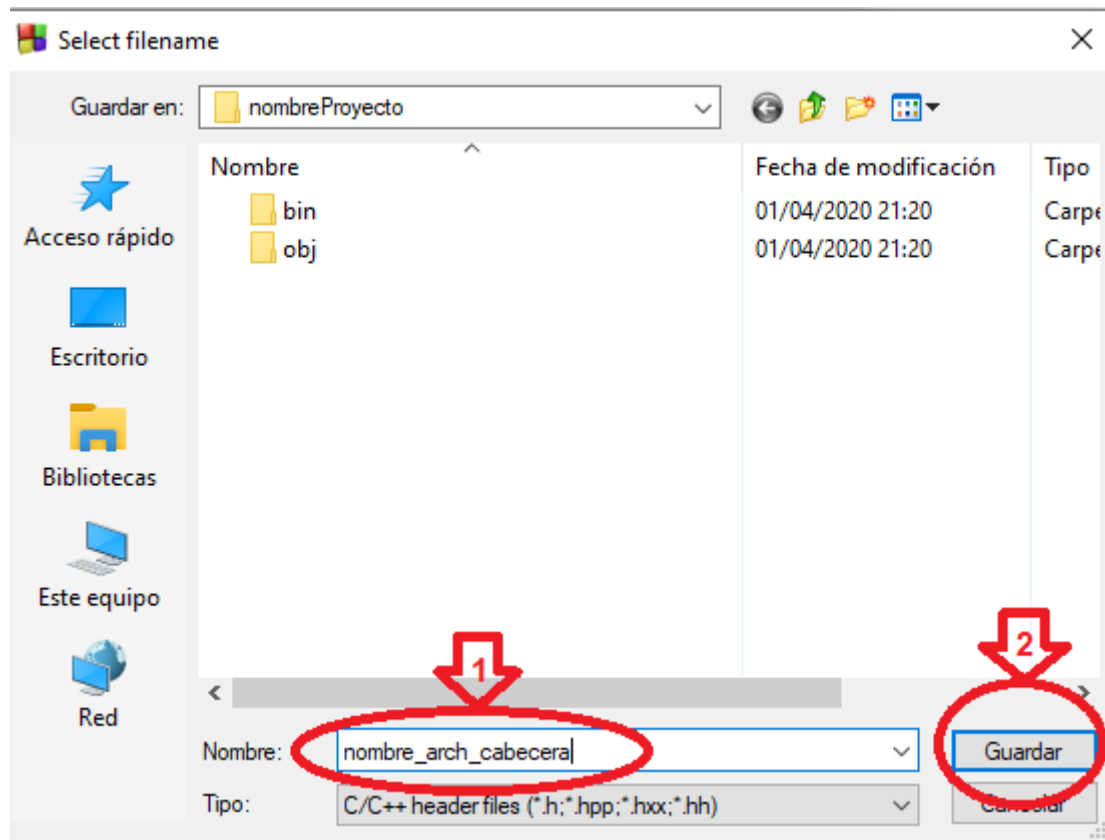
1. ...

Para completar la ruta y el nombre de nuestro archivo de cabecera



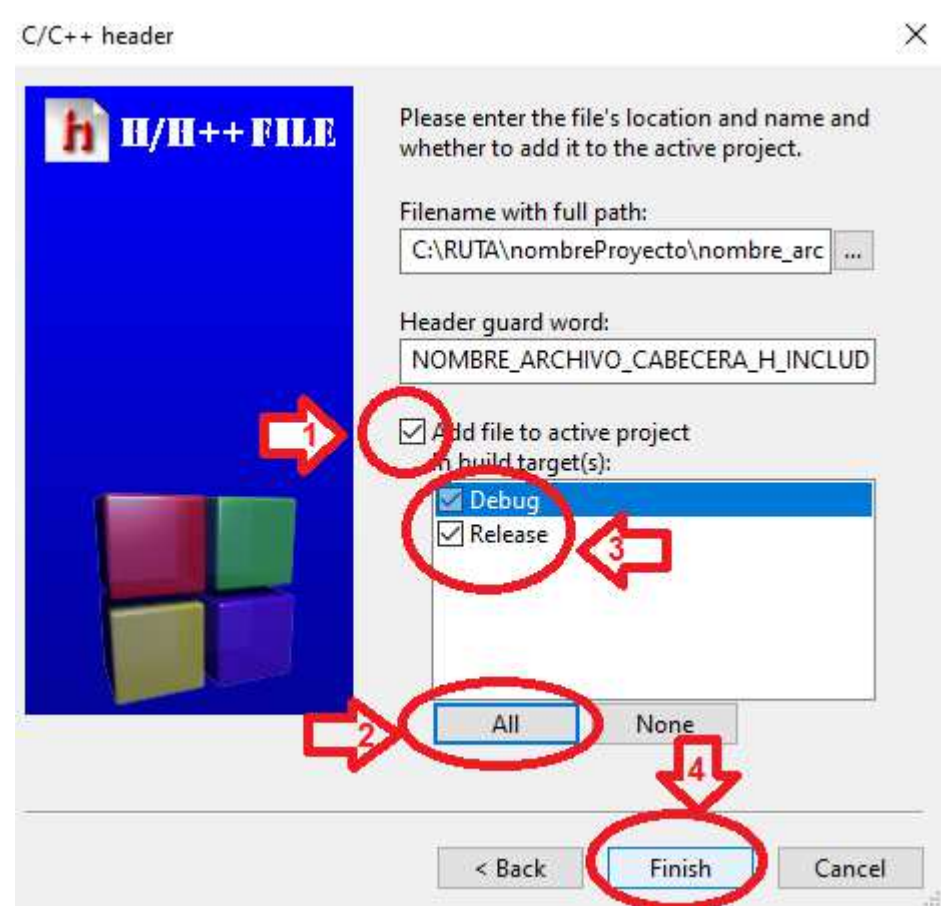
Al hacer click en “...” se abre la siguiente ventana:

1. Colocamos el nombre que tendrá nuestro nuevo archivo de cabecera.
2. Guardar

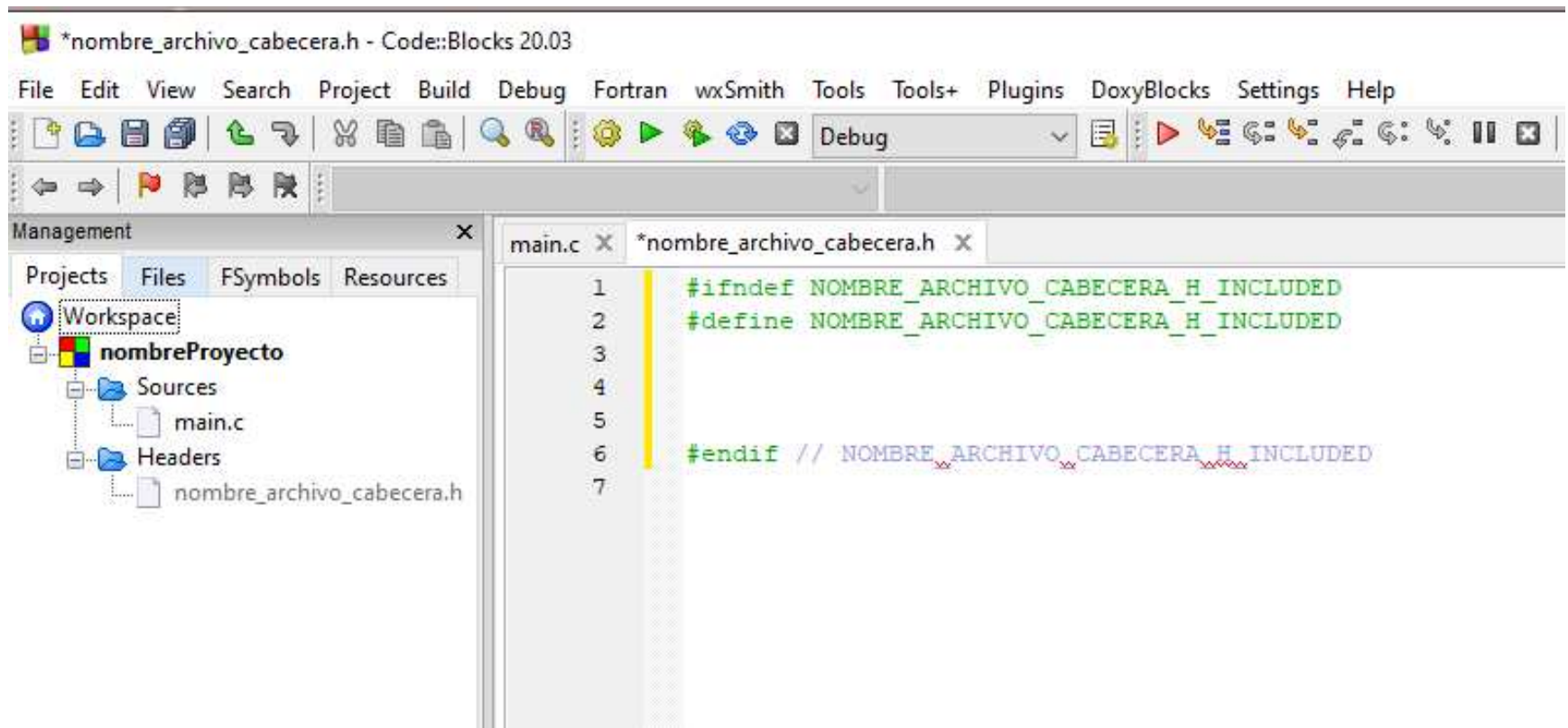


Volvemos a la ventana anterior, donde aparece el nombre y la ruta que ingresamos en la ventana anterior:

1. Nos aseguramos de colocar el tilde para que el archivo se añada al proyecto actual
2. Hacemos click en “All”
3. Al hacer click en “All” coloca los tildes en “Debug” y “Release”, va a crear las dos versiones: final y para debug.
4. Finish (crear el archivo)



