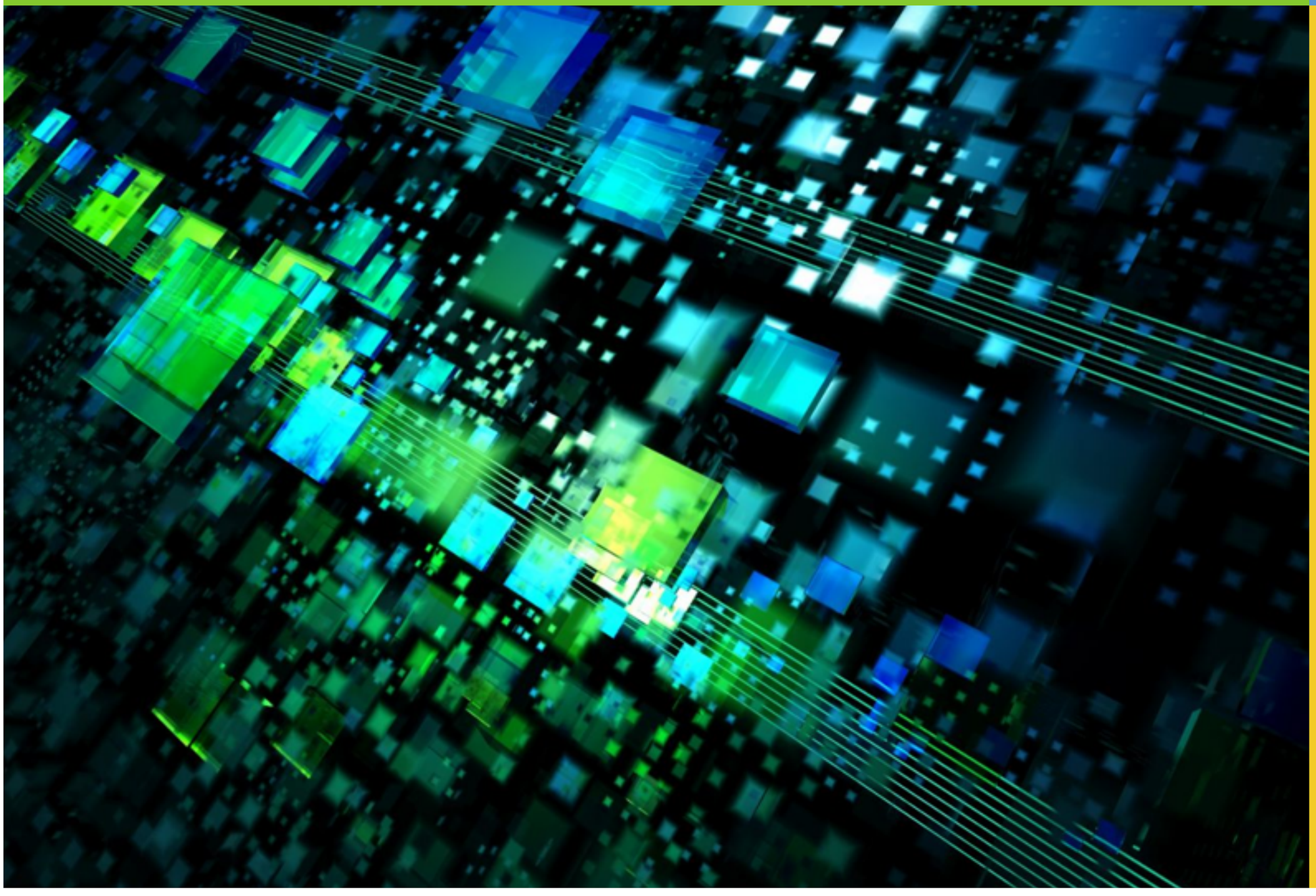
Programação de Computadores
Primeiros Passos Práticos em C++



