

# Code::Blocks, installazione ed uso

ALESSIO MAURO FRANCHI \*<sup>1</sup>

<sup>1</sup>*Dipartimento DEIB, Politecnico di Milano, Milano*

9 ottobre 2014

## Indice

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduzione</b>                       | <b>1</b>  |
| <b>2</b> | <b>Installare Code::Blocks: Windows</b>   | <b>3</b>  |
| <b>3</b> | <b>Installare Code::Blocks: Linux</b>     | <b>4</b>  |
| <b>4</b> | <b>Installare Code::Blocks: OsX</b>       | <b>4</b>  |
| <b>5</b> | <b>Cominciare a programmare</b>           | <b>8</b>  |
| <b>6</b> | <b>Compilare ed eseguire il programma</b> | <b>13</b> |
| <b>7</b> | <b>Comandi rapidi utili</b>               | <b>14</b> |

## 1 Introduzione

Il linguaggio di programmazione C fa parte della famiglia dei linguaggi "compilati"; a questa famiglia appartengo quei linguaggi che necessitano di un compilatore per poter essere eseguiti.

In poche parole, tralasciando ogni dettaglio tecnico extra, il compilatore è un programma in grado di tradurre il vostro codice sorgente in una sequenza di istruzioni leggibili ed eseguibili dal computer. Per programmare è quindi necessario avere installato nel proprio computer un editor di testo qualsiasi

---

\*alessiomauro.franchi@polimi.it



## 2 Installare Code::Blocks: Windows

Questa sezione vi guiderà nell'installazione di Code::Blocks in ambiente Windows; la seguente guida è valida per Windows 2000/XP/7/8/8.1.

Scaricate dal sito <http://www.codeblocks.org/> il file di setup; lo trovate seguendo il percorso:

”Downloads” → ”Download the binary release” → ”Windows 2000/XP/-Vista/7//8”

Sono elencate tre versioni; la prima non include il compilatore, la seconda e la terza sì; queste due ultime installeranno infatti nel vostro sistema anche “MinGW”<sup>4</sup>, un piccolo software che comprende un compilatore e tutti gli strumenti richiesti per la compilazione.

La differenza tra questi due ultimi file è esclusivamente la versione di MinGW; se non siete sicuri, scegliete il secondo file dalla lista (quello senza ”TDM-GCC-XXX”).

Come per ogni applicazione eseguite il file appena scaricato, accettate la licenza e seguite il percorso guidato; ad installazione ultimata avrete nel menù Start un collegamento a Code::Blocks.

Per chi, invece, usasse Windows 8/8.1 troverete una icona sul Desktop e un collegamento nell'interfaccia Metro; se non la trovate semplicemente recatevi nell'interfaccia Metro (premendo il tasto ”Windows” sulla tastiera) e digitare ”Code::Blocks” per ricercarlo.



Figura 2: Per avviare Code::Blocks in Windows 8/8.1 utilizzate la funzione ”cerca”.

<sup>4</sup>Minimalist GNU for Windows, <http://www.mingw.org/>

### 3 Installare Code::Blocks: Linux

La seguente sezione riguarda l'installazione di Code::Blocks in Linux, distribuzioni Ubuntu/Ubuntu-based (ad esempio Linux Mint).

Per prima cosa facciamo un aggiornamento del sistema che fa sempre bene:

```
$ sudo apt-get update
$ sudo apt-get upgrade
```

Successivamente dovremo accertarsi di avere installato il compilatore (“gcc”<sup>5</sup>) e i tool correlati:

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

Ed infine possiamo installare Code::Blocks:

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

Terminata l'installazione verrà creata nel menù un link all'applicazione.

### 4 Installare Code::Blocks: OsX

La seguente sezione riguarda l'installazione di Code::Blocks in ambiente OsX; la versione impiegata per la guida è Mavericks 10.9.