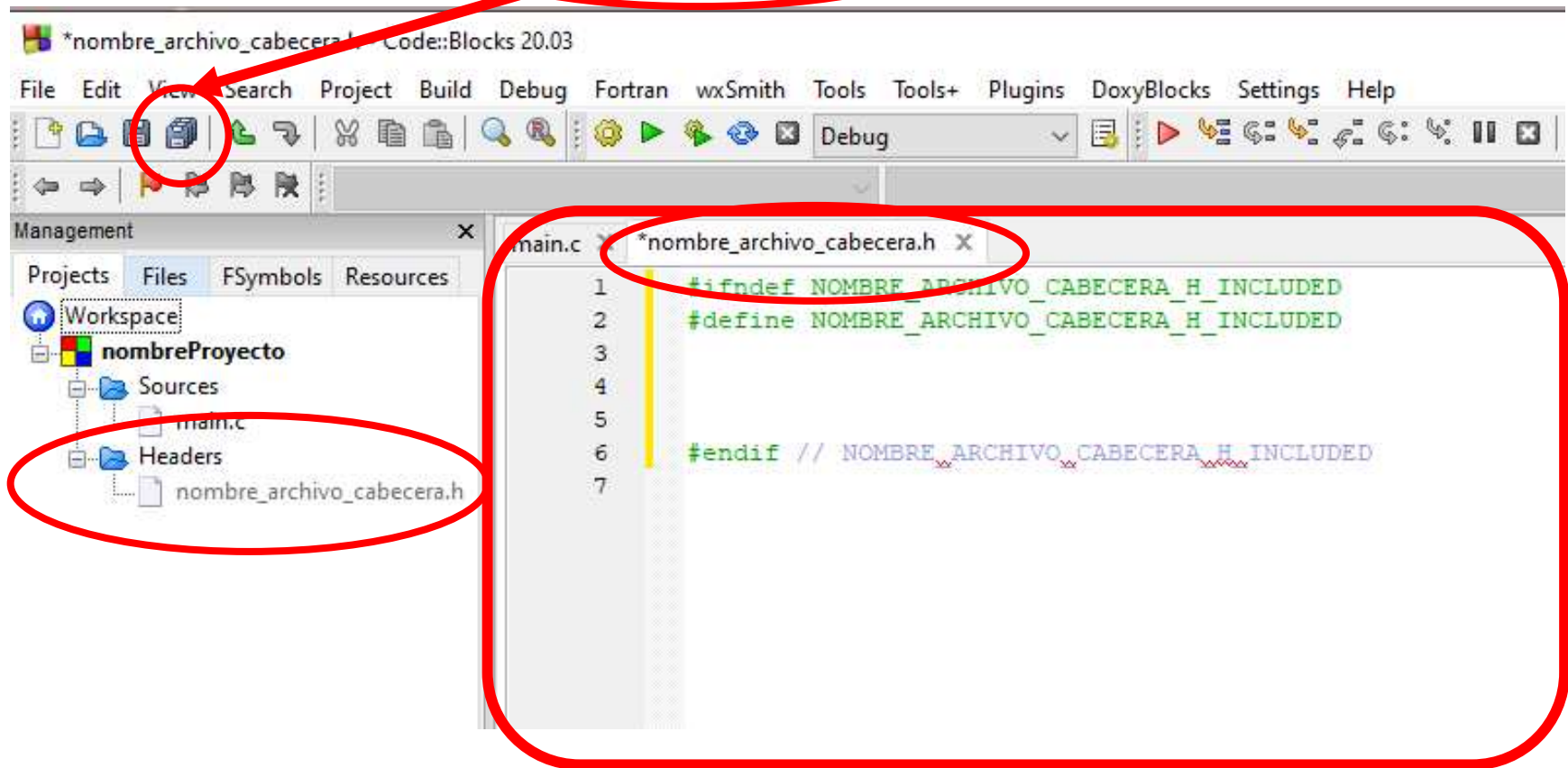
Volvemos al IDE:



Volvemos al IDE:

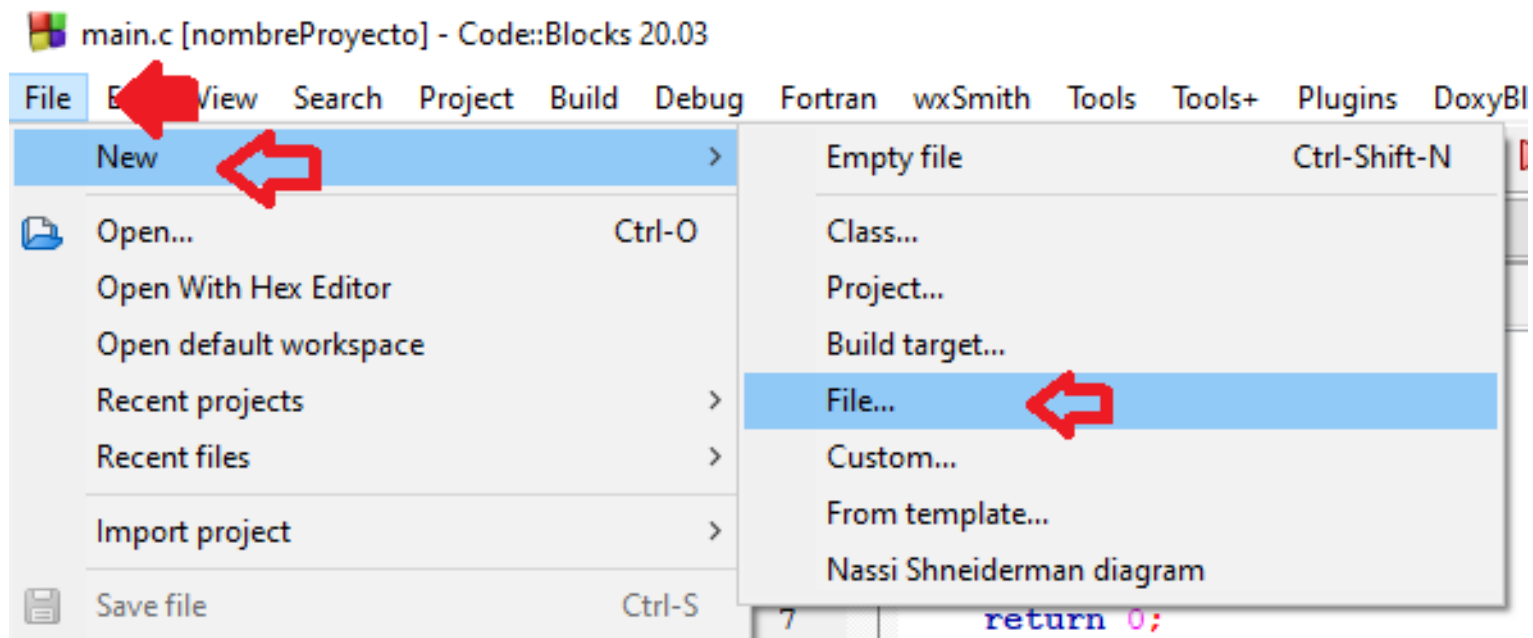
Vemos que agregó el archivo de cabecera al proyecto.

¡No nos olvidemos de guardar el proyecto!