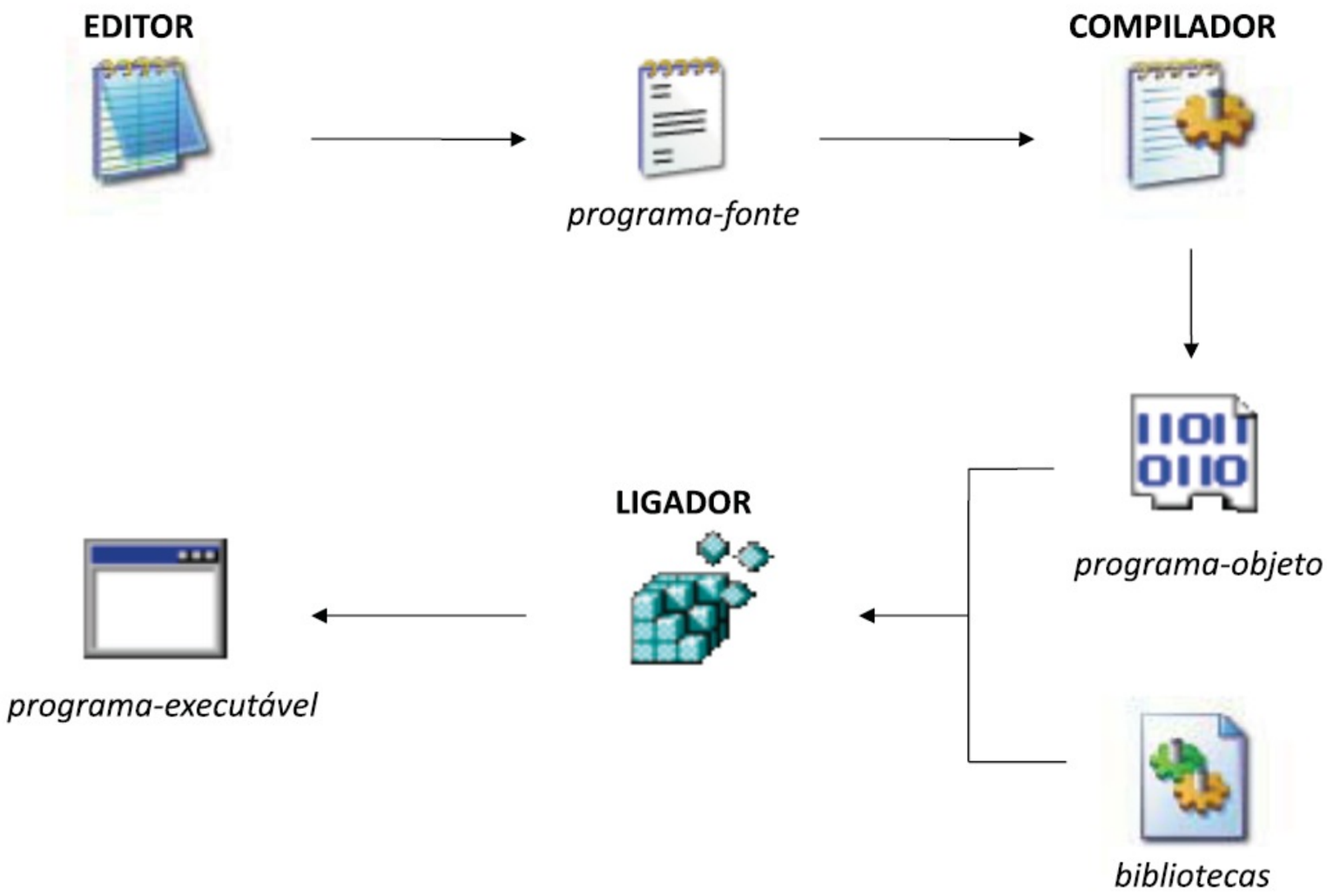
Alan de Freitas

Marco Antonio Carvalho

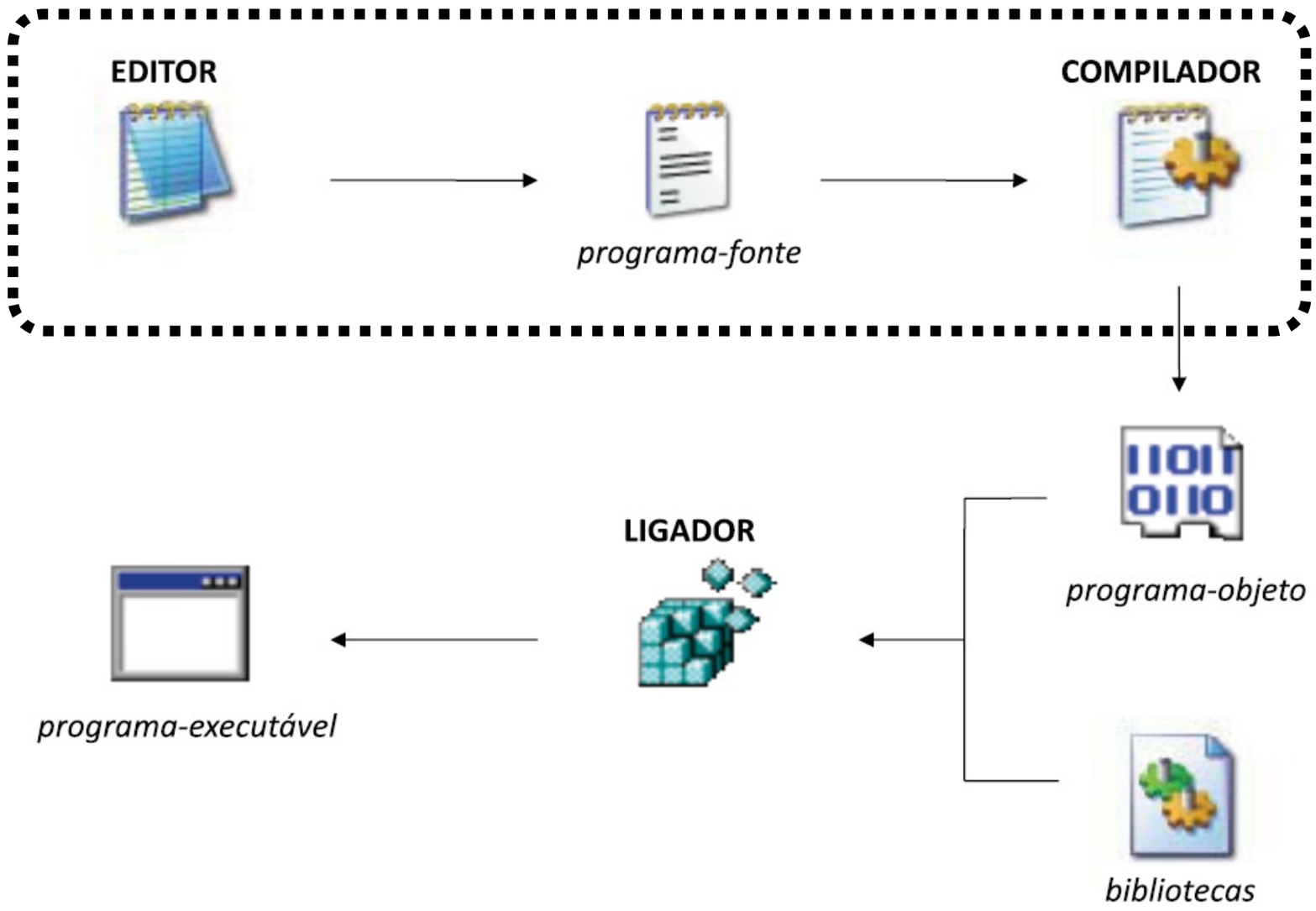
Compilação de Programas

- O compilador é o responsável por transformar nosso código em um programa propriamente dito.
- Neste exercício vamos compilar nosso primeiro programa.

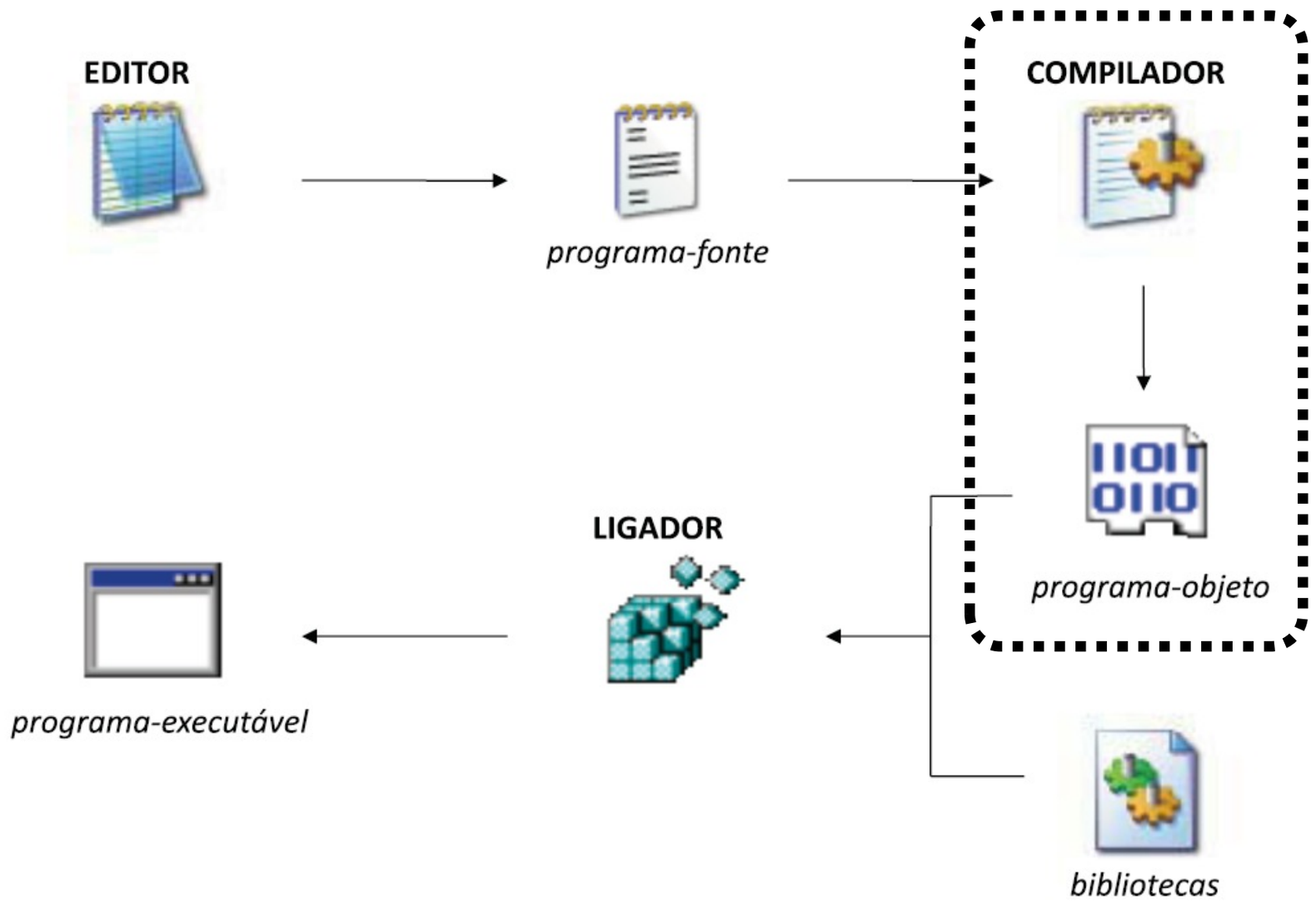
Esse desenho representa o processo básico de compilação de um programa em C++



