

Introduzione al linguaggio C

Prof. Paola Barra

Laboratorio di Programmazione 1.

Aula 1 : lunedì 11:00-13:00

Aula 1 : mercoledì 14:00-16:00

Ricevimento: il lunedì pomeriggio nel mio ufficio (stanza 411)

O a richiesta da remoto su teams.

paola.barra@uniparthenope.it



Titolo unità didattica: Introduzione al linguaggio C

[03]

Titolo modulo : Linguaggio di programmazione C

[02-C]

Aspetti generali del linguaggio C ed 'esecuzione' di un programma C

Argomenti trattati:

- ✓ introduzione storica al C
- ✓ compilatore e precompilatore
- ✓ ambienti di sviluppo
- ✓ ambiente CodeBlocks
- ✓ esecuzione di un semplice programma in C

Prerequisiti richiesti: AP-03-01-T

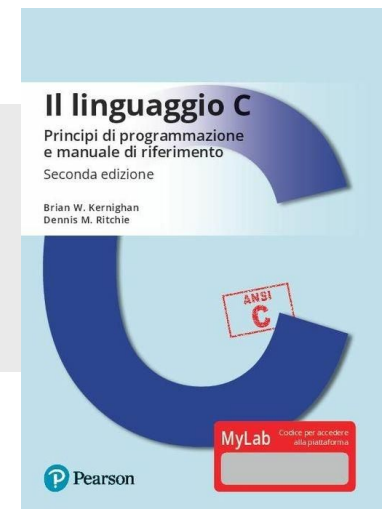
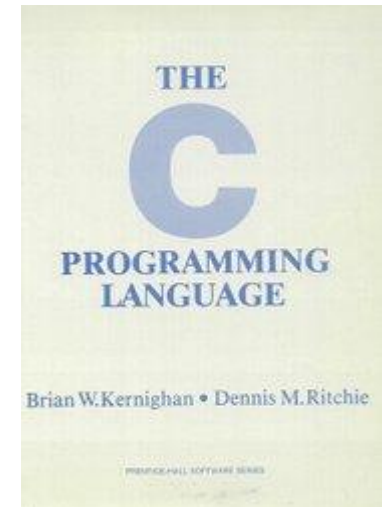
il **linguaggio C** è un **linguaggio di programmazione** standardizzato (**ANSI/ISO C**, standard riconosciuto a livello internazionale)



ideatori:

Ken Thompson, Dennis Ritchie, 1970

- ✓ sviluppo di software applicativo
- ✓ sviluppo di software di sistema
(Unix, Linux, Matlab sono scritti in C)



il **linguaggio C** è un **linguaggio di programmazione** standardizzato (**ANSI/ISO C**, standard riconosciuto a livello internazionale)



Dennis Ritchie

Dennis Ritchie

Nascita: 9 settembre 1941

Morte: 8 ottobre 2011

Contributi:

- **Linguaggio C:** Ritchie ha sviluppato il linguaggio di programmazione C presso Bell Labs negli anni '70. Questo linguaggio è diventato uno dei più influenti e utilizzati nella storia della programmazione, ed è alla base di molti sistemi operativi e applicazioni software.
- **Sistema Operativo Unix:** Insieme a Ken Thompson, Ritchie ha contribuito allo sviluppo di Unix, un sistema operativo innovativo che ha influenzato molti altri sistemi operativi moderni, inclusi Linux e macOS.

il **linguaggio C** è un **linguaggio di programmazione** standardizzato (**ANSI/ISO C**, standard riconosciuto a livello internazionale)

Ken Thompson

Nascita: 4 febbraio 1943

Contributi:

- **Sistema Operativo Unix:** Thompson è il co-creatore di Unix. Ha iniziato a svilupparlo nel 1969, inizialmente come un progetto personale. La sua visione per un sistema operativo semplice e portabile ha avuto un impatto duraturo sul design dei sistemi operativi.
- **Linguaggio B:** Prima di C, Thompson ha sviluppato il linguaggio di programmazione B, che ha influenzato la creazione di C stesso.



Ken Thompson

il **linguaggio C** è un **linguaggio di programmazione** standardizzato (**ANSI/ISO C**, standard riconosciuto a livello internazionale)



Dennis Ritchie e Ken Thompson sono due figure fondamentali nella storia dell'informatica, noti principalmente per il loro lavoro sulla creazione del sistema operativo Unix e del linguaggio di programmazione C.

Collaborazione

Ritchie e Thompson hanno lavorato insieme a Bell Labs e hanno avuto un ruolo cruciale nella formazione del panorama dell'informatica moderna. Il loro lavoro ha non solo portato alla creazione di strumenti e sistemi operativi utilizzati ancora oggi, ma ha anche influenzato profondamente la comunità dei programmatori e lo sviluppo di software.

Riconoscimenti

Nel 1983, Ritchie e Thompson sono stati insigniti del premio Turing, un riconoscimento prestigioso nel campo dell'informatica, per il loro contributo fondamentale al settore.

Questi due pionieri hanno lasciato un'eredità duratura, e le loro invenzioni continuano a influenzare il modo in cui programiamo e utilizziamo i computer oggi.



C è un linguaggio di programmazione **general-purpose, strutturato a blocchi, procedurale, imperativo** sviluppato nel 1972 presso i Bell Telephone Laboratories per esser usato con il sistema operativo Unix

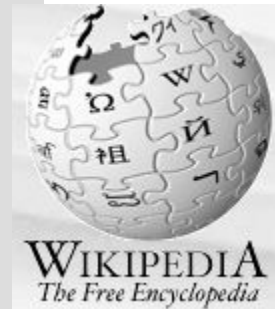
sebbene nato per scrivere software di sistema, C è anche ampiamente utilizzato per il software applicativo

C ha anche influenzato molti altri linguaggi di programmazione, specialmente C++, che è stato progettato per essere una versione avanzata di C

C ha influenzato lo sviluppo di altri linguaggi di programmazione:

JAVA, C#, PERL, PHP, JAVASCRIPT

<http://www.ansi.org>



corso di C su **Wikiversity**

<http://en.wikiversity.org/wiki/Topic:C>



the Wikiversity content-development project for the **C** computer language, part of the Division of Computer Programming

libro di C su **Wikibooks**

http://en.wikibooks.org/wiki/C_Programming

il più semplice dei programmi C

```
#include <stdio.h>
void main()
{
printf("Hello world!\n");
}
```

Per i primi esperimenti provate su

<https://www.mycompiler.io/it/new/c>

```
#include <stdio.h>
int main() {
    printf("Hello world!\n");
    return 0;
}
```

l'esecuzione del programma visualizza
sullo schermo la frase

Hello world!

