

C Programming Language

Dr. Mostafa A. Elhosseini

Agenda

- ✧ C history
- ✧ Features of C
- ✧ ANSI standard C
- ✧ C and C++
- ✧ C program development
- ✧ Standard input / standard output
- ✧ IDE
 - How to Download "CodeBlocks":

C history

- ✧ C's predecessor was a language called B
- ✧ C was created by Dennis Ritchie at Bell Laboratories in early 1970s
 - Kernighan and Ritchie C - K&R C
 - there were no mechanism for checking parameters passed to functions
 - There were standard library but
 - The same routine would have different names
 - Sometimes the same routine worked in different ways
- ✧ C's intimate association with UNIX
- ✧ Increasing applications
 - Systems and application programming

Features of C

✂ Seems to be High-level language

- Able to access computer hardware and peripherals
- Assembly programmers considers C as High-level
- High-level programmers considers C as Low-level

✂ Fast code

- Often difficult to produce significant speed increase by rewriting in Assembly
- Racing car without seatbelt – built for ultimate speed, people are badly hurt if there is a crash

✂ Modular language

- Functions
- Standard library – toolkit of routines
 - ... Do not start from the scratch

✂ Portable

- It is easier to convert a C program to run on a different machine

ANSI-Standard C

- ✧ A **new version** of the language emerged from the American standards Institute
- ✧ This **version includes many features** that were not present in the Kernighan and Ritchie C
 - **Type Void**
 - **Function prototyping**

C and C++

- ✧ C++ is the **superset of C**
- ✧ C++ is an **Object-Oriented Programming OOP**
- ✧ C++ is most useful for **large program**
- ✧ **C** is suitable for programming **Microcontrollers**

C Program development environment

✂ C programs typically go through **six phases** to be executed

- Edit – Preprocess – compile – Link – Load – Execute

✂ **Edit / Create**

- Editor
 - Text editor
 - IDE
 - Codeblocks
 - Eclipse
 - Microsoft Visual Studio
- Save the file with extension **.c** - **Source file**

C Program development environment

✧ **Preprocessor program** executes automatically before the compiler's translation begin

- The Preprocessor obeys special commands called preprocessor directives
- The preprocessor is responsible for **removing comments** and **interpreting** special C language **preprocessors directives**
 - Begin with # symbol
- **# include**: **merge** the contents of named file “**Header file**” with the **source program** being compiled
- **# define**: it assigns a **symbolic name to a constant** – more **readable**
- **# ifdef**: **conditional compilation** and allow the programmer to **write** programs that adaptable to different operating environment

C Program development environment

- ✂ **Compiler** translates the C program into **machine-language code**
 - Machine language code (object code)
 - When a **compiler can not recognize** a statement –
 - **syntax error**
 - **Compile error**
 - The **compiler issues an error message** to help you locate and fix the incorrect statement

C Program development environment

- ✗ C programs typically contain references to functions defined elsewhere, such as standard libraries or private library
- ✗ The object code produced by the C compiler typically contain “holes” due to these missing parts
- ✗ A Linker links the object code with the code for the missing functions to produce an executable image
 - Standard library functions like printf and scanf are not part of the C programming language
 - For example, the compiler can not find a spelling error in printf or scanf
 - When the compiler compiles a printf statement, it provides space in the object program for a call to the library function

C Program development environment

- ✧ Before a program can be executed, the program must first be placed in memory – this is done by the **loader**

C Program development environment

- ✧ Finally, the computer, under the control of its CPU **executes** the program one instruction at a time

Standard Input – Standard Output

- ✧ Most C programs input and/or output data
- ✧ The Keyboard is the standard input stream
 - But `stdin` can be connected to another stream
- ✧ Data is often output to `stdout`
 - Normally is the computer screen
 - `stdout` can be connected to another stream

Note that...

- ✧ Learning C as is true with any language, is largely a matter of practice
- ✧ The more you look at C programs, the less complex they appear
- ✧ The exercises provide you with an opportunity to “go wrong”

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.