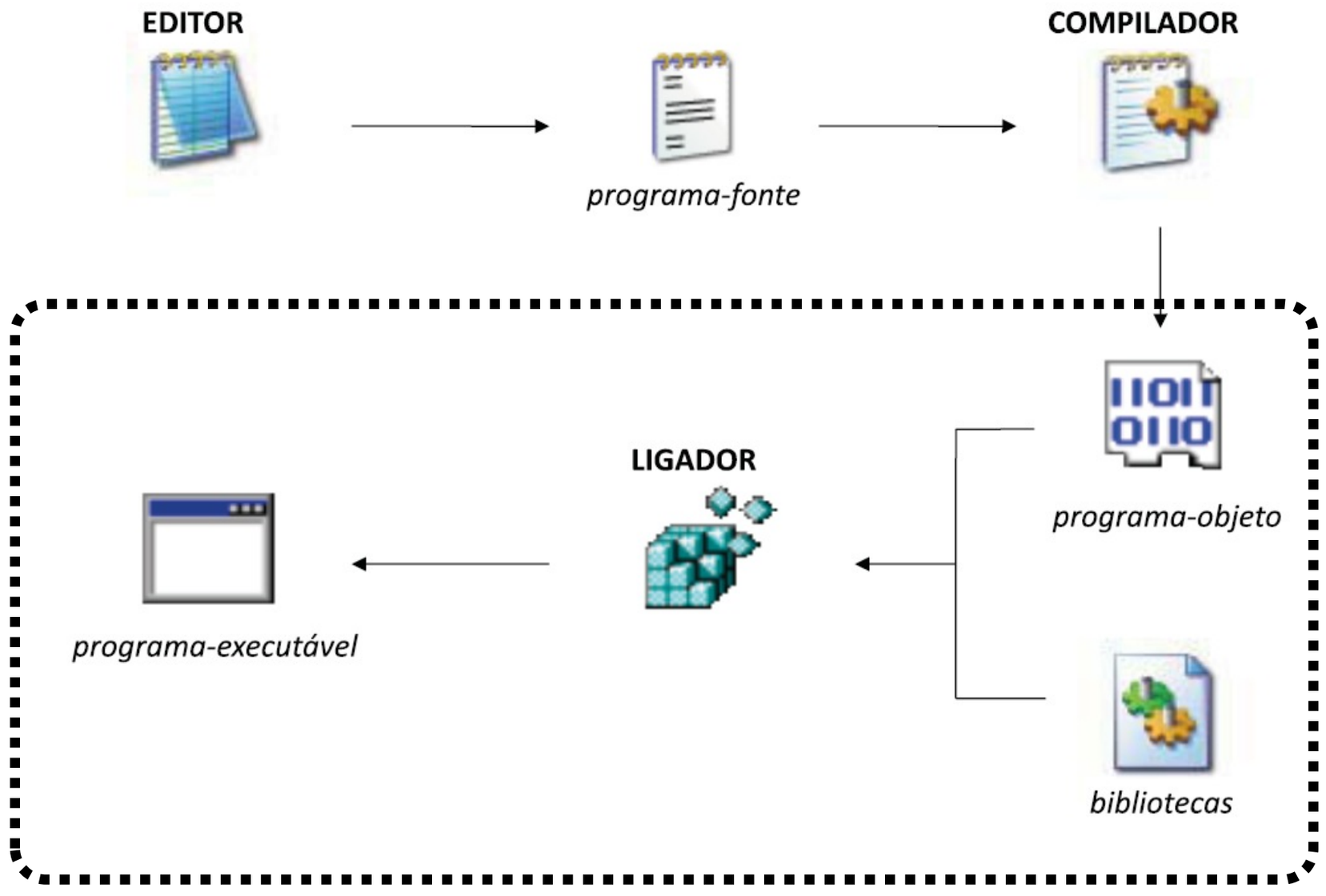
Fazemos o nosso código em um editor e o enviamos para um compilador.



O compilador gera um programa-objeto.



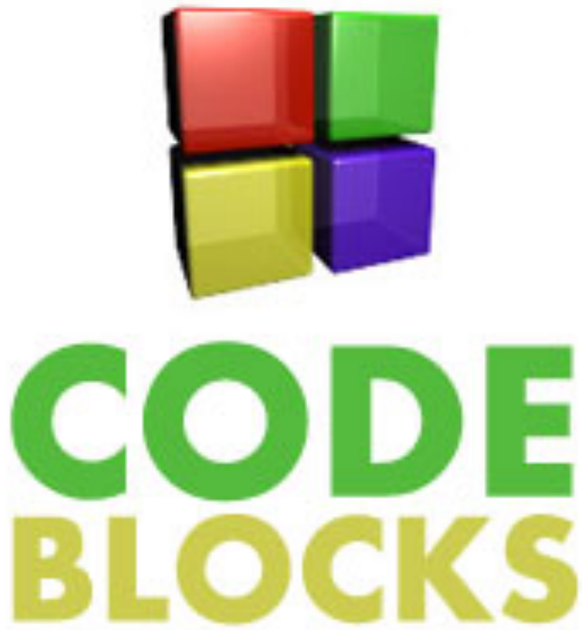
O programa-objeto e bibliotecas do sistema são unidas por um ligador, que gera um programa executável.