1. Scaricate dal sito <http://www.codeblocks.org/> il file di setup; lo trovate seguendo il percorso:

”Downloads” → ”Download the binary release” → ”Mac OS X”

2. Troverete nella cartella Download il file applicazione ”Code::Blocks”; per installarlo è sufficiente spostarlo in ”Applications”.

---

<sup>5</sup>Gnu C Compiler, <https://gcc.gnu.org/>

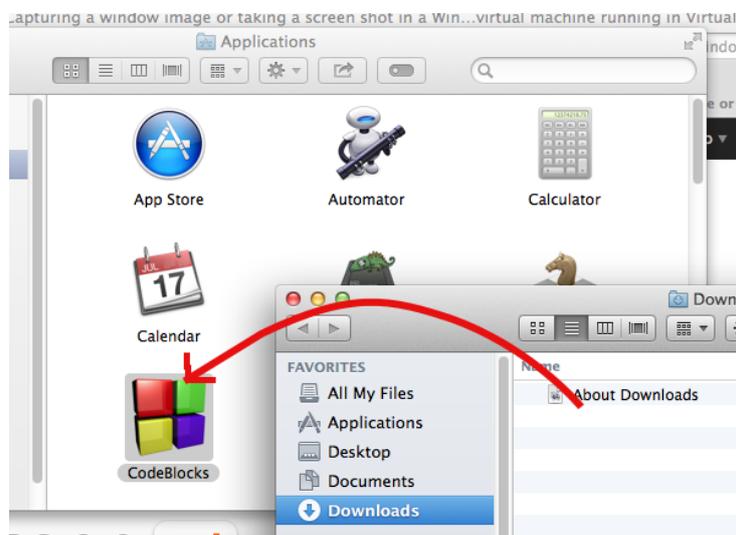


Figura 3: Per installare Code::Blocks basta spostare il file scaricato nella cartella "Applications"

3. Avviate Code::Blocks con un doppio click. A questo punto potrebbe darvi problemi e non avviare il programma; questo perchè Code::Blocks non è stato installato da un canale ufficiale, ma lo avete scaricato da Internet.

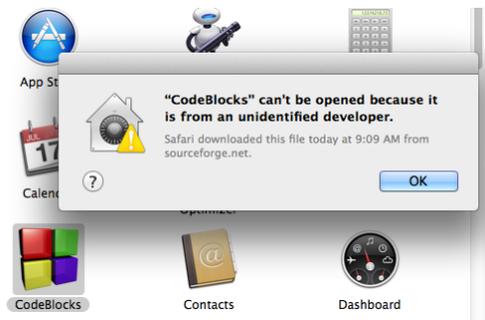


Figura 4: Errore di sicurezza nell'avvio di Code::Blocks

4. Avete due modi per ovviare a questo problema; il primo richiede che facciate click sul file con il pulsante destro del mouse e clicchiate su "Apri"; vi chiederà di confermare, date "Ok". Il programma si avvia normalmente, ma con questa modalità ogni volta dovrete aprirlo come appena fatto ed inoltre potrebbe di tanto in tanto chiedervi la conferma per svolgere certe operazioni. Se usate spesso Code::Blocks vi consigliamo di seguire questo secondo metodo:

5. Aprite le "Impostazioni di sistema", cliccando su "System Preferences" dal menù principale.

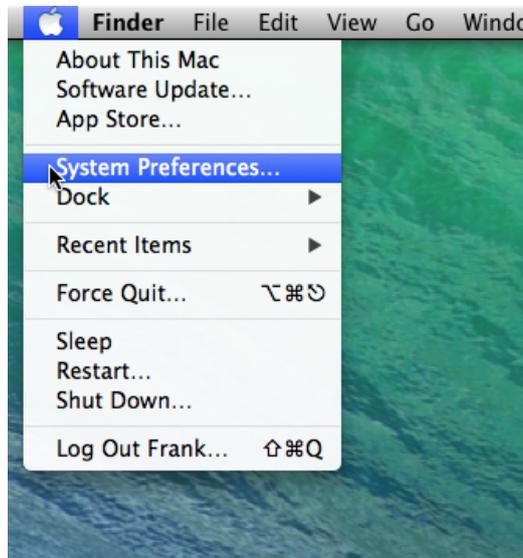


Figura 5: Aprite le impostazioni di sistema

6. Cercate "Security and privacy". Le impostazioni che ci servono sono oscurate; per poterle abilitare dovete cliccare sul lucchetto in basso a sinistra e inserire quindi nome utente e password (per ottenere privilegi di amministratore).