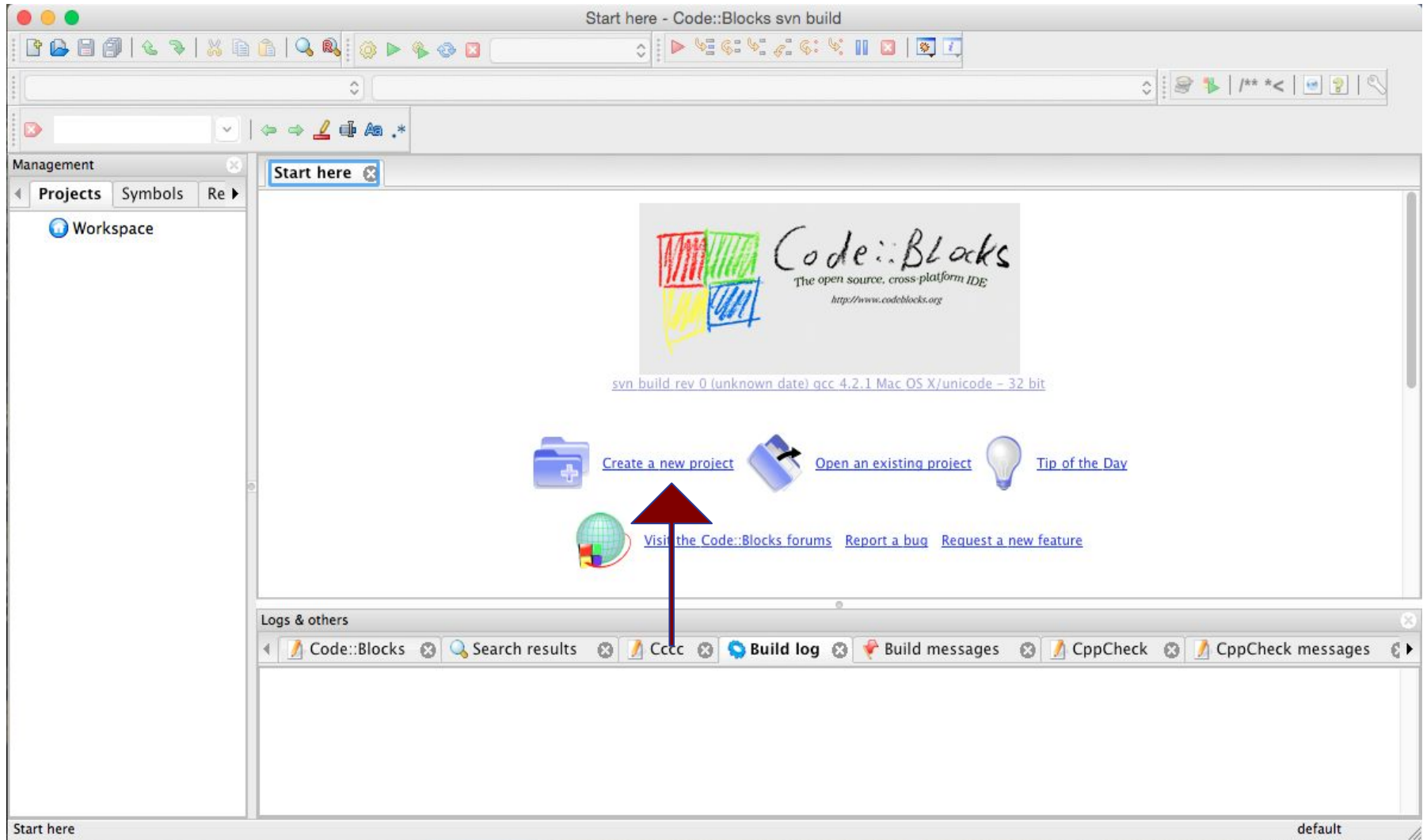
Press any key to continue_

```
#include <stdio.h>
int main() {
    printf("questo e' il mio primo programma C\n");
    return 0;
}
```

l'esecuzione del programma visualizza
sullo schermo la frase

```
questo e' il mio primo programma C
Press any key to continue_
```

ambiente di sviluppo Code::Blocks















New from template

Projects Build targets Files Custom User templates

Go

Cancel

Category: <All categories>

 ARM Project	 Console application	 FLTK project	 Fortran library	GTK
 AVR Project	 D application	 Fortran DLL	 GLFW	Irrli
 Code::Blocks plugin	 Empty project	 Fortran application	 GLUT	Ligh

View as
 Large icons
 List

TIP: Try right-clicking an item

1. Select a wizard type first on the left
2. Select a specific wizard from the main window (filter by categories if needed)
3. Press Go



Console

Please select the language you want to use.

Please make a selection



< Back

Next >

Cancel



Console

Please select the folder where you want the new project to be created as well as its title.

Project title:

Folder to create project in:

Project filename:

Resulting filename:





Please select the compiler to use and which configurations you want enabled in your project.

Compiler:

GNU GCC Compiler

Create "Debug" configuration: Debug

"Debug" options

Output dir.: bin/Debug/

Objects output dir.: obj/Debug/

Create "Release" configuration: Release

"Release" options

Output dir.: bin/Release/

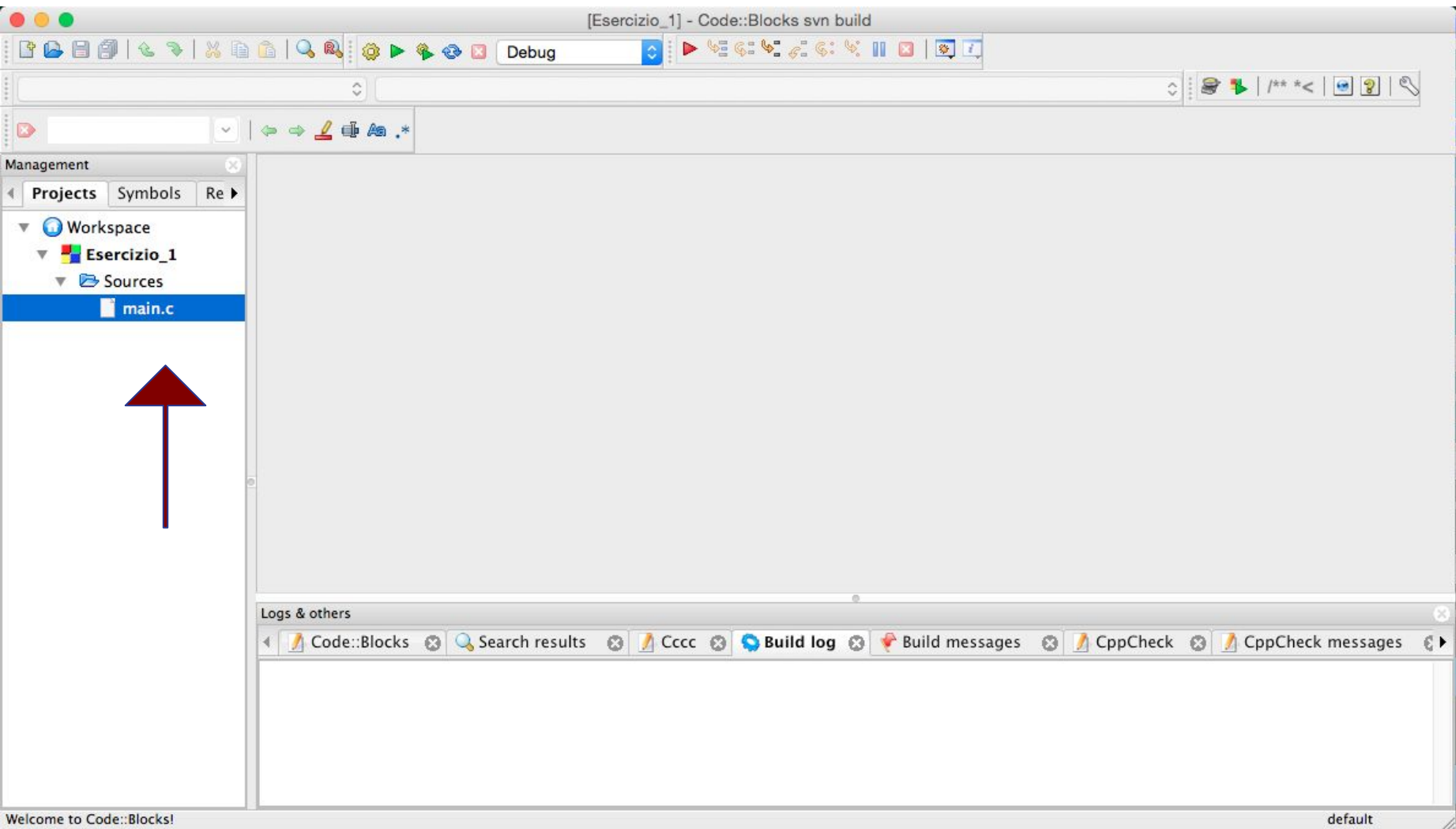
Objects output dir.: obj/Release/



< Back

Finish

Cancel



In Code::Blocks

```
#include <stdio.h>
int main()
{
printf("Hello world!\n");

return 0;
}
```