Code::Blocks

Go online for: www.codeblocks.org



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)

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.

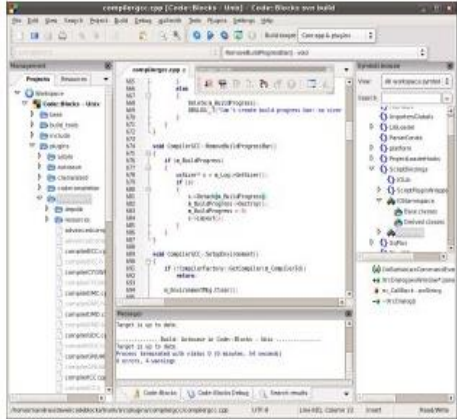
Finally, an IDE with all the features *you* need, having a consistent look, feel and operation across platforms.

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



Go online for: www.codeblocks.org

Downloads

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

- Download the binary release

This is the easy way for installing Code::Blocks. Download the setup file, run it on your computer and follow the instructions. Can't get any easier than that!

- **Download a nightly build:** There are also more recent so-called *nightly builds* available in the **repository** and **Jens' Fedora repository**. Other distributions usually follow provided by the maintainers. Please consider nightly builds to be *stable*, usually, unless stated otherwise.

Go online for: www.codeblocks.org

Please select a setup package depending on your platform:

- Windows XP / Vista / 7 / 8.x / 10
- Linux 32-bit
- Linux 64-bit
- Mac OS X

NOTE: For older OS'es use 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 Debian and Fedora users) in **Jens' Debian repository** and **Je** note that we consider nightly builds to be *stable*, usually.

NOTE: We have a **Changelog** for **16.01**, that gives you an overview over the enhancements and fixes we have put in the new release.



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

File	Date	Download from
codeblocks-16.01-setup.exe	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01-setup-nonadmin.exe	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01-nosetup.zip	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01mingw-setup.exe	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01mingw-nosetup.zip	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01mingw_fortran-setup.exe	28 Jan 2016	Sourceforge.net or FossHub

Go online for: www.codeblocks.org

Please select a setup package depending on your platform:

- Windows XP / Vista / 7 / 8.x / 10
- Linux 32-bit
- Linux 64-bit
- Mac OS X

NOTE: For older OS'es use 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 Debian and Fedora users) in [Jens' Debian repository](#) and [Jens' Fedora repository](#). Please note that we consider nightly builds to be *stable*, usually.

NOTE: We have a [Changelog for 16.01](#), that gives you an overview over the enhancements and fixes we have put in the new release.



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

File	Date	Download from
codeblocks-16.01-setup.exe	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01-setup-nonadmin.exe	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01-nosetup.zip	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01mingw-setup.exe	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01mingw-nosetup.zip	28 Jan 2016	Sourceforge.net or FossHub
codeblocks-16.01mingw_fortran-setup.exe	28 Jan 2016	Sourceforge.net or FossHub

Installation of "CodeBlocks":



Choose Components

Choose which features of CodeBlocks you want to install.

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

Select the type of install: Full: All plugins, all tools, just everything

Or, select the optional components you wish to install:

- ☒ **Default install**
- ☒ Contrib Plugins
- ☒ C::B Share Config

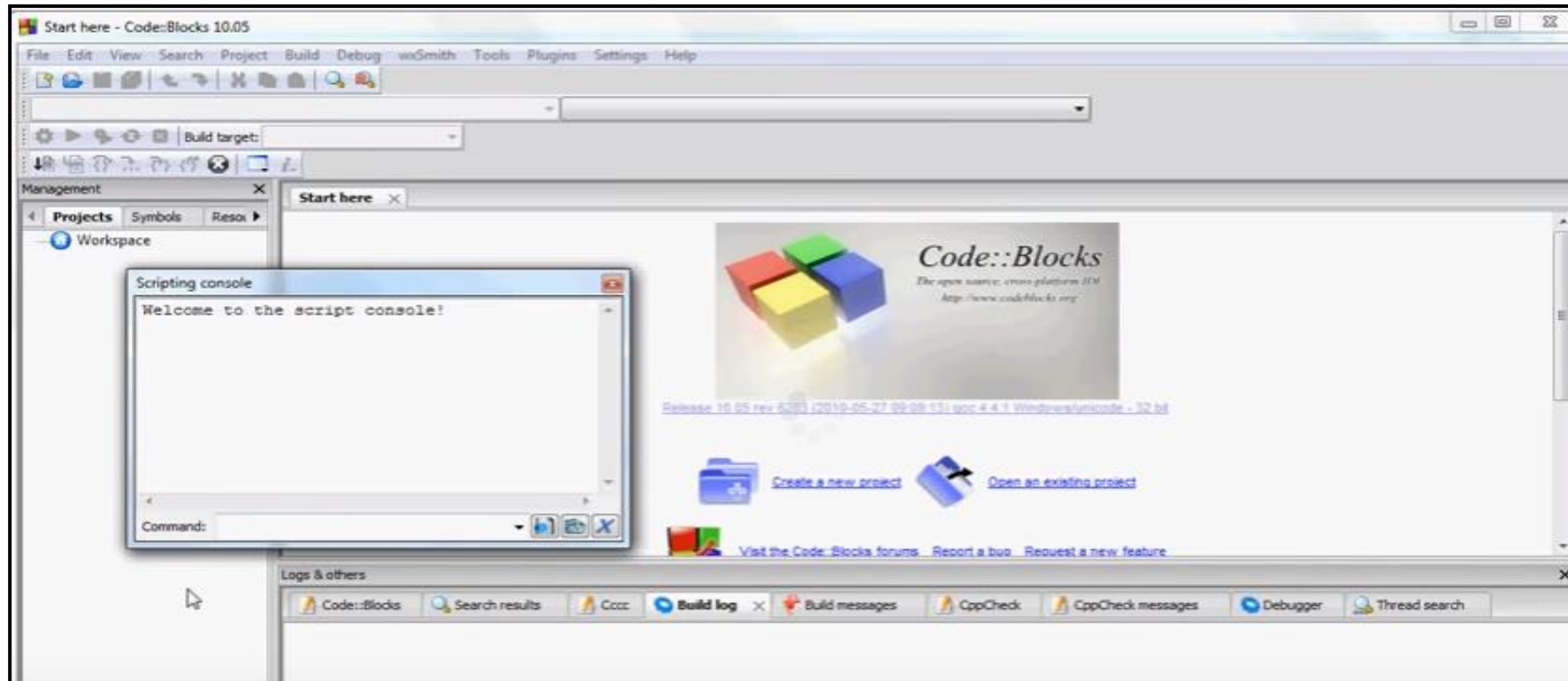
Space required: 71.4MB

Description
Position your mouse over a component to see its description.