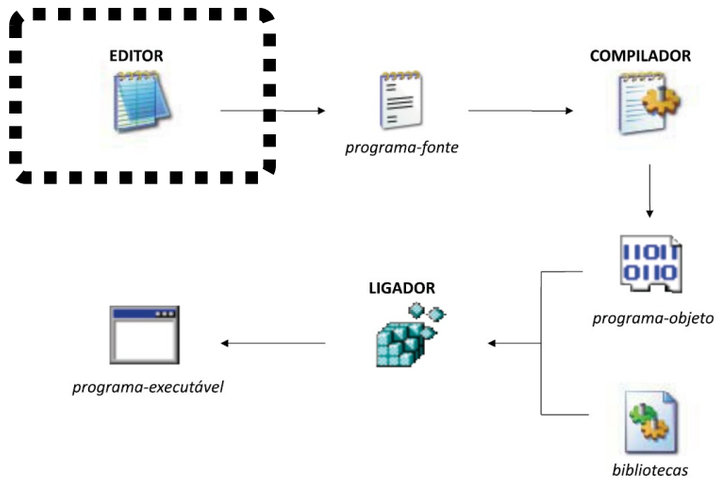


Code::Blocks

- Code::Blocks é um ambiente de desenvolvimento que conta com as ferramentas para geração de programas.

- Ele fará o papel do editor em nosso esquema de compilação





Download do Code::Blocks

- Se você está estudando em casa:
  - Nosso primeiro passo será então fazer o download do Code::Blocks
- O Code::Blocks já está instalado nos laboratórios do DECOM
- O Code::Blocks está disponível gratuitamente na internet no website:
  - www.codeblocks.com

Acesse www.codeblocks.com e clique em *Downloads*

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
- BugTracker
- PatchTracker
- Browse SVN
- Browse SVN log

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.

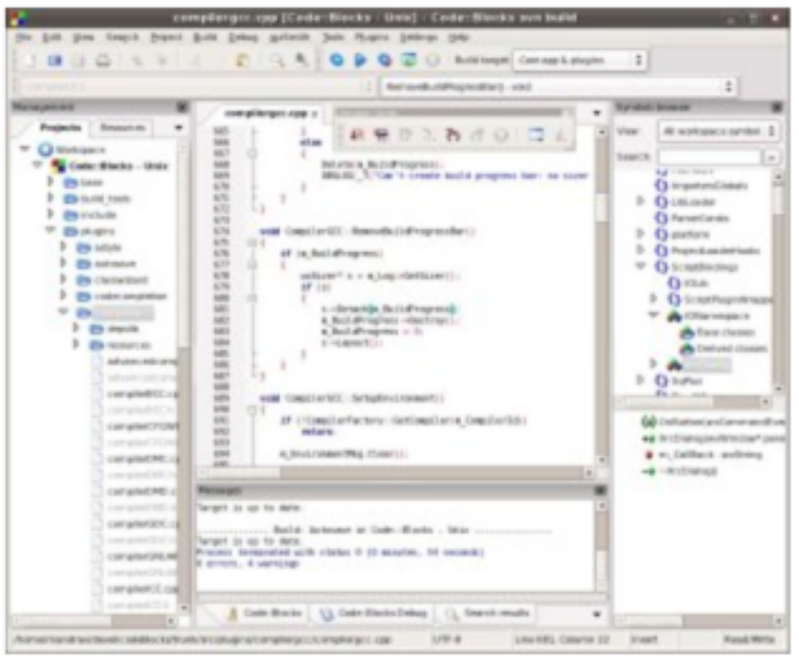
Finally, an IDE with all the features *you* need, having a consistent look, feel and operation across platforms.

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



Code::Blocks 13.12 is here!

Clique em *Download the binary release* para baixar o instalador em forma executável.

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
- BugTracker
- PatchTracker
- Browse SVN

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, 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** or (for Debian and Fedora users) in **Jens' Debian repository** and **Jens' Fedora repository**. Other distributions usually follow provided by the community (Big "Thank you" for that!) Please note that we consider nightly builds to be *stable*, usually, unless stated otherwise.

- **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!

Escolha o seu sistema operacional...

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
- BugTracker
- PatchTracker

Please select a setup package depending on your platform:

- **Windows 2000/XP/Vista/7//8**
- **Linux 32-bit**
- **Linux 64-bit**
- **Mac OS X**

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.

MIRRORS: BerliOS mirrors all files *usually* at SourceForge using a "BerliOS robot" **here**. As this sometimes doesn't work, we have mirrored all file releases at SourceForge, too **here**. The latter is managed by us.

IMPORTANT NOTE: If you try to download from BerliOS and get a "Too many clients!" - error, you should retry to download the file. According to a BerliOS - admin, this can happen several times, before the download starts. Alternatively use on of the mirrors.

NOTE: We have a **Changelog for 13.12**, that gives you an overview over the enhancements and fixes we have put in the new release.

Windows 2000 / XP / Vista / 7:

Para Windows, baixe a versão *mingw-setup*.

- Forums
- Forums (mobile)
- Nightlies
- BugTracker
- PatchTracker
- Browse SVN
- Browse SVN log

Windows 2000 / XP / Vista / 7:

File	Date	Download from
codeblocks-13.12-setup.exe	27 Dec 2013	**BerliOS** or **Sourceforge.net**
codeblocks-13.12mingw-setup.exe	27 Dec 2013	**BerliOS** or **Sourceforge.net**
codeblocks-13.12mingw-setup-TDM-GCC-481.exe	27 Dec 2013	**BerliOS** or **Sourceforge.net**

NOTE: The codeblocks-13.12mingw-setup.exe file *includes* the GCC compiler and GDB debugger from **TDM-GCC** (version 4.7.1, 32 bit). The codeblocks-13.12mingw-setup-TDM-GCC-481.exe file includes the TDM-GCC compiler, version 4.8.1, 32 bit. While v4.7.1 is rock-solid (we use it to compile C::B), v4.8.1 is provided for convenience, there are some known bugs with this version related to the compilation of Code::Blocks itself.