In Code::Blocks sostituire `void` con `int` e aggiungere `return 0;`

main.c [prova_1] - Code::Blocks svn build

Debug

<global>

Management

- Projects
- Symbols
- Re ▶

Workspace

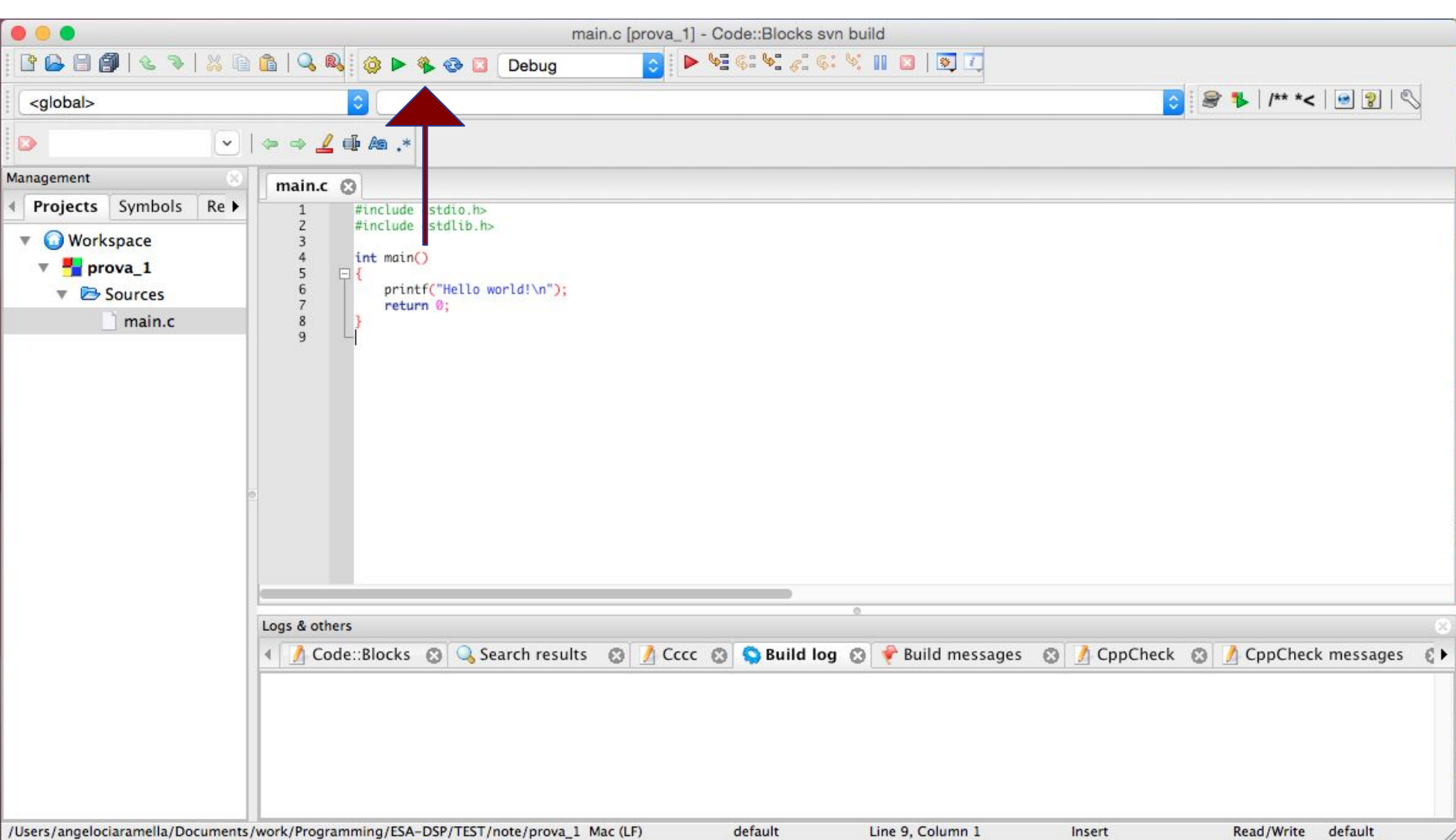
- prova_1
 - Sources
 - main.c

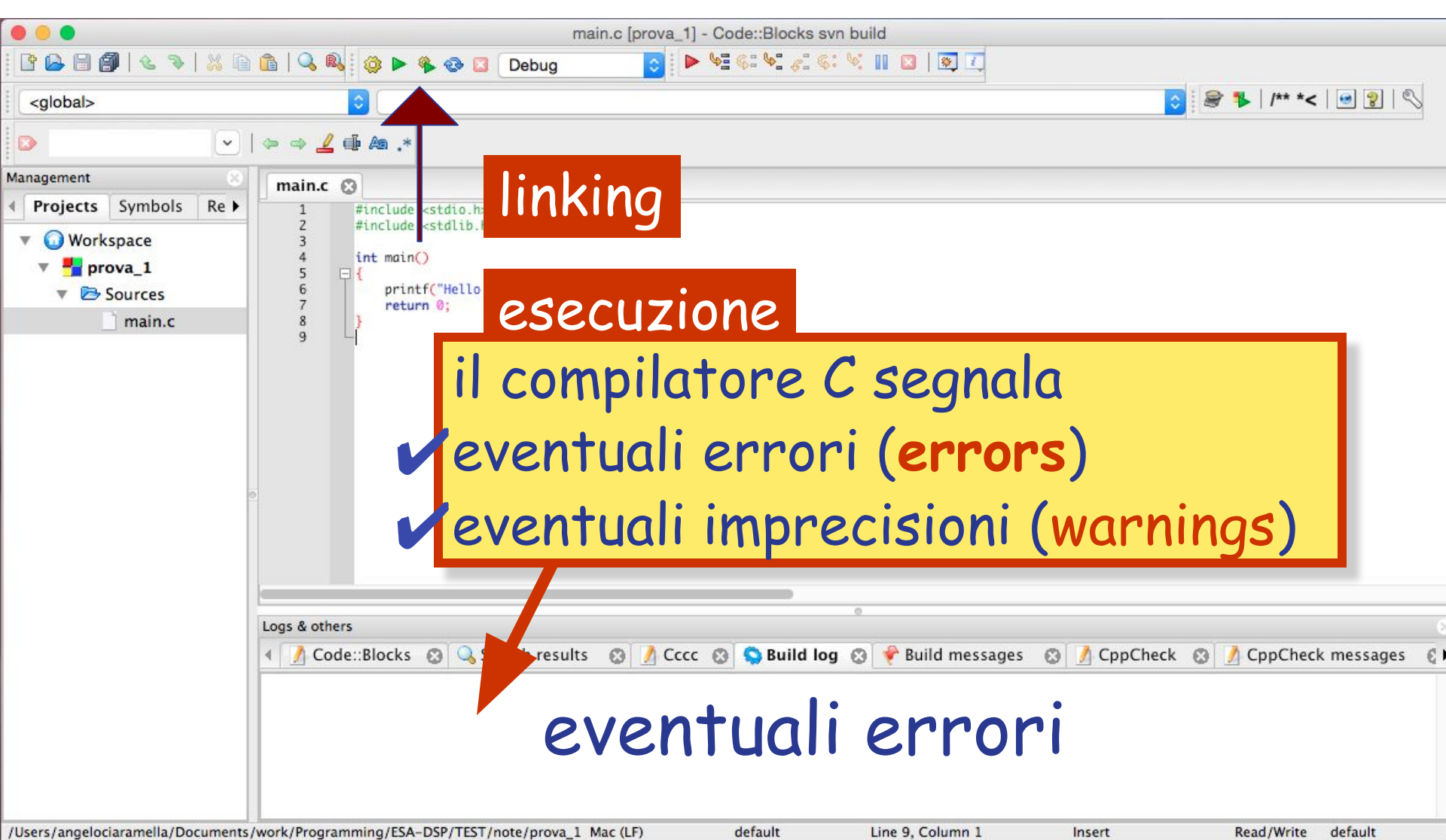
```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     printf("Hello world!\n");
7     return 0;
8 }
9
```

Logs & others

- Code::Blocks
- Search results
- Cccc
- Build log
- Build messages
- CppCheck
- CppCheck messages

/Users/angelociaramella/Documents/work/Programming/ESA-DSP/TEST/note/prova_1 Mac (LF) default Line 9, Column 1 Insert Read/Write default