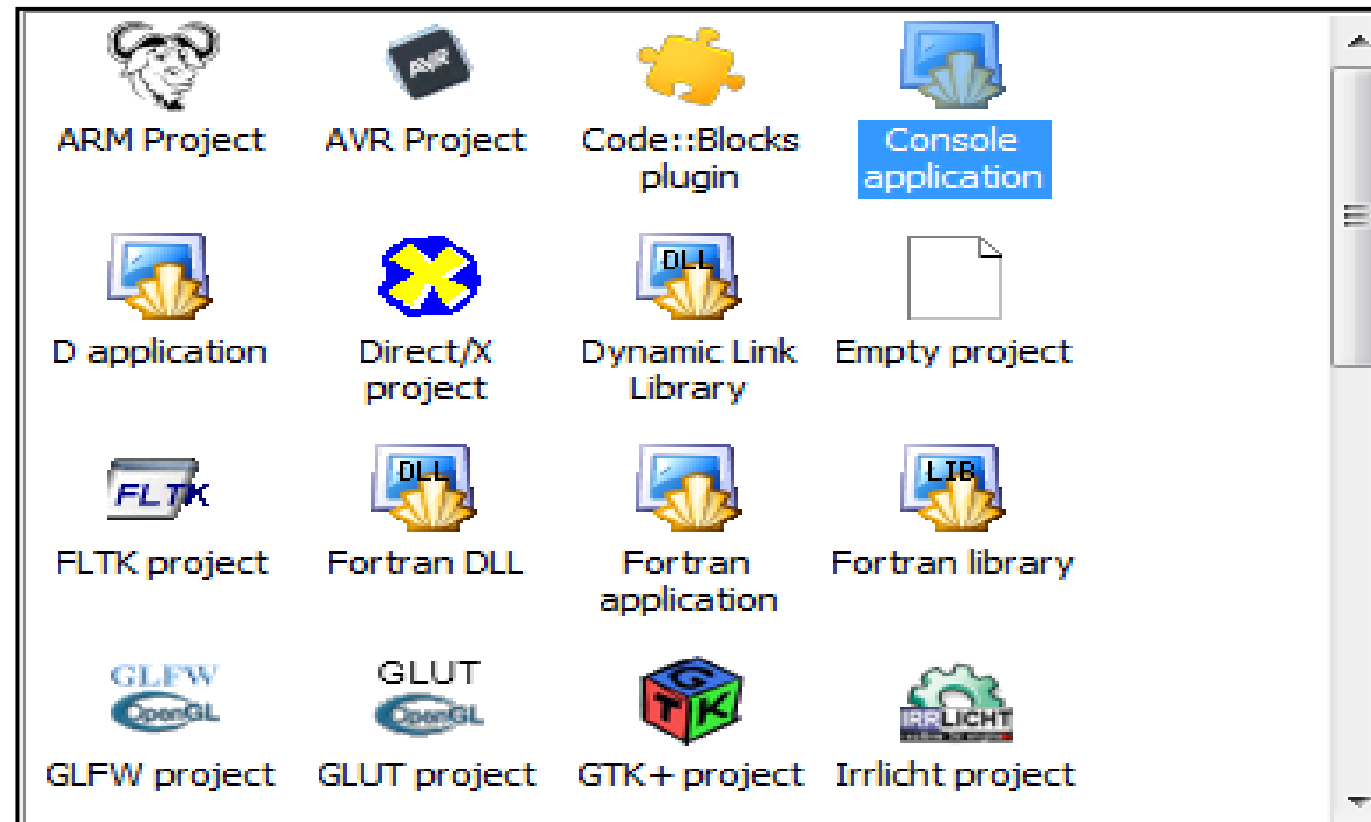
Nullsoft Install System v2.46

< Back Next > Cancel

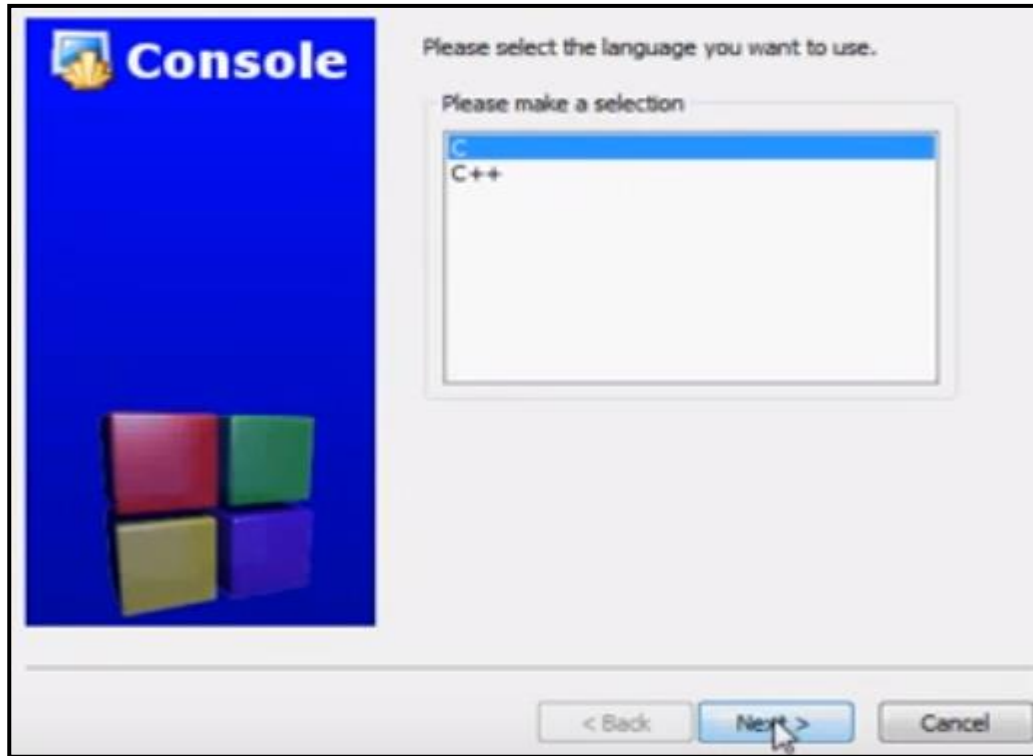