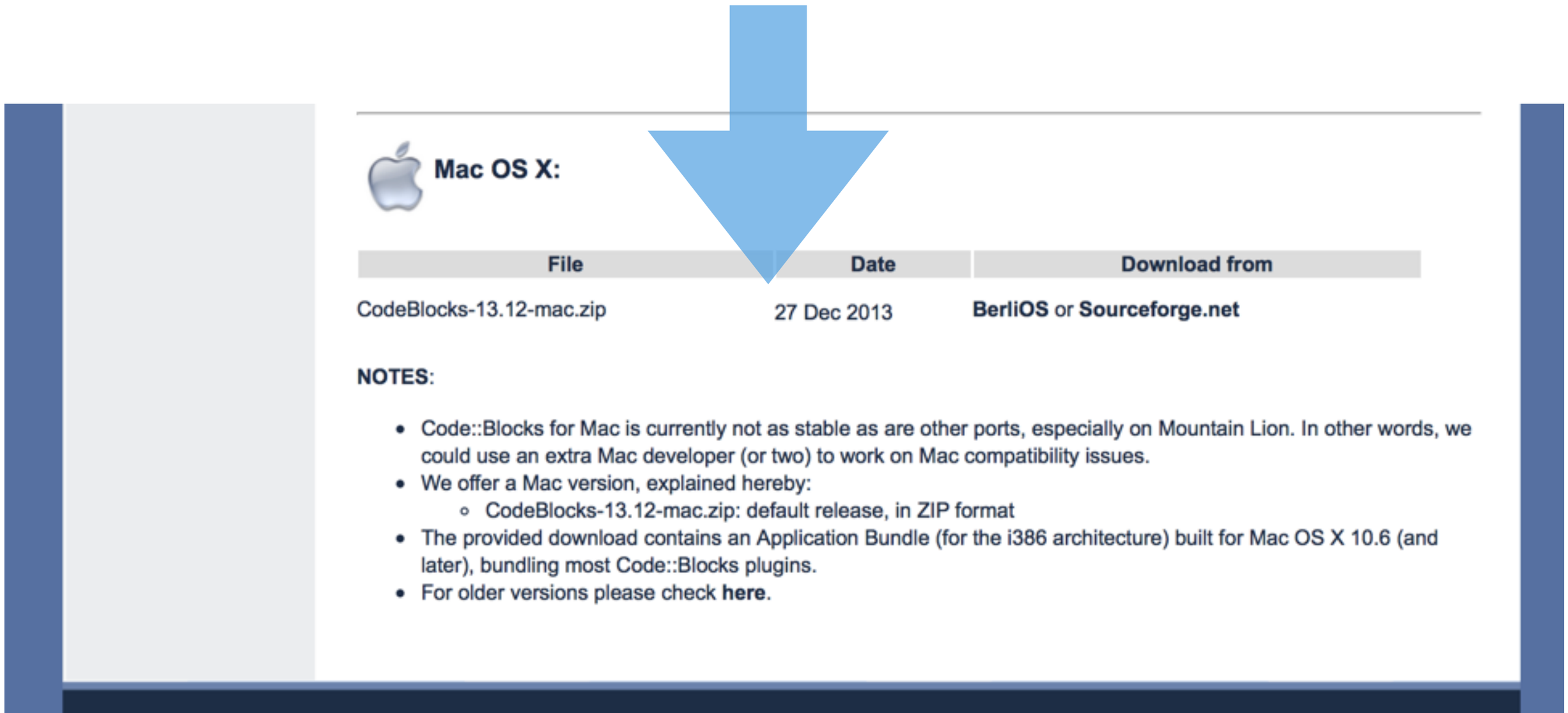
IF UNSURE, USE "codeblocks-13.12mingw-setup.exe"!

Linux 32-bit:

Distro	File	Date	Download from
	codeblocks-13.12-1_i386.debian.stable.tar.xz	27 Dec 2013	**BerliOS** or **Sourceforge.net**
	codeblocks-13.12-1_i386.debian.testing.tar.xz	27 Dec 2013	**BerliOS** or **SourceForge.net**
	codeblocks-debuginfo-13.12-1.el18.i686.tar.xz	27 Dec 2013	**BerliOS** or **Sourceforge.net**

Se o seu sistema operacional for Linux, você deve escolher a distribuição apropriada.

Se o seu sistema operacional for Mac, há apenas uma opção.



Mac OS X:

File	Date	Download from
CodeBlocks-13.12-mac.zip	27 Dec 2013	**BerliOS** or **Sourceforge.net**

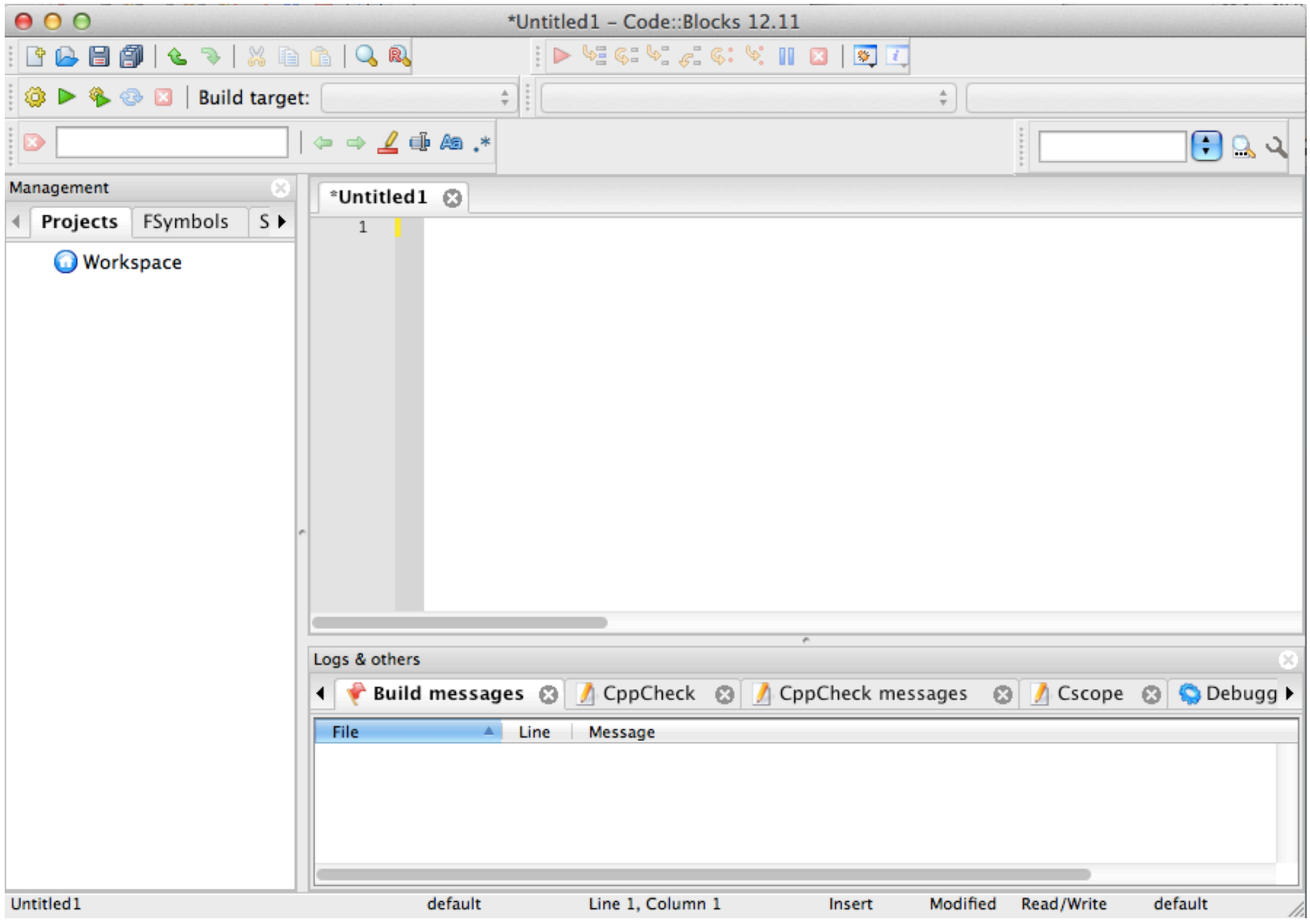
NOTES:

- Code::Blocks for Mac is currently not as stable as are other ports, especially on Mountain Lion. In other words, we could use an extra Mac developer (or two) to work on Mac compatibility issues.
- We offer a Mac version, explained hereby:
  - CodeBlocks-13.12-mac.zip: default release, in ZIP format
- The provided download contains an Application Bundle (for the i386 architecture) built for Mac OS X 10.6 (and later), bundling most Code::Blocks plugins.
- For older versions please check **here**.