Para agregar un archivo de funciones .c:

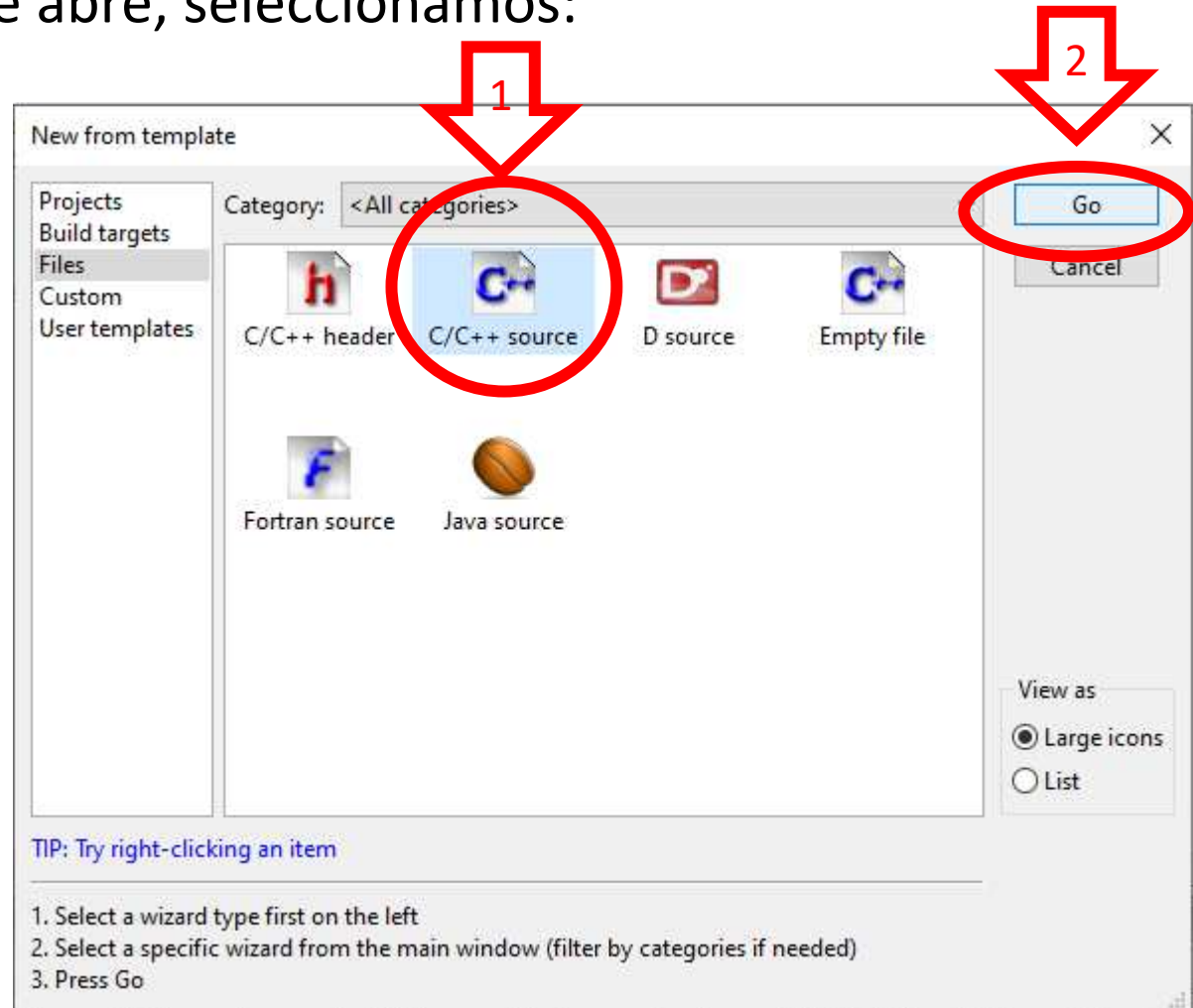
FILE → NEW → File



Para agregar un archivo de funciones .c:

En la ventana que se abre, seleccionamos:

1. **C/C++ source**
2. Go (siguiente)



En la ventana que se abre, seleccionamos:

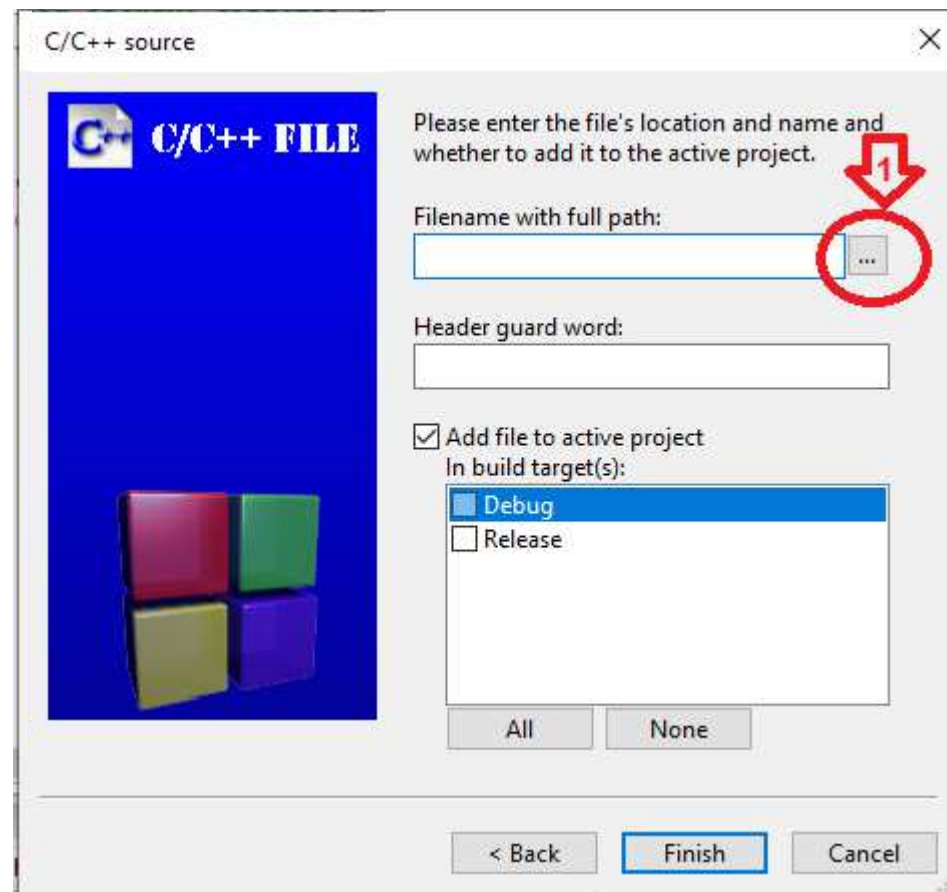
1. Next (Siguiente)