Installation of "CodeBlocks":



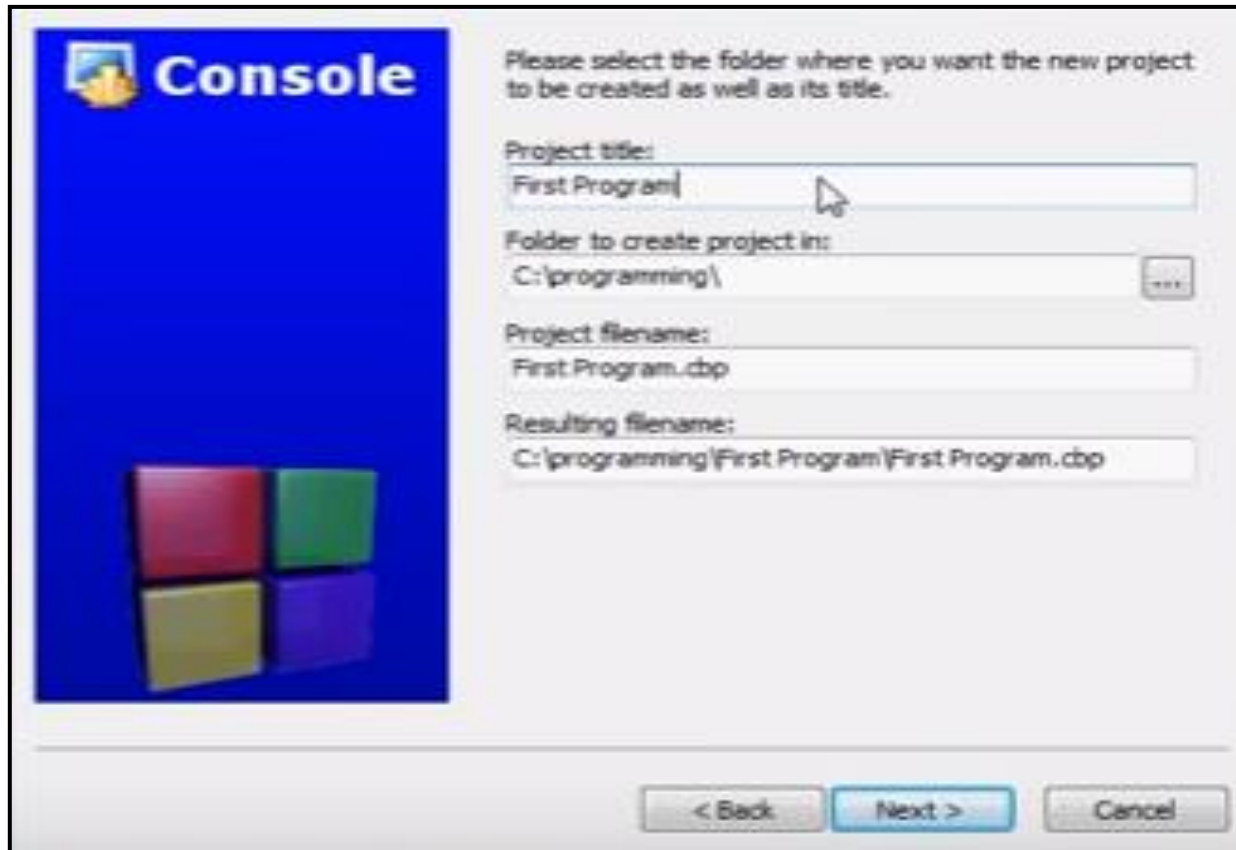
Installation of "CodeBlocks":