Abrindo o Code::Blocks

- Abre o Code::Blocks
- Veja como ele tem um ambiente para edição de textos
- Porém, ele tem recursos específicos para edição de códigos

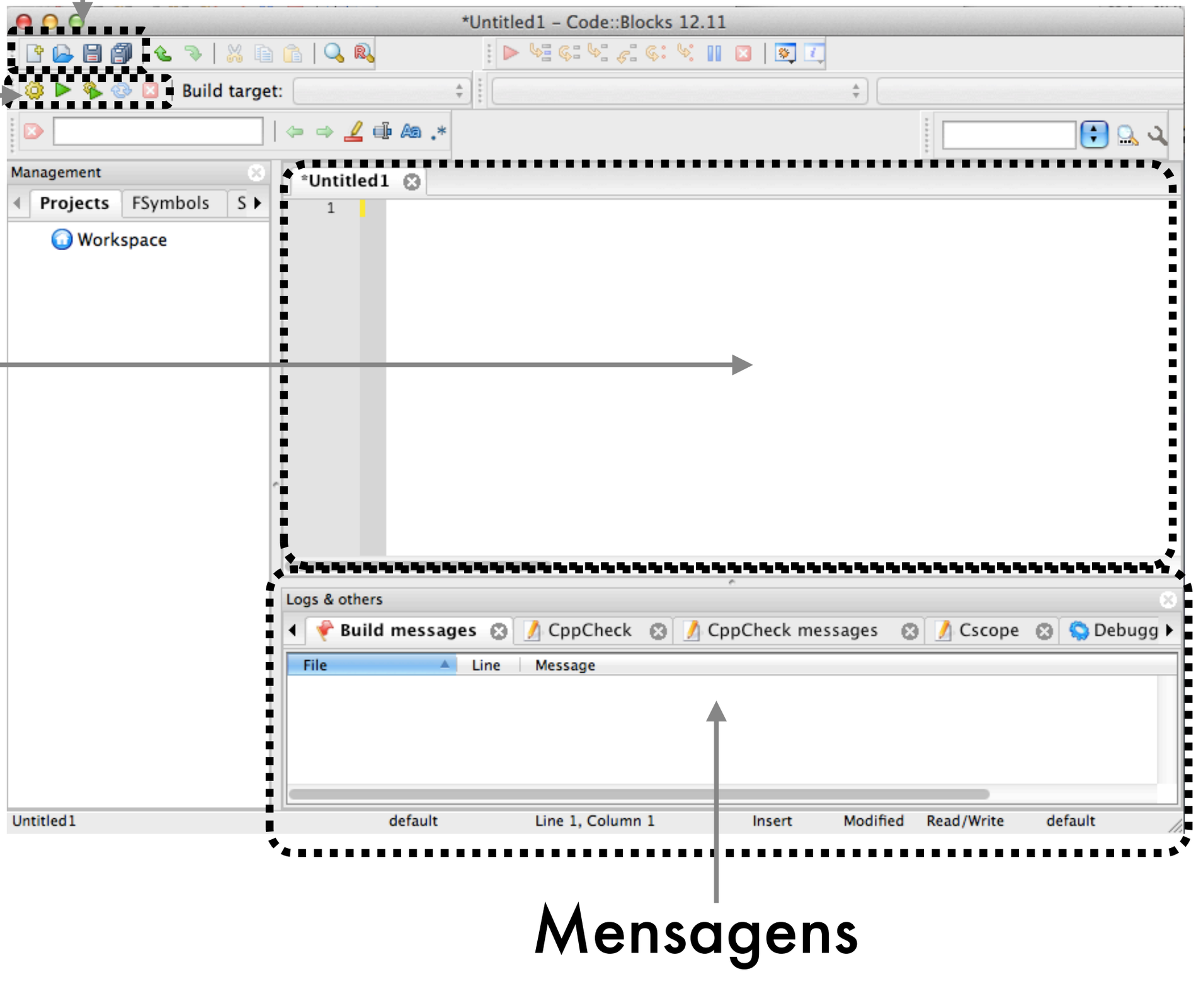
O ambiente do Code::Blocks



Arquivos

Compilação

Janela de edição do código-fonte



Mensagens

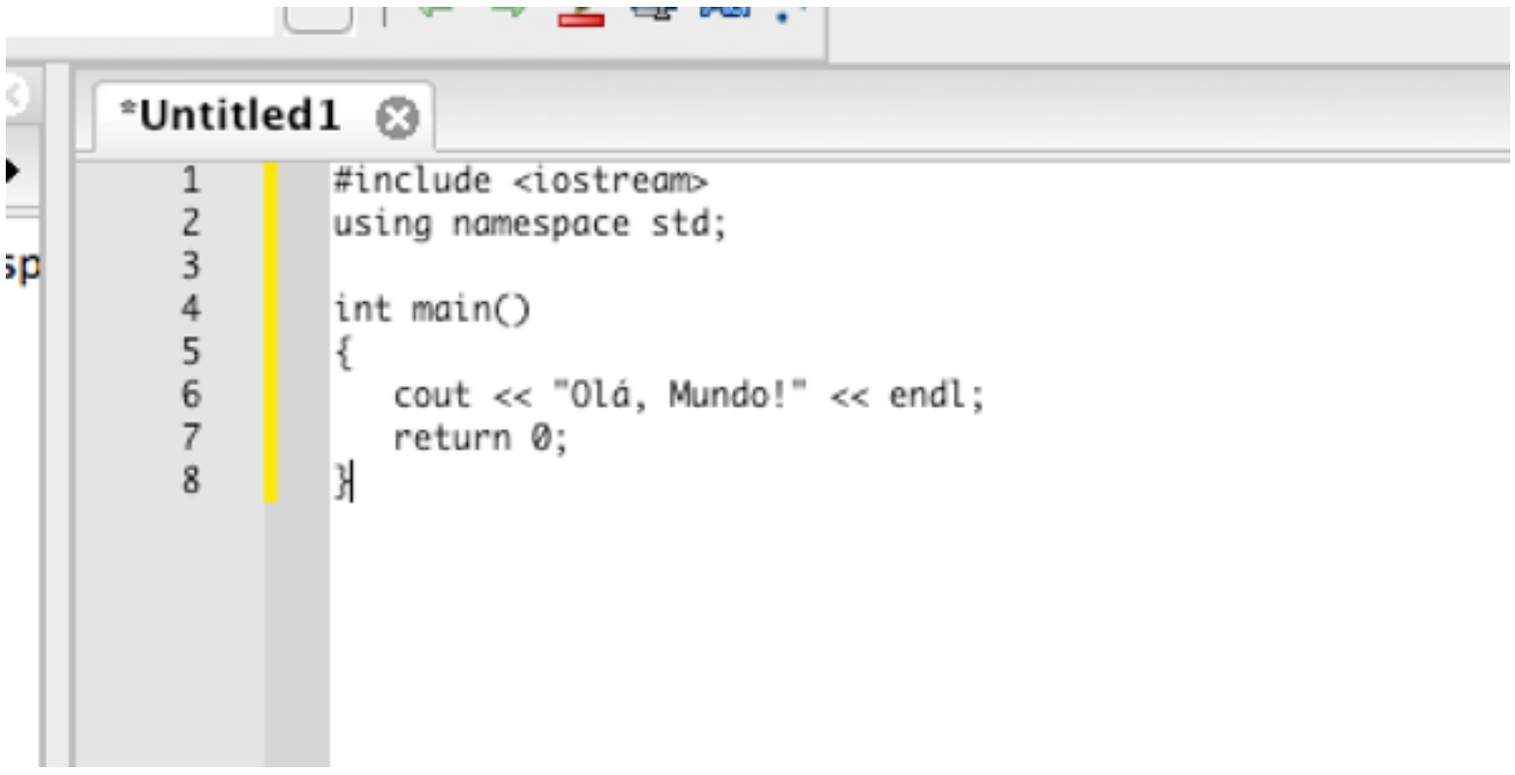
- Novo código-fonte
  - Empty File
- Abrir código-fonte existente
- Salvar
- Salvar como
- Desfazer
- Refazer
- Recortar
- Copiar
- Colar
- Localizar
- Localizar e Substituir