En la ventana que se abre, seleccionamos:

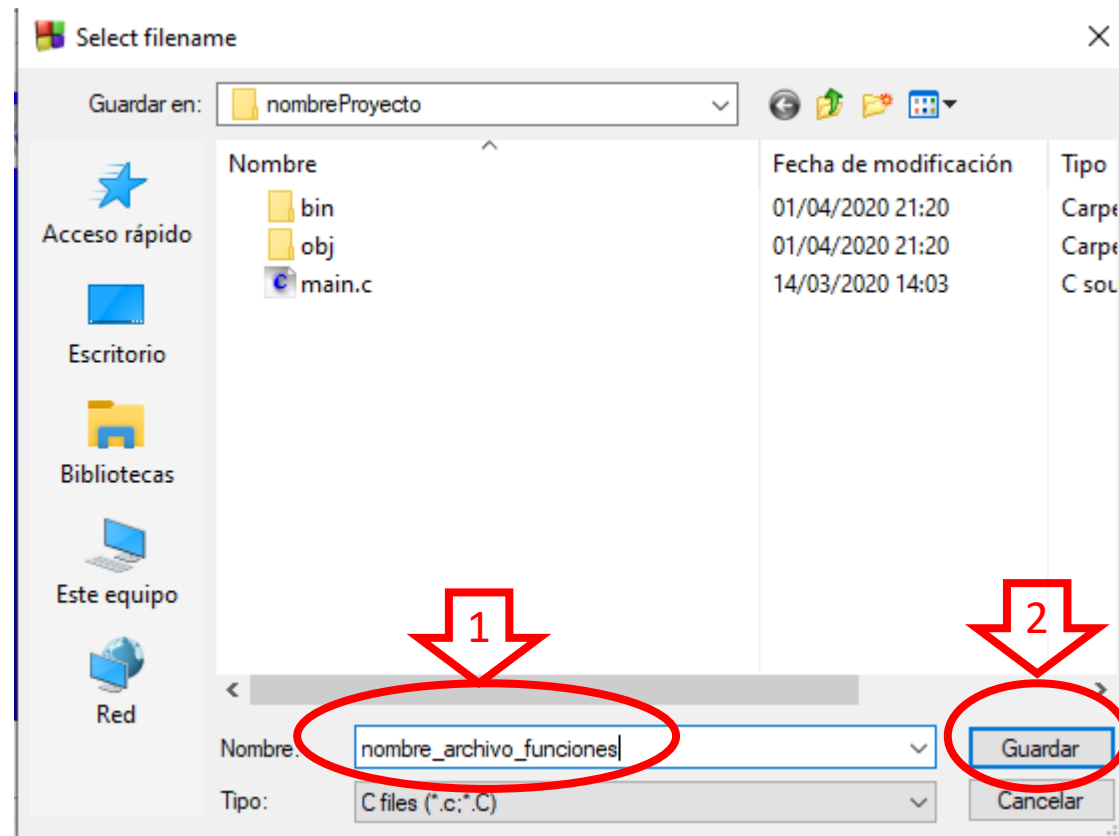
1. ...

Para completar la ruta y el nombre de nuestro archivo



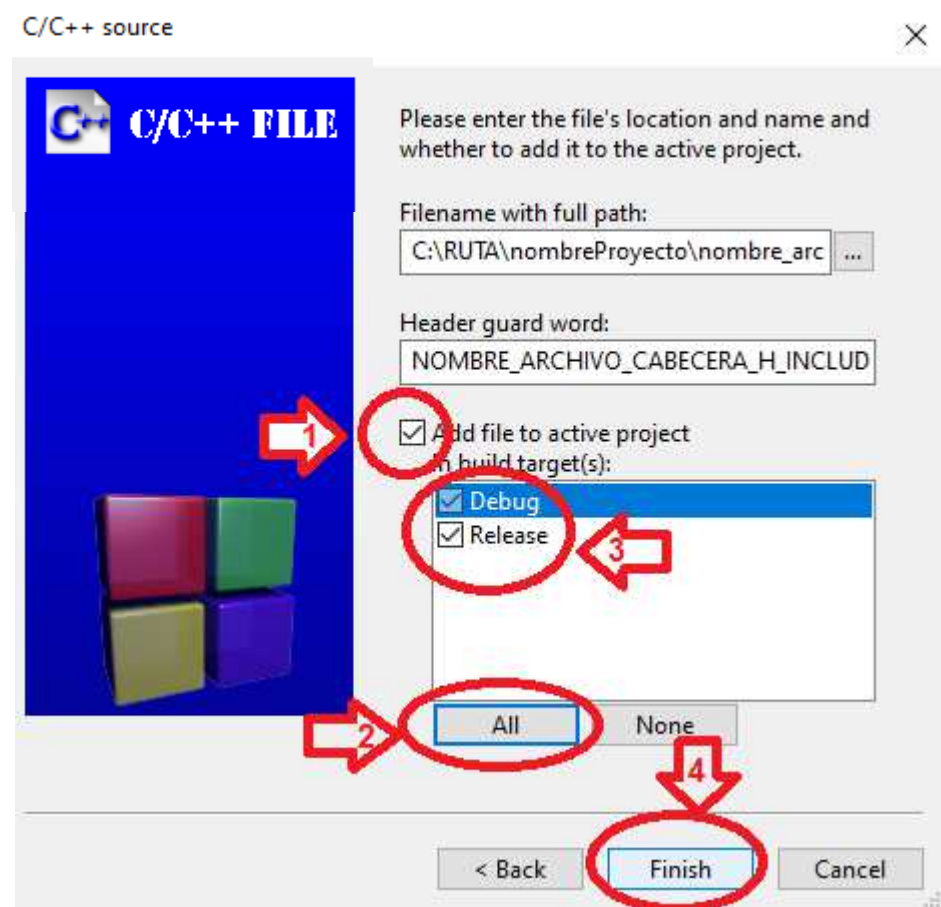
Al hacer click en “...” se abre la siguiente ventana:

1. Colocamos el nombre que tendrá nuestro nuevo archivo.
2. Guardar



Volvemos a la ventana anterior, donde aparece el nombre y la ruta que ingresamos en la ventana anterior:

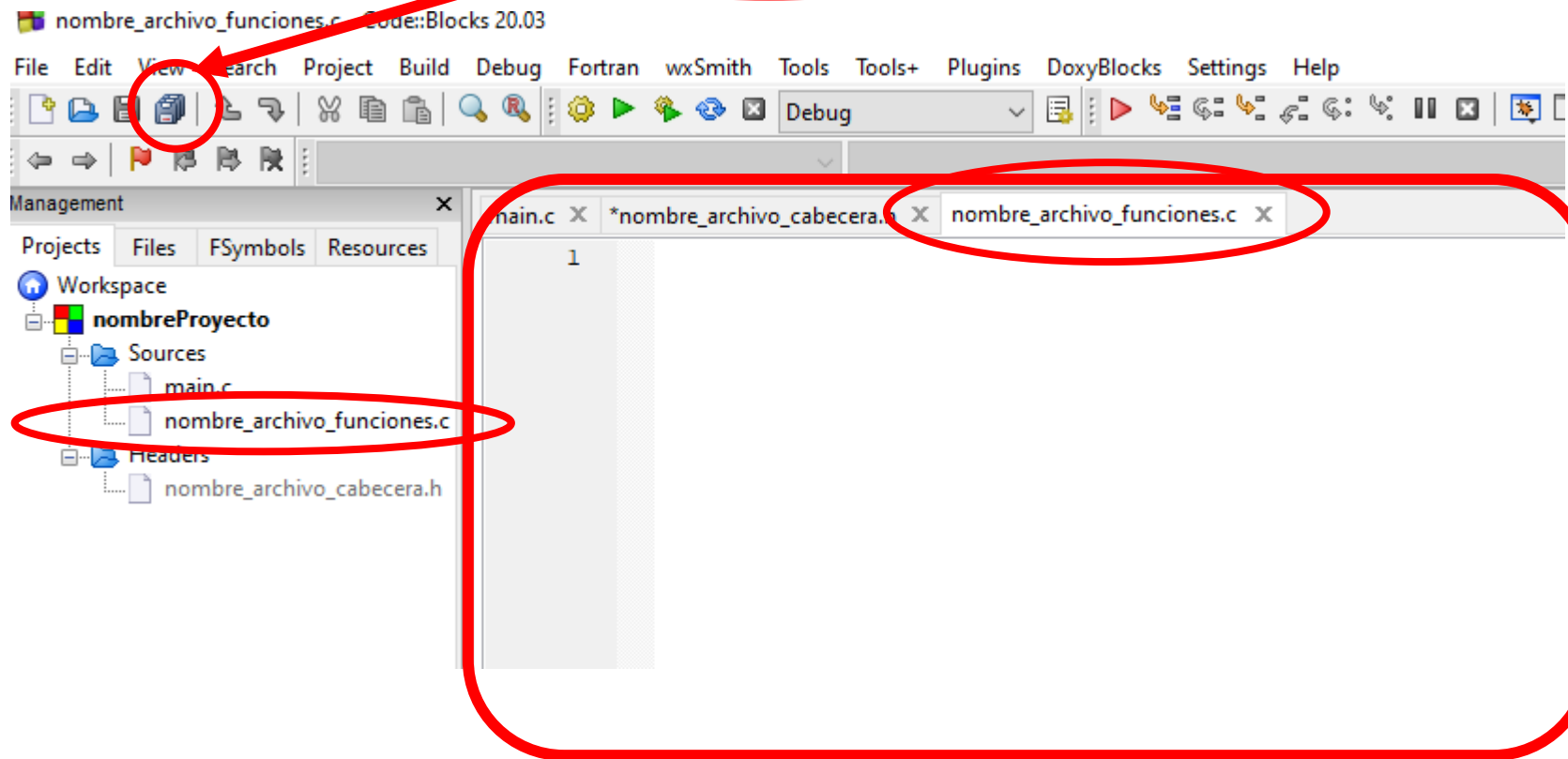
1. Nos aseguramos de colocar el tilde para que el archivo se añada al proyecto actual
2. Hacemos click en “All”
3. Al hacer click en “All” coloca los tildes en “Debug” y “Release”, va a crear las dos versiones: final y para debug.
4. Finish (crear el archivo)



Volvemos al IDE:

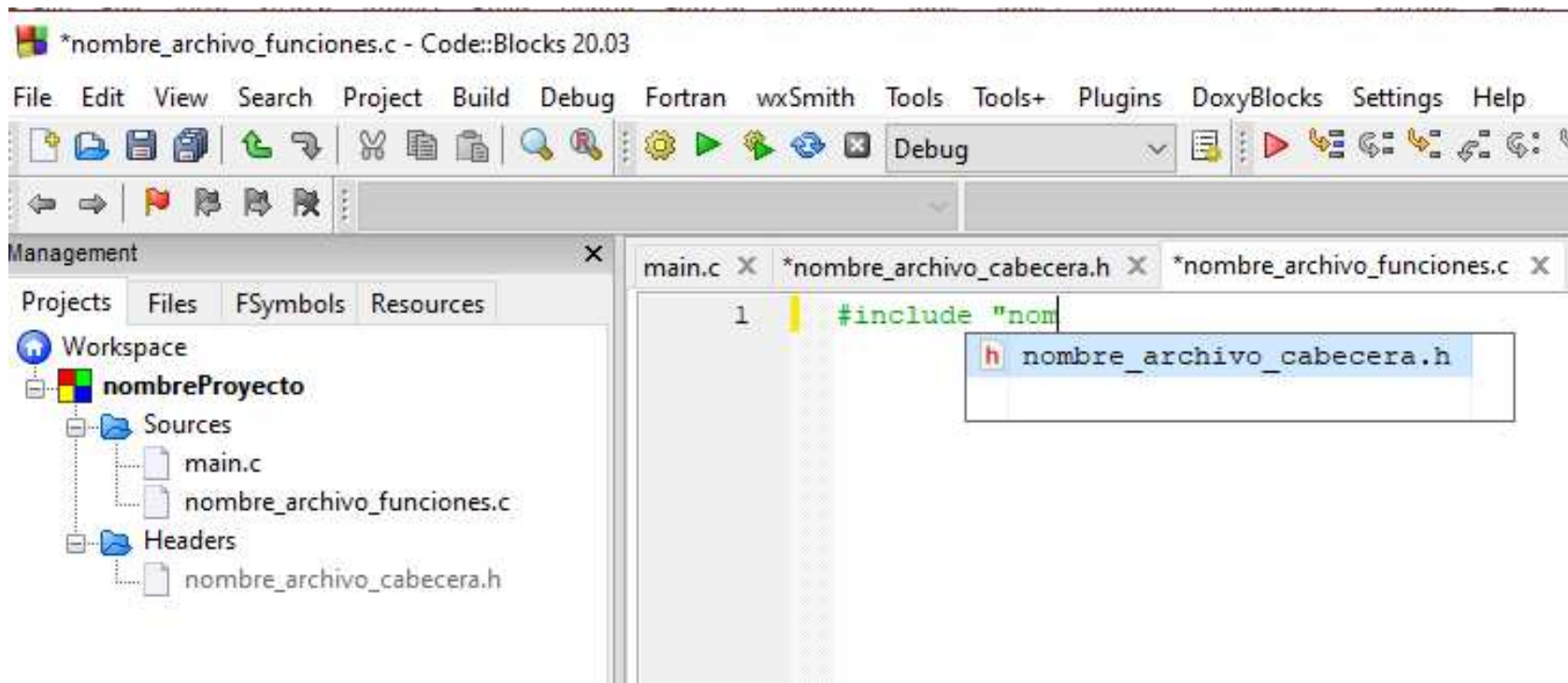
Vemos que agregó el archivo de cabecera al proyecto.

No nos olvidemos de guardar el proyecto!

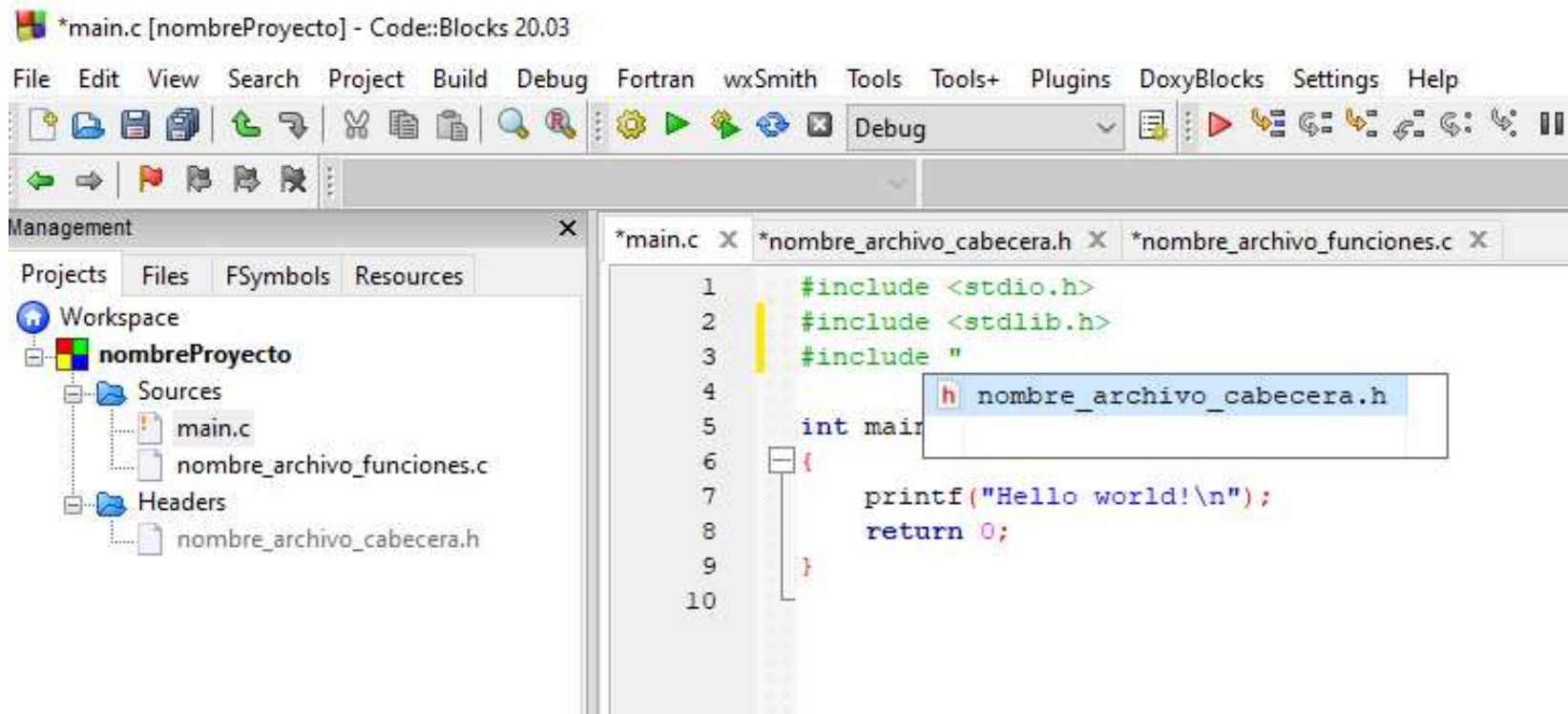


- El archivo de **cabecera.h** contiene las bibliotecas, estructuras de datos y prototipos de las funciones necesarias para su funcionamiento.
- El archivo de **funciones.c** contendrá el desarrollo de todas las funciones que esté implementando, o sea las declaradas en el archivo “.h” correspondiente.
- Para que los archivos puedan acceder a las funciones que implementan debemos “presentarlos” para ello, **tanto el main como funciones.c deben incluir al archivo de cabecera.h**