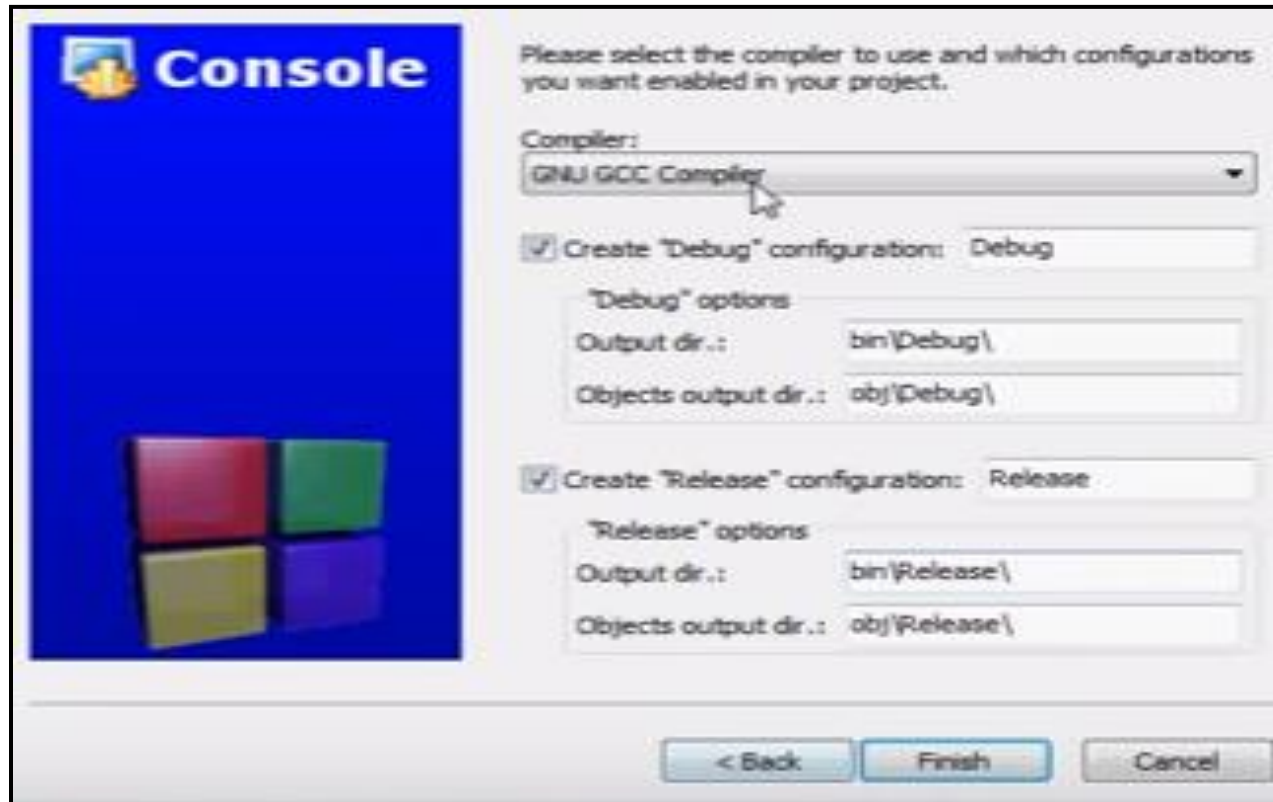
Installation of "CodeBlocks":