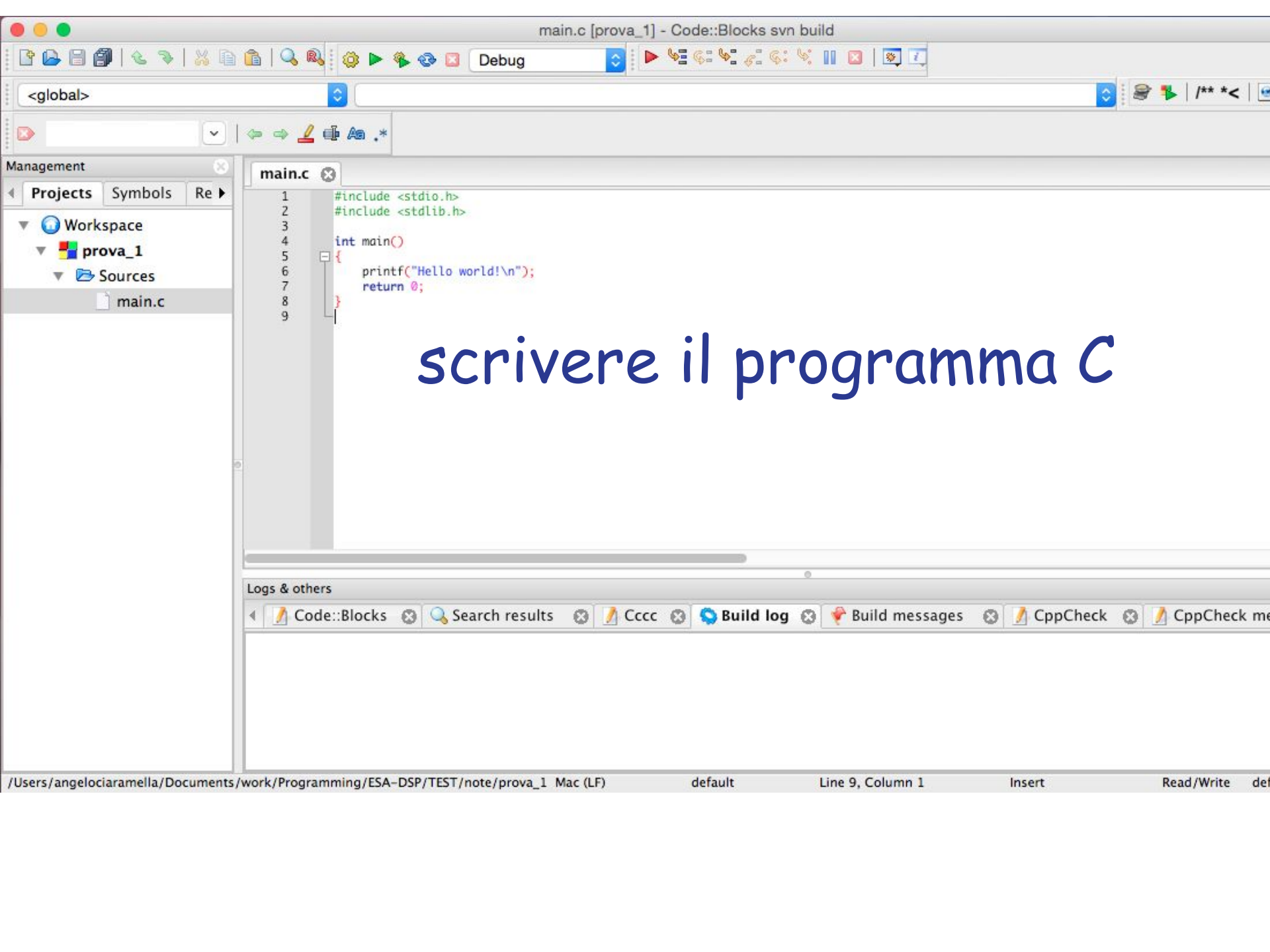
linking

esecuzione

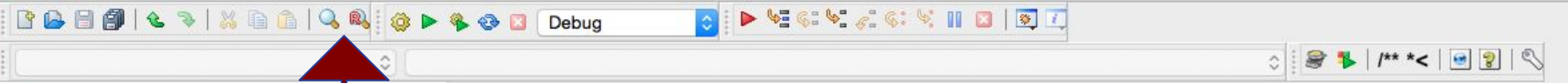
il compilatore C segnala

- ✓ eventuali errori (**errors**)
- ✓ eventuali imprecisioni (**warnings**)

eventuali errori



scrivere il programma C



Management

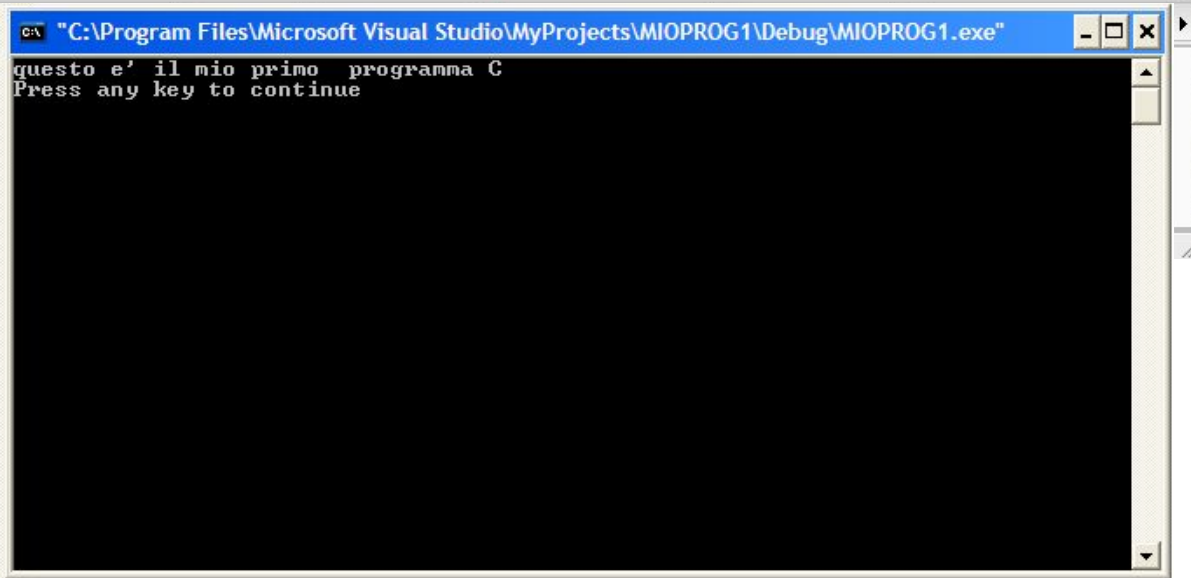
Projects Symbols Re ▶

- Workspace
 - prova_1
 - Sources
 - main.c

```
1 #include <stdio.h>  
2 int main()  
{  
printf("questo e' il mio primo programma C\n");  
  
return 0;  
}
```

Logs & others

Code::Blocks



```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
printf("questo e' il mio primo programma C\n");
```

```
}
```

struttura di un compilatore per ANSI/ISO C

PRECOMPILATORE

COMPILATORE

richiesta al (pre)compilatore di richiamare le istruzioni di I/O

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
printf("questo e' il mio primo programma C\n");
```

```
return 0;
```

```
}
```

è l'incipit di ogni programma in C

istruzione di visualizzazione della frase (stringa)
questo e' il mio primo programma C

\n provoca l'andata a capo (stampa del carattere di fine riga)

```
#include <stdio.h>
void main()
{
printf("questo e' il mio primo programma C\n");
}
```

una coppia di apici doppi racchiude una **stringa di caratteri**

; chiude le **istruzioni**

le parentesi { e } racchiudono il **corpo del programma**

il più semplice dei programmi C