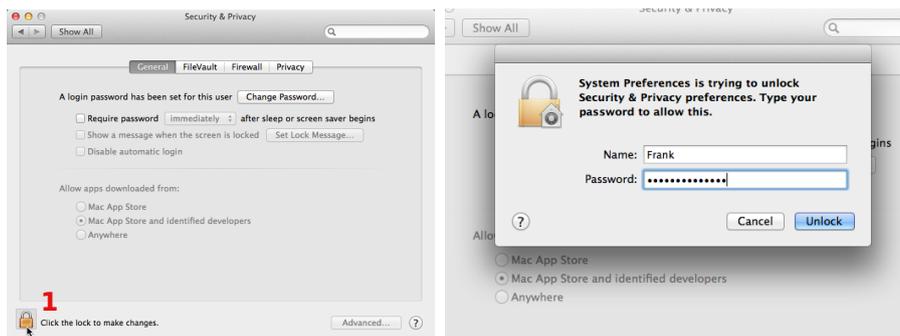


Figura 6: Accedete come amministratore di sistema per sbloccare le impostazioni ricercate

7. Ora possiamo impostare "Anywhere" nel menù "Allow apps downloaded from"; vi chiederà di confermare, fatelo.



Figura 7: Impostate come mostrato in figura; questo vi permetterà di eseguire ogni applicazione che installerete.

8. Eseguite Code::Blocks; il compilatore (gcc) dovrebbe essere già installato nel sistema. Quando richiesto selezionatelo; è necessario però perchè Code::Blocks funzioni a dovere un ulteriore software chiamato "command line developer tools"; in automatico vi comparirà un pop-up chiedendovi cosa fare. Cliccate su "Install".

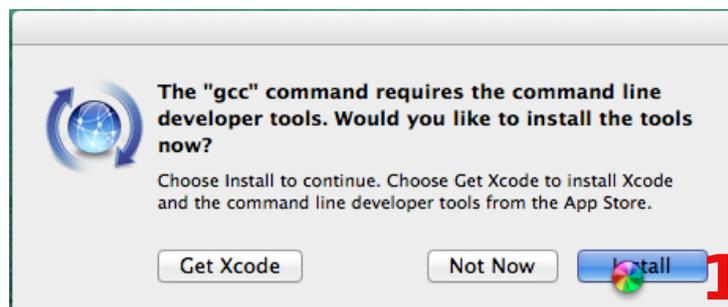


Figura 8: Date conferma per installare tutti i tool necesarri per Code::Blocks

9. A questo punto Code::Blocks è pronto.

## 5 Cominciare a programmare

Siete ora pronti per cominciare a programmare.

Eseguite Code:Blocks. Al primo avvio vi chiederà di scegliere da una lista quale compilatore impiegare; se avete seguito correttamente questa guida sarà selezionato in modo automatico GCC (essendo l'unico installato); confermate premendo su "Ok".

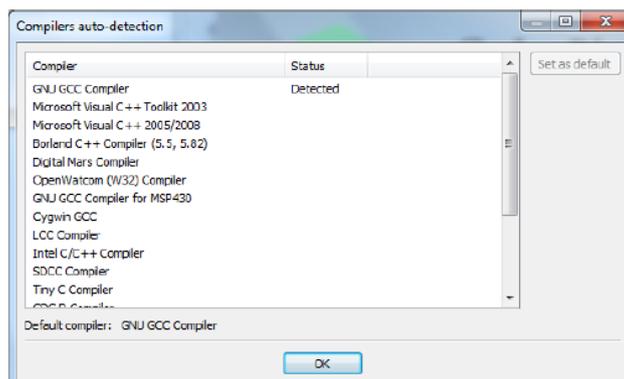


Figura 9: La lista dei compilatori; selezionate GCC e confermate con "Ok"

A questo punto dovrebbe aprirsi il programma e portarvi nella schermata principale; al primo avvio potrebbe essere necessario qualche secondo extra prima che il programma sia pronto, in particolare modo sui sistemi Mac.