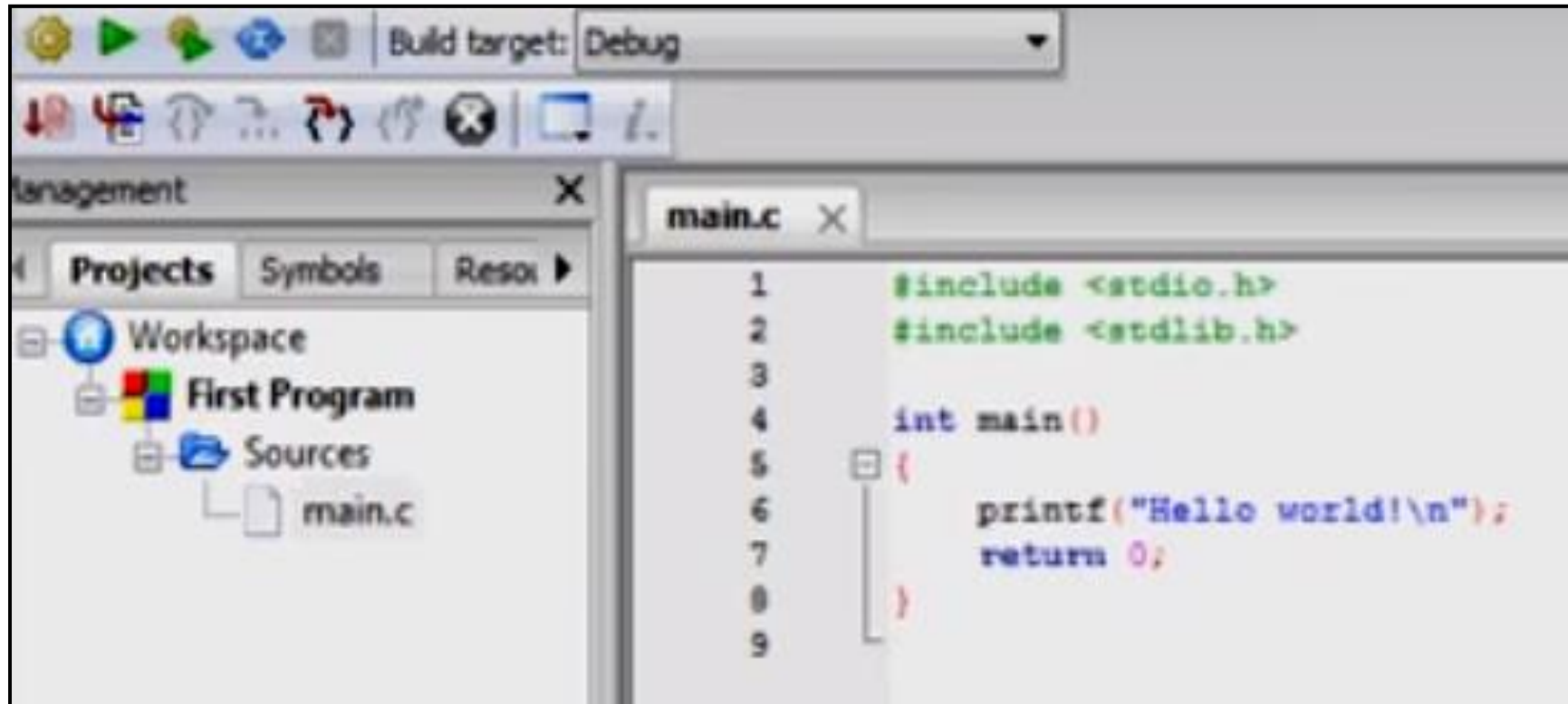
Installation of "CodeBlocks":