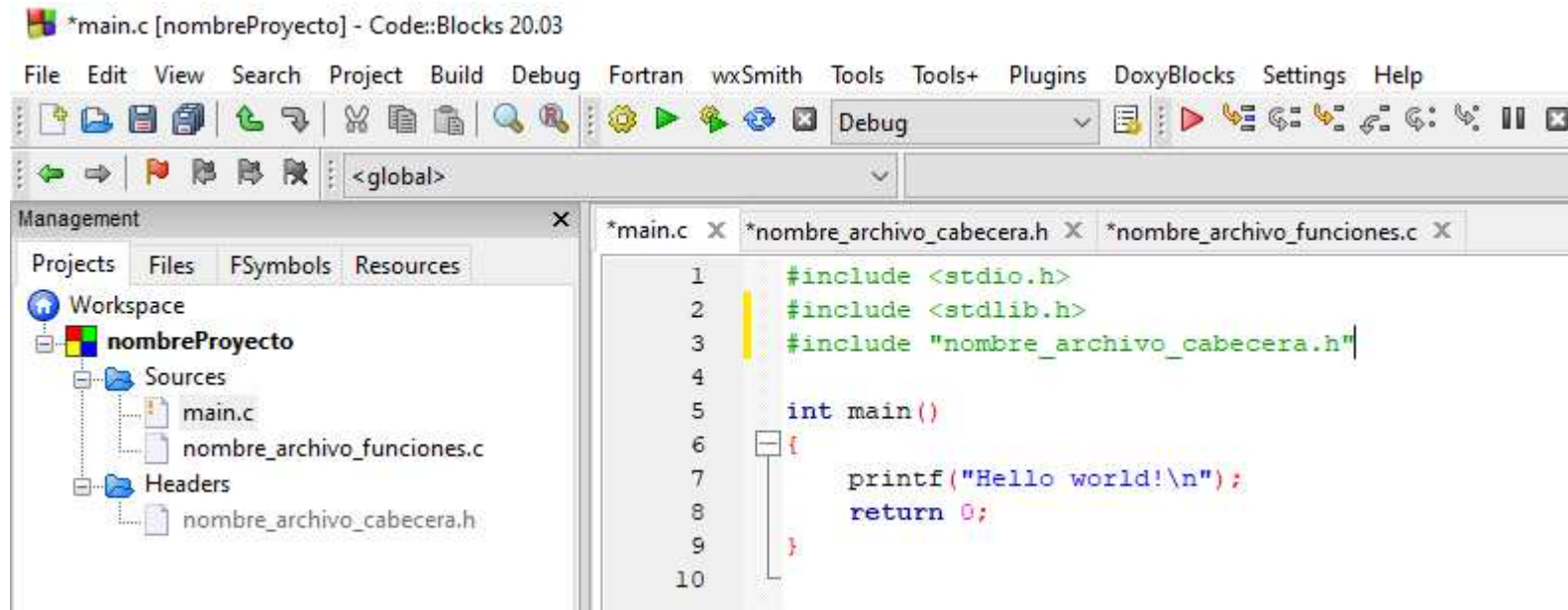
Incluimos en el archivo de funciones.c al archivo “cabecera.h”



Incluimos en el archivo main.c el archivo “cabecera.h”



Incluimos en el archivo main.c el archivo “cabecera.h”



Nota:

Recuerde que los `#include <...>` indican que los archivos se encuentran en la ruta de instalación del compilador.

Mientras que `#include "..."` indica que el archivo se encuentra en la ruta del proyecto actual o la ruta que se especifique.

Para mejorar nuestros Proyectos:

Para que nuestros proyectos queden más ordenados, disminuir el acoplamiento y facilitar su reutilización, vamos a dividir nuestro proyecto en varios programas, cada uno referido a un área temática. Cada programa va a constar de un archivo de cabecera (.h) y un archivo (.c) con el desarrollo de las funciones. Incluso el programa principal “main” se implementará con dos archivos “main.h” y “main.c”

Cada archivo de cabecera (.h) contendrá las bibliotecas, tipos de datos, declaraciones de funciones que necesite y/o implemente. Cada programa debe manejar solo la información que le sea útil.

Actividad Propuesta

Por ejemplo si nuestro proyecto maneja cálculos y funciones de texto, lo dividiríamos en 3 programas. El programa principal, el programa que maneja la parte de cálculos y el que maneja las funciones de texto.

El proyecto estaría formado por los siguientes archivos:

- main.h
- main.c
- calculo.h
- calculo.c
- cadena.h
- cadena.c

Cree un proyecto que represente la situación planteada.

¿Dónde colocaría los include de los archivos de cabecera?

Recuerde que cada programa solo debe manejar la información estrictamente necesaria para su correcto funcionamiento.



¡¡Actividad Propuesta – SPOILER!!

- “**main.h**” debe hacer los include de las bibliotecas estándar y de las bibliotecas creadas “**calculo.h**” y “**cadena.h**”
- “**main.c**” sólo debe hacer include a “**main.h**”.
- lo mismo con “**calculo.c**” que sólo debe hacer include de “**calculo.h**”.
- lo mismo con “**cadena.c**” que sólo debe hacer include de “**cadena.h**”.



Universidad Nacional
de La Matanza

DIIT



Departamento de Ingeniería e
Investigaciones Tecnológicas