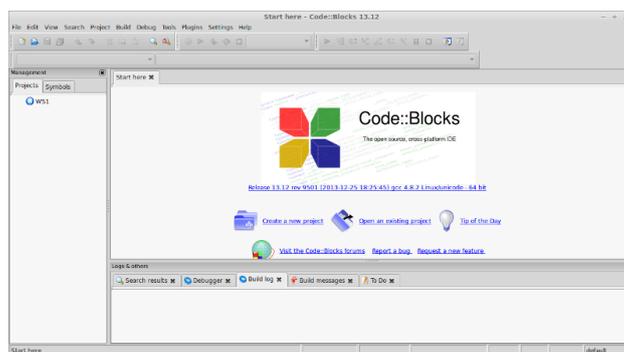


Figura 10: La schermata di avvio di Code::Blocks

In questa schermata, a sinistra avrete la lista dei "Workspace". Un Workspace non è altro che una raccolta di progetti in qualche modo legati tra loro.

Un progetto è invece un insieme di file contenenti codice sorgente da compilare assieme per poter essere eseguiti correttamente.

Generalmente per questo corso un progetto conterrà un solo file, mentre un workspace avrà al suo interno più progetti, ognuno dedicato ad un problema da risolvere.

Per cominciare a programmare ci sono due differenti modalità:

1. (Sottosezione 5.2) Il più semplice e veloce consiste nel creare un nuovo file vuoto, al cui interno scriveremo il nostro codice, salvarlo con estensione ".c", compilare il programma ed eseguirlo; stiamo in questo caso tralasciando il discorso dei workspace e dei progetti.
2. (Sottosezione 5.1) Il più completo, certamente necessario per la programmazione di software più complessi, consiste nel creare un nuovo progetto, aggiungervi tutti i file di codice sorgente necessari, compilare ed eseguire tutto il progetto. Questo secondo metodo permette di avere una migliore organizzazione dei file in fase di programmazione e soprattutto mette a disposizione il debugger, un utile strumento per trovare e risolvere bug (malfunzionamenti) del vostro programma.

A voi la scelta a seconda delle vostre esigenze.

## 5.1 Come creare un progetto

All'apertura di Code::Blocks troverete un Workspace di default, vuoto; potete aggiungervi nuovi progetti seguendo i passi di seguito descritti:

1. Cliccate su "File" → "New" → "Project"

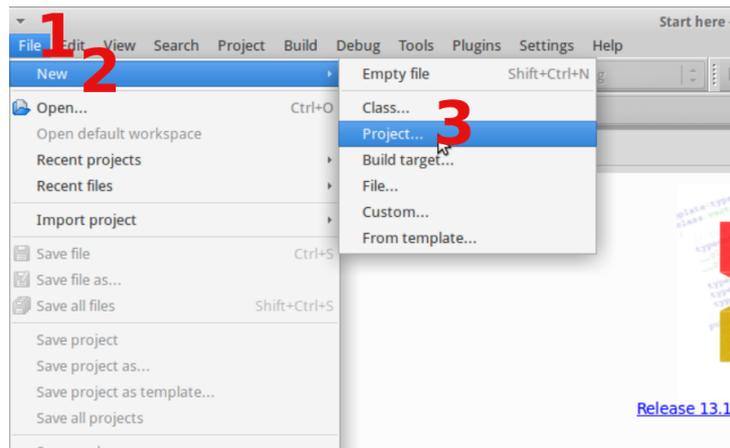


Figura 11: Sequenza per la creazione di un nuovo progetto

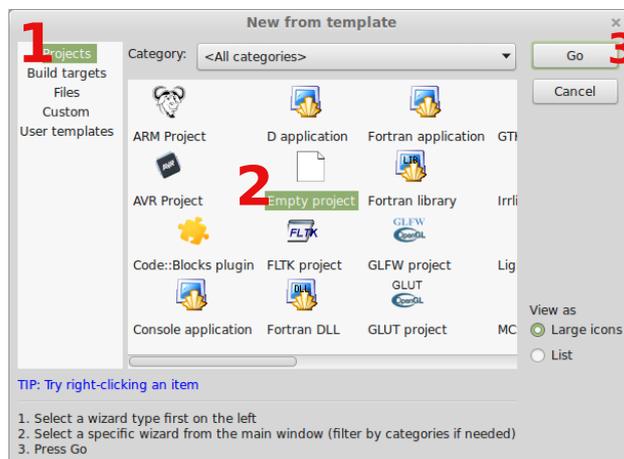


Figura 12: Da questa schermata scegliete Empty project o Console application

2. Scegliete "Empty project" (o "Console Application" se volete il main del vostro programma preimpostato)
3. Inserite un nome per il vostro progetto ed una directory dove salvarlo. **Consiglio:** evitate nomi e percorsi con spazi o caratteri speciali, ed usate sempre nomi evocativi!
4. Nell'ultima schermata impostate tutto come da immagine 13.

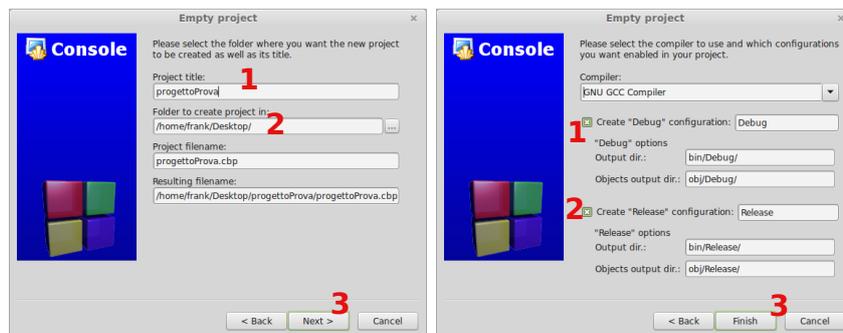


Figura 13: Nelle ultime due schermate impostate un nome, la directory del progetto e impostate tutto come da immagine

5. A questo punto potete aggiungere tutti i file di cui avete bisogno. Cliccate su "File" → "New" → "Empty File".
6. Vi chiederà se volete aggiungere il file al progetto attivo. Confermate con "Yes".

7. Nella finestra che si aprirà dovrete scegliere il nome del file; la posizione è già preimpostata, non cambiatela. **Consiglio:** utilizzate sempre nomi significativi, senza spazi e caratteri speciali. Ricordatevi di aggiungere alla fine del nome l'estensione “.c”, come mostrato nelle seconda delle seguenti figure.
8. Nell'ultima schermata selezionate entrambe le voci (“Debug” e “Release”) e confermate con “Ok”