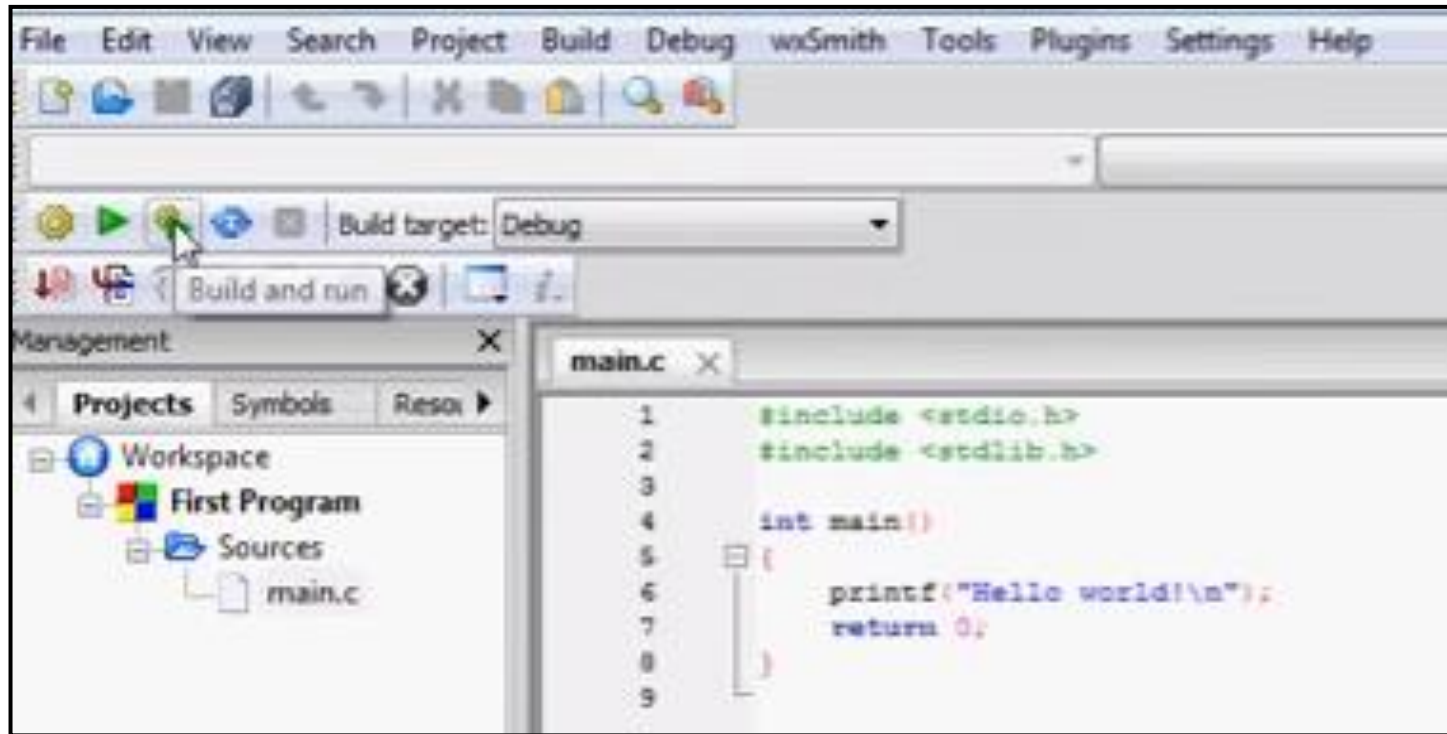
Installation of "CodeBlocks":



Hello World



Build and Run



Build and Run