```
1  #include <stdio.h>
2
3  int main()
4  {
5      printf("Hello World!");
6      return 0;
7  }
```

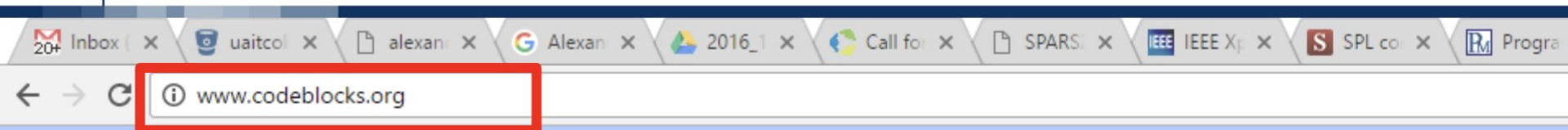
Installazione Codeblocks

Installazione Code::Blocks (Win, Mac OS e Linux)





Scaricare Code::Blocks per Windows



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
- Ticket

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

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 plugins. Some functionality is already provided by plugins!

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

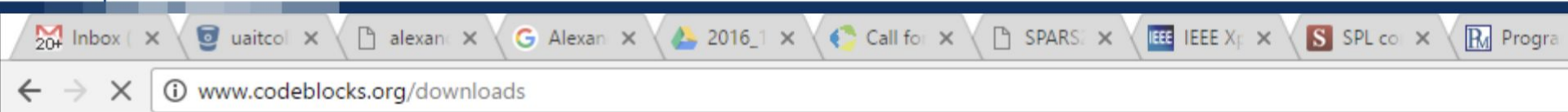
We hope you enjoy using Code::Blocks!

The Code::Blocks Team

Code::Blocks 16.01 is here!



Download the Binary Release



Code::Blocks

Code::Blocks - The IDE with all the features you need, having

Home

Features

Downloads

Forums

Wiki

Main

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

Quick links

- FAQ
- Wiki
- Forums
- Forums

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 Code::Blocks will be installed.

- **Download a nightly build:** There are also more recent so-called *nightly builds* available in the forums or on SourceForge. They usually follow provided by the community (Big "Thank you" for that!) Please note that we consider nightly builds as unstable.

- **Download the source code**

If you feel comfortable building applications from source, then this is the recommend way to download Code::Blocks. It gives you the ability to update to newer versions or, even better, create patches for bugs you may find and contributing them back to the project.

- **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 to customize the IDE to your needs. You can also retrieve a stable release to benefit from bug-fixes!

Besides Code::Blocks itself, you can compile extra plugins from contributors to extend its functionality.



Scegliete il Vostro OS



Code::Blocks

Code::Blocks - The IDE with all the features you need, hav

Home

Features

Downloads

Forums

Wiki

Main

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

Quick links

- FAQ
- Wiki
- Forums
- Forums

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](#)

NOTE: There are also more recent *nightly builds* available in the [forums](#) or (for Debian and Fedora users) in [Jens' Del](#)

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



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

File	
codeblocks-16.01-setup.exe	28 Ja



Download per Windows

Sono circa 100MB e contiene sia l'IDE (Integrated Development Environment) che il compilatore

Cliccare sul repository da cui scaricare (e.g. Sourceforge.net)



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

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

NOTE: The codeblocks-16.01-setup.exe file includes Code::Blocks with all plugins. The codeblocks-16.01-setup-nonadmin.exe file is provided for convenience to users

NOTE: The codeblocks-16.01mingw-setup.exe file includes *additionally* the GCC/C++ compiler and GDB debugger from TDM-GCC (version 4.9.2, 32 bit, SJLJ). The codeblocks-16.01mingw_fortran-setup.exe file includes the GFortran compiler (TDM-GCC).

NOTE: The codeblocks-16.01(mingw)-nosetup.zip files are provided for convenience to users that are allergic against installers. However, it will not allow to select plugins and shortcuts. For the "installation" you are on your own.

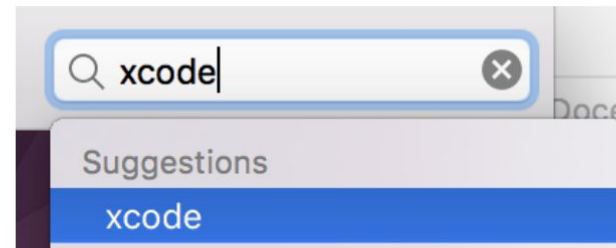
If unsure, please use codeblocks-16.01mingw-setup.exe!



Prerequisiti MacOS: Installare Xcode

Prima di installare Code::Blocks, è necessario installare un prerequisito: Xcode

Andare sull'AppStore e scaricare Xcode



Search Results for "xcode"



Xcode

Developer Tools

★★★★☆ 23 Ratings

Essentials

INSTALL ▾



Prerequisiti: Installare Xcode (cnt)

Se non si vuole scaricare Xcode, si può in alternativa installare la Command Line Tools for Xcode:

1. Aprire il Terminale



2. Scrivere sul Terminale:

```
xcode-select --install
```

3. Premere Invio e seguire le istruzioni di installazione

Download per MacOS (10.15 Catalina)

- Per MacOS 10.15 occorre Code::Blocks 64 bit
- Una versione stabile è disponibile al seguente link:
<https://sourceforge.net/projects/codeblocks/files/Binaries/17.12/Mac/>

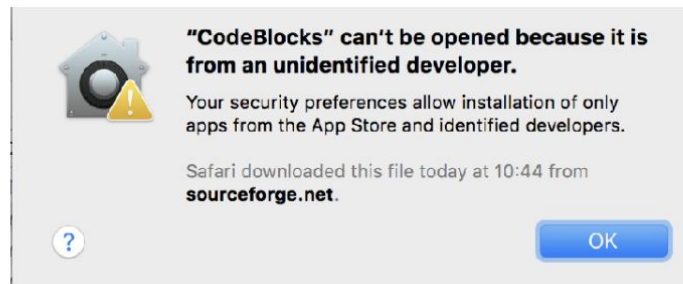
The screenshot shows the SourceForge website for Code::Blocks. The page title is "Code::Blocks" and it is described as "A free C, C++ and Fortran IDE". The page is for the "Binaries" section, specifically for "17.12" on "Mac". A green button labeled "Download Latest Version" is visible, with the file name "codeblocks-17.12_OSX64.dmg (18.4 MB)" and a "Get Updates" button. Below this, a table lists the files available for download. The file "codeblocks-17.12_OSX64.dmg" is highlighted with a red box. The table has columns for Name, Modified, Size, and Downloads / Week.

Name	Modified	Size	Downloads / Week
Parent folder			
codeblocks-17.12_OSX64.dmg	2018-04-04	18.4 MB	1,092

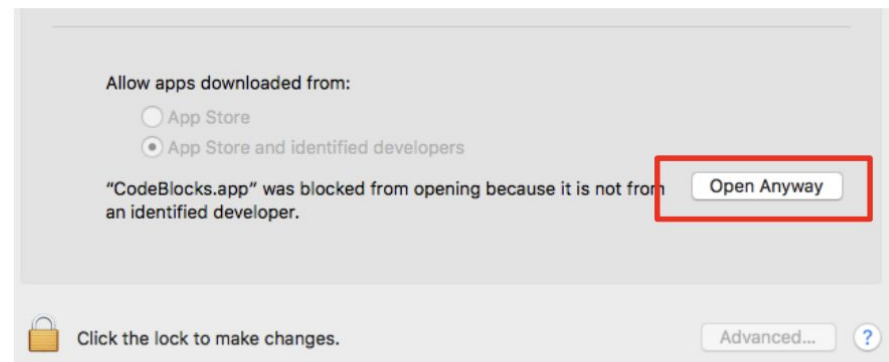
- Scaricare il file segnalato e proseguire nell'installazione

Installazione per MacOS

- Scaricare lo zip e scompattarelo con doppio click
- Poiché l'eseguibile non è firmato digitalmente, apparirà:

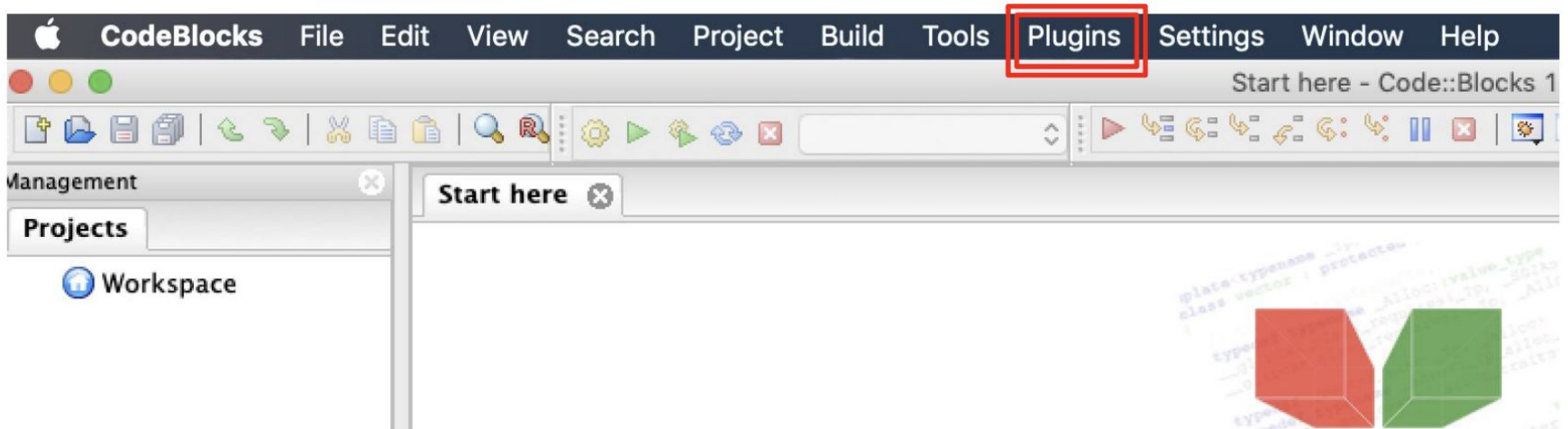


- Andare nelle preferences di MacOS sezione pannello Security e abilitarlo con «Open Anyway»



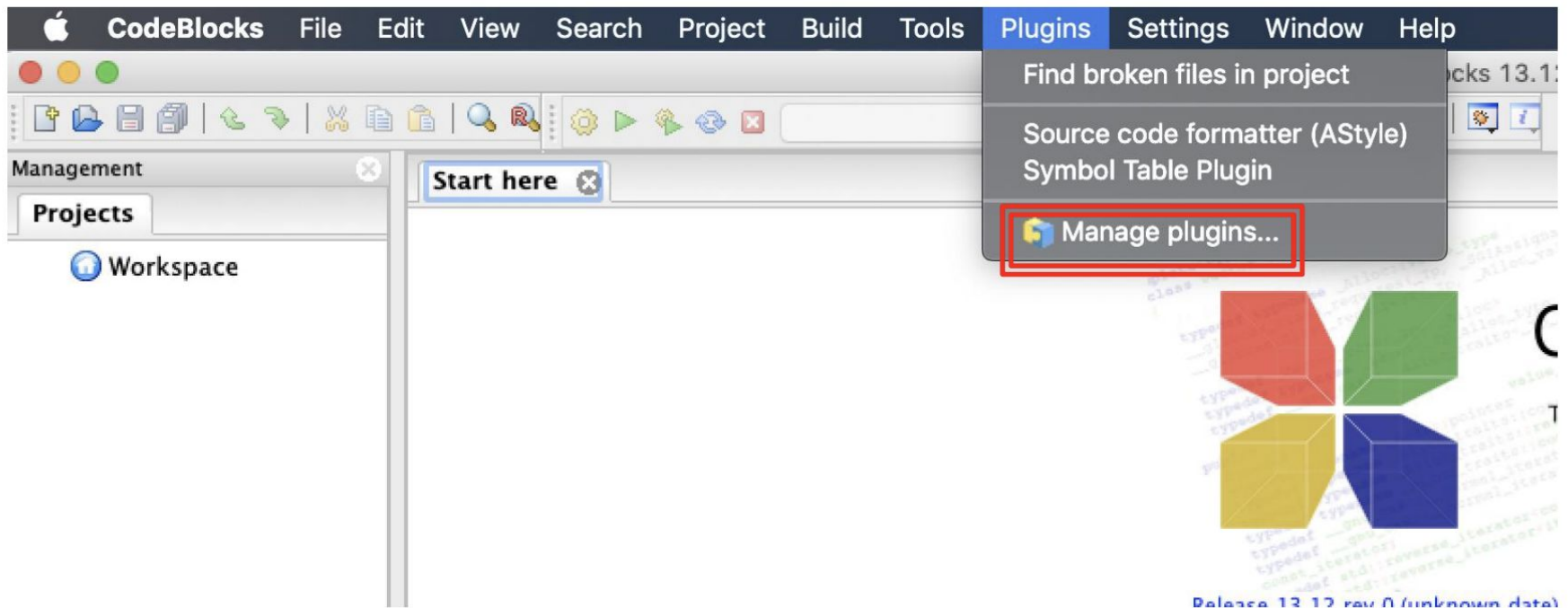
Installazione per MacOS: disabilitare plugins

- La versione completa di CodeBlocks per MacOS potrebbe risultare instabile e incorrere in fastidiosi crash durante l'esecuzione
- Per risolvere questo problema è sufficiente disabilitare o disinstallare una serie di componenti aggiuntivi (plugins) non necessari per i fini del corso



Installazione per MacOS: disabilitare plugins

- Aprire CodeBlocks ed accedere alla finestra di gestione dei plugins





Installazione per MacOS: disabilitare plugins

- Quindi disinstallare (Uninstall) o disabilitare (Disable) tutti i plugin **non presenti** in questa lista:

Manage plugins

Installed plugins:

Title	Version	Enabled	Filename
Symbol Table Plugin	1.0	Yes	libSymTab.dylib
Source code formatter (AStyle)	1.2	Yes	libastyle.dylib
SmartIndentCpp	0.1	Yes	libSmartIndent
Open files list	1.0	Yes	libopenfileslist.
Files extension handler	1.0	Yes	libdefaultmime
Compiler	0.99	Yes	libcompiler.dyli
BrowseTracker	1.2.99 2012/11/18	Yes	libBrowseTrack
Autosave	0.1	Yes	libautosave.dyl

Buttons: Enable, **Disable**, Install new, **Uninstall**, Export



Installazione per Linux (Ubuntu, Debian e derivate)

- Aprire un terminale (ctrl+alt+t)
- Installare il compilatore scrivendo (verrà richiesta la password):

```
sudo apt-get install build-essential
```

(potrebbe essere già installato nella vostra distribuzione)

- Installare Code::Blocks scrivendo:

```
sudo apt-get install codeblocks
```

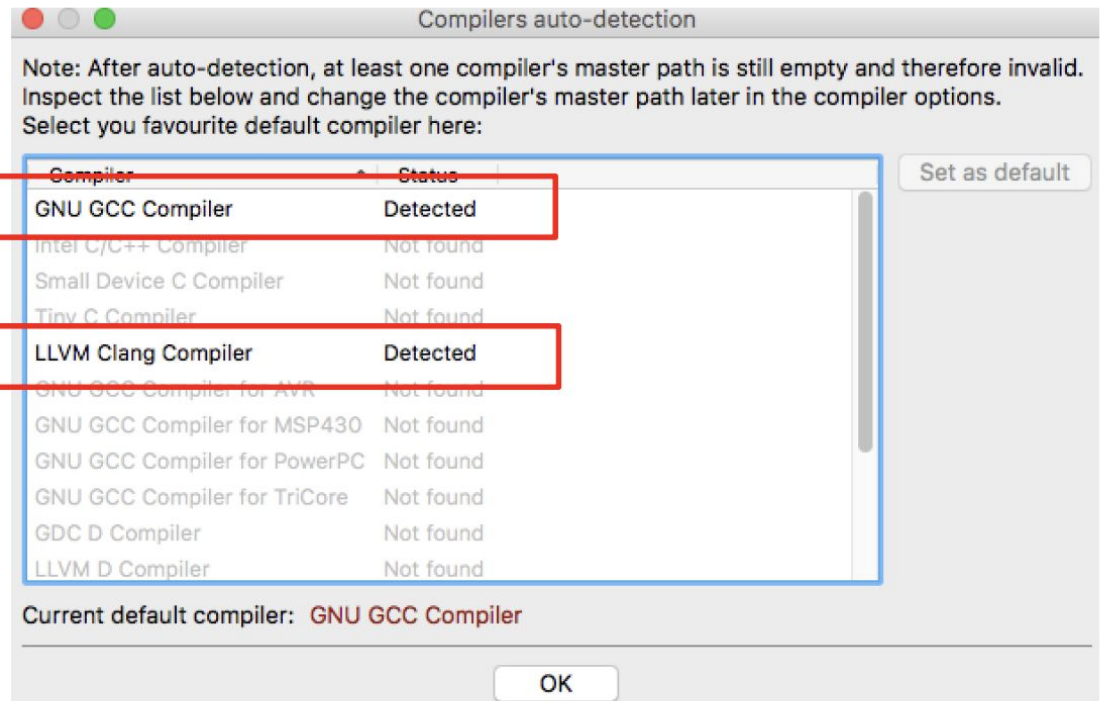


Impostazione del Compilatore

- All'apertura di Code::Block, dovrebbe apparire

Per Windows e
Linux

Per MacOS

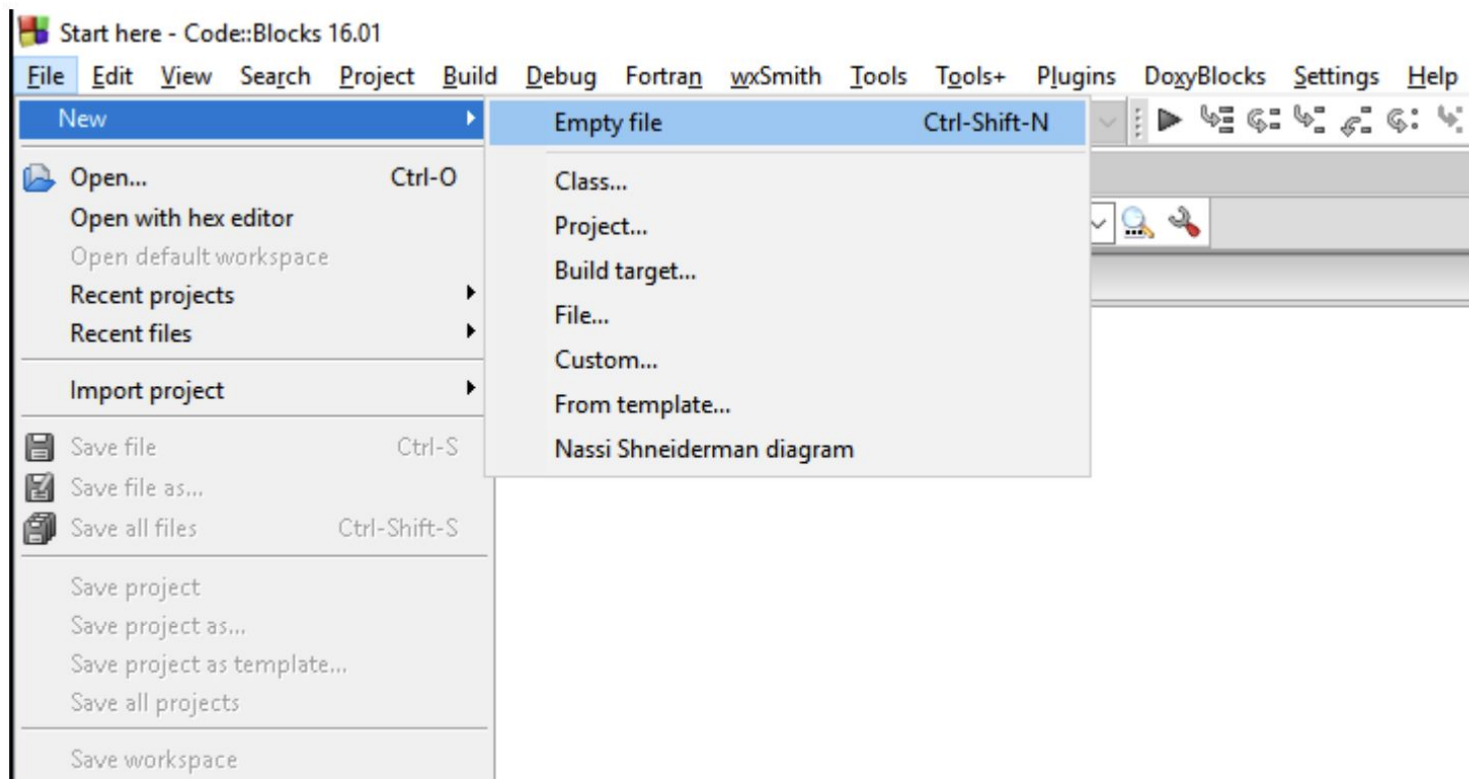


- Se non appare, andare nel menu: Settings->Compiler ed impostarlo manualmente



Per Iniziare

Selezionare «empty file» dal menu File->New





Salvare il file con estensione .c e tipo C/C++ files

Save file

< > > This PC > Desktop > lez2Live

Organize ▾ New folder

Name	Date modified	Type	Size
far2cel.c	05/10/2016 11:15	C source file	
hello.c	05/10/2016 10:41	C source file	

File name: helloWorld.c

Save as type: C/C++ files

Hide Folders

Save

Alternative a Code::Blocks

- Xcode
- Visual Studio Code
- Dev C++
- Replit
<https://replit.com/>
- Coding C (per google Play)
- ...

E se non ho un portatile...

Replit può essere usato da tablet o cellulare.

Il consiglio è sempre quello di fare gli esercizi in classe, anche carta e penna, e poi provare a compilarli a casa.

MA...

Sono comunque previste esercitazioni in laboratorio prima delle prove intercorso per chi non ha potuto esercitarsi in classe con gli altri.

Segnalatemi questa esigenza e risolviamo insieme.