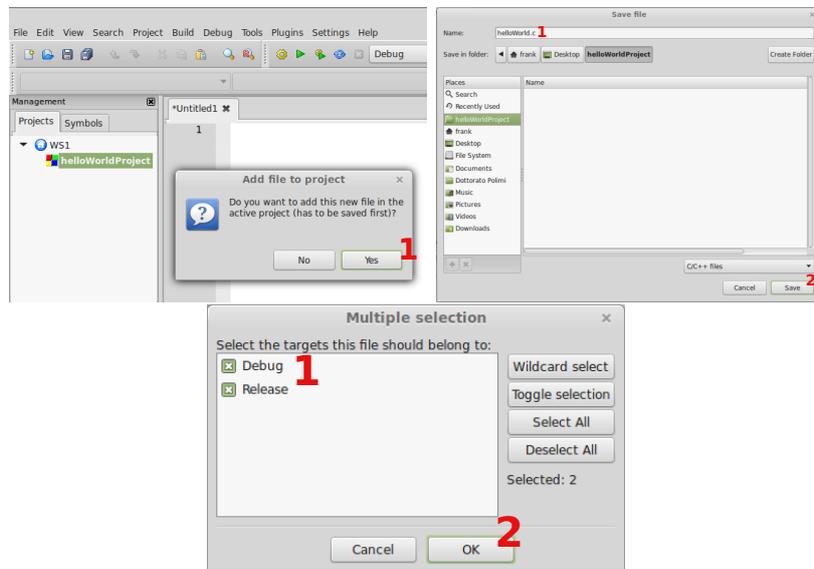


Figura 14: Procedura per aggiungere un nuovo file al progetto corrente

9. Nella parte destra del programma si aprirà una nuova scheda con il nome del vostro file; qui dentro dovrete scrivere il vostro programma.

## 5.2 Come creare un file semplice

Se non vi interessa creare un progetto, potete cominciare a programmare partendo da un file singolo; la procedura è la seguente:

1. Cliccate su "File" → "New" → "Empty File"
2. Nella parte destra del programma si aprirà una nuova scheda intitolata "untitled1"; qui dentro dovrete scrivere il vostro programma.

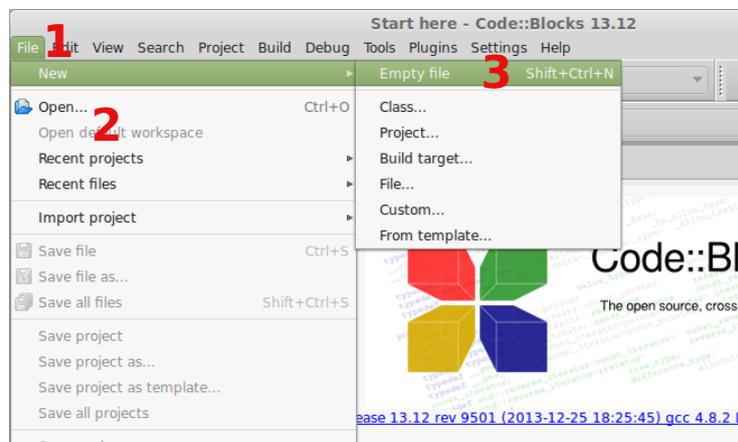


Figura 15: Sequenza per la creazione di un file singolo

3. Una volta terminato, prima di poter compilare, dovete salvare il file; potete o cliccare sul dischetto (singolo o doppio è indifferente) o cliccare su "File" → "Save files" ("Save file as" e "Save all files" hanno funzioni simili). Nella finestra che si aprirà dovrete scegliere dove salvare il file e il suo nome. **Consiglio:** utilizzate sempre nomi significativi, senza spazi e caratteri speciali. Ricordatevi di aggiungere alla fine del nome l'estensione ".c", come da immagine 16.

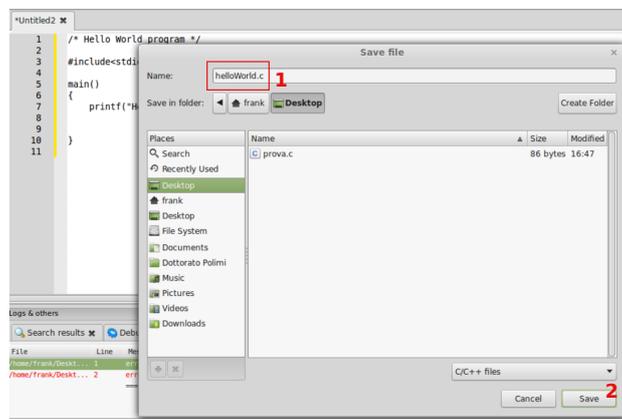


Figura 16: Quando salvate un file usate sempre nomi significativi; salvare con estensione .c

## 6 Compilare ed eseguire il programma

Una volta che avrete terminato di scrivere il vostro programma, questo va compilato e lanciato.

1. Per compilarlo cliccate su “Build”, o dalla barra degli strumenti o dal menù “Build”, cliccando sull'icona con l'ingranaggio;
2. Per eseguirlo cliccate su “Run” (l'icona con la freccia verde);

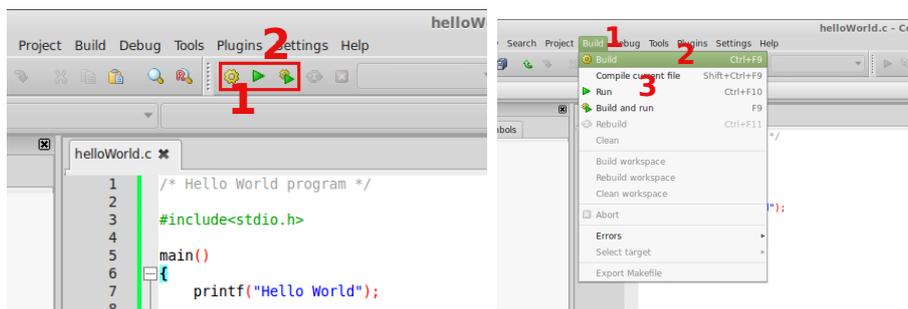


Figura 17: Come compilare ed eseguire un programma.

3. Durante la compilazione, nella sezione inferiore di Code::Blocks viene stampato il riepilogo di questa fase; se non ci sono errori nel codice il processo termina e viene indicato “0 error”; altrimenti il file da eseguire non viene creato e sono elencati gli errori presenti nel codice. In questo caso leggete attentamente il log: le indicazioni scritte vi indicheranno con precisione dove e quali errori sono stati commessi (figura 19).

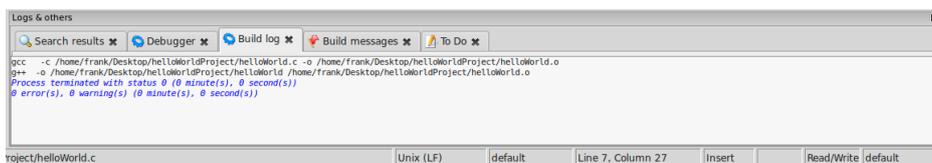


Figura 18: Il log della compilazione; nessun errore è stato rilevato.

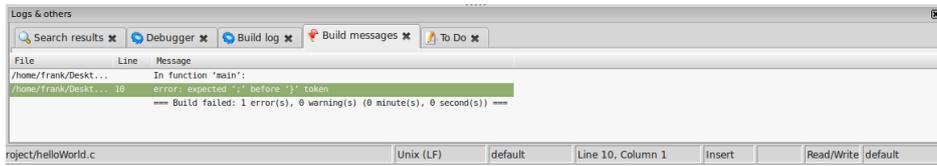


Figura 19: Il log della compilazione; il codice sorgente presente errori di sintassi.

## 7 Comandi rapidi utili

Di seguito una lista di scorciatoie da tastiera:

- "Ctrl+F9" → Compila tutto il vostro progetto o il singolo file;
- "Shift+Ctrl+F9" → Compila solo il file corrente attivo;
- "Ctrl+F10" → Esegue il programma compilato;
- "F9" → Compila ed esegue il progetto o il singolo file corrente attivo;
- "Ctrl+F11" → Ricompila tutto il vostro progetto o il singolo file;
- 'F8" → Start/Continue, avvia il debugger;
- 'Shift+F8" → Stop debugger, interrompe il debugger;
- "F4" → Run to cursor, vai al cursore;
- "F7" → Next Line, vai a prossima istruzione;
- "Shift+F7" → Step Into, vai a prossima istruzione; entra nelle sotto-funzioni;
- "Ctrl+F7" → Step Out, vai a prossima istruzione; ritorna alla funzione chiamante.