Clique no ícone indicado para criarmos um novo programa.

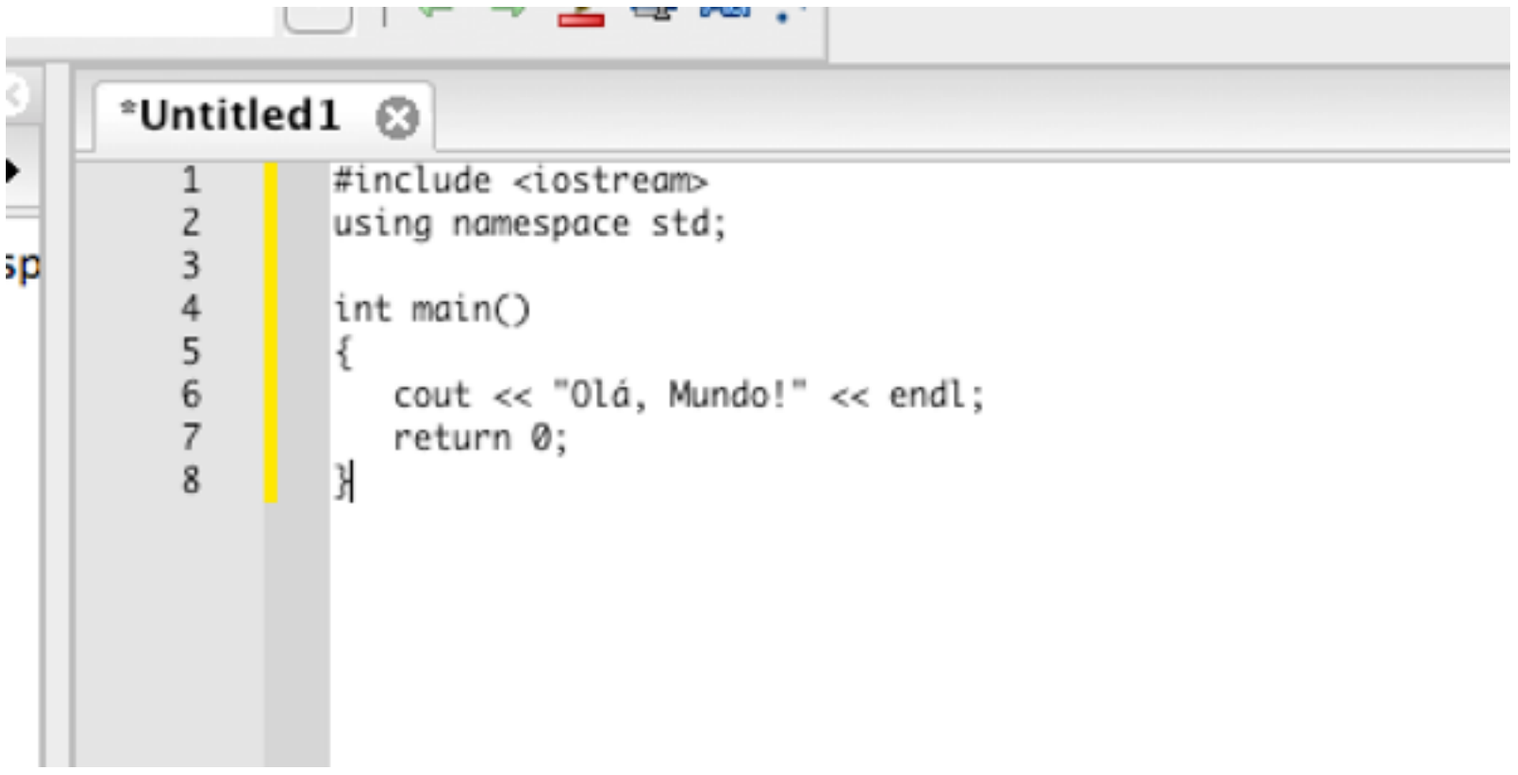
Durante o curso, utilizaremos o botão "*Novo código-fonte*", opção "*Empty File*";

Não criem projetos.

Digite o seguinte código no arquivo criado:

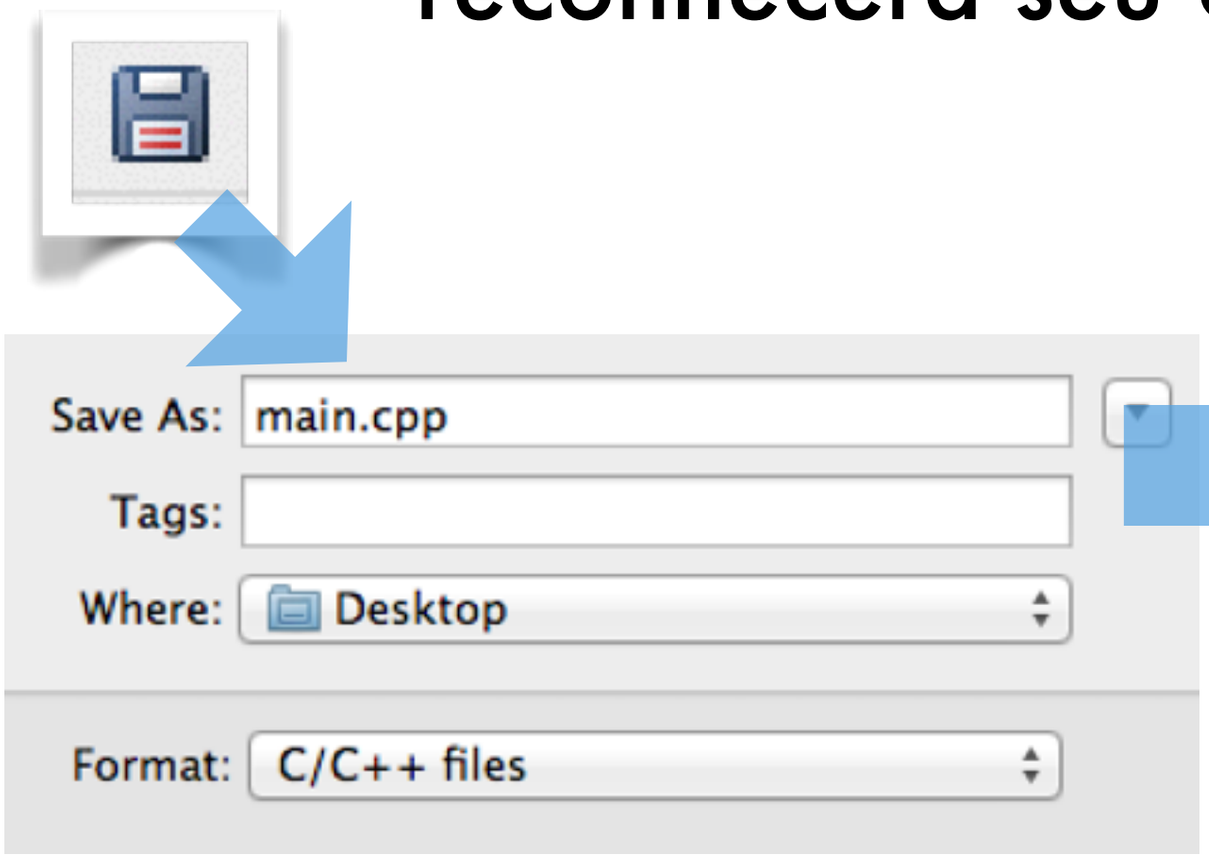


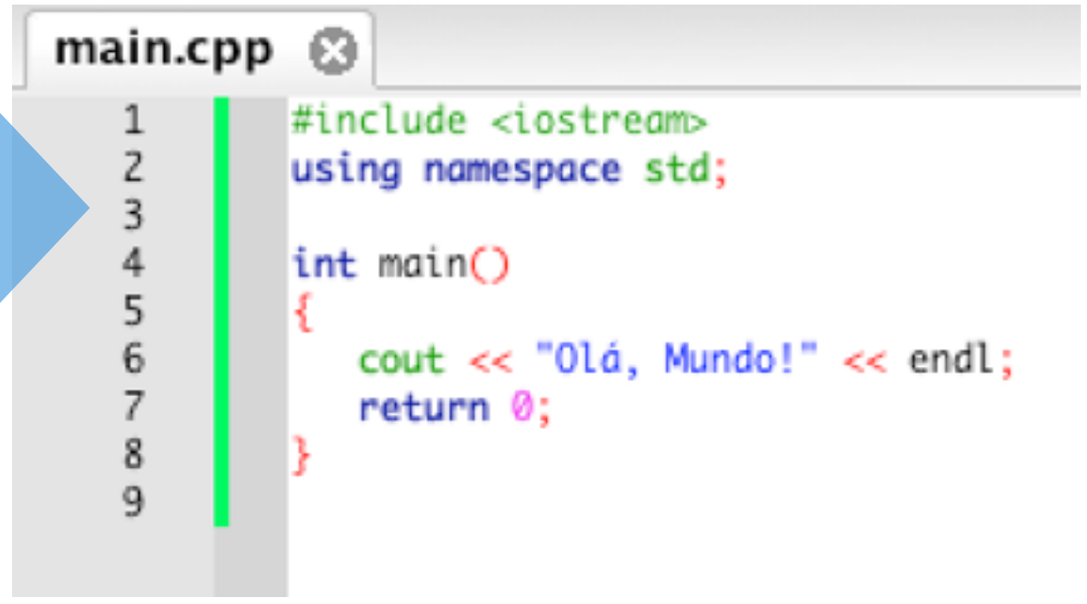
Em nosso novo arquivo, o Code::Blocks ainda não irá colorir as instruções relevantes.



Isto acontece pois o Code::Blocks ainda não sabe que este é um código de C++.

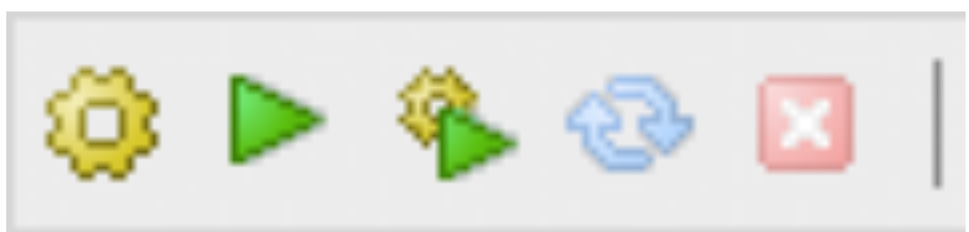
Salve o arquivo com a extensão .cpp.
Certifique-se de ter digitado .cpp explicitamente ou a extensão .c será adicionada e Code::Blocks não reconhecerá seu código como C++!





Como o arquivo agora tem a extensão .cpp, o Code::Blocks o trata como um código-fonte de C++.

Compilando um programa

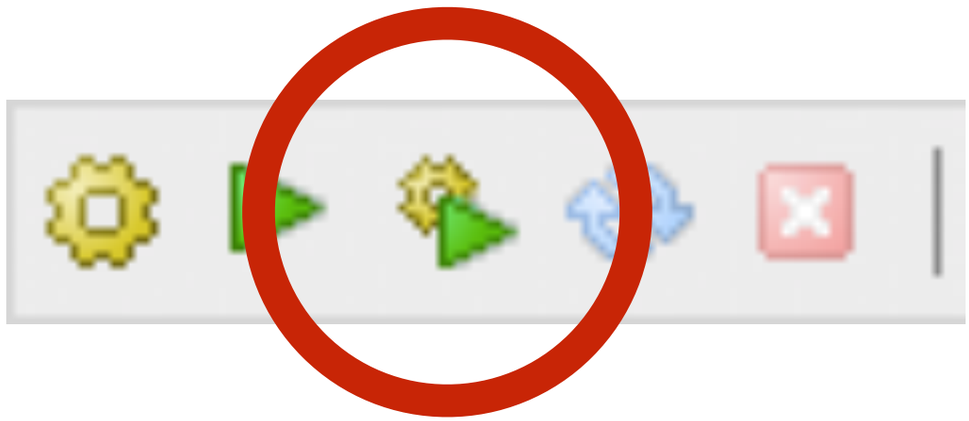


- Build (ou ctrl+f9): Compilar o código-fonte;

- Run: “Rodar”, executar o programa, depois de compilado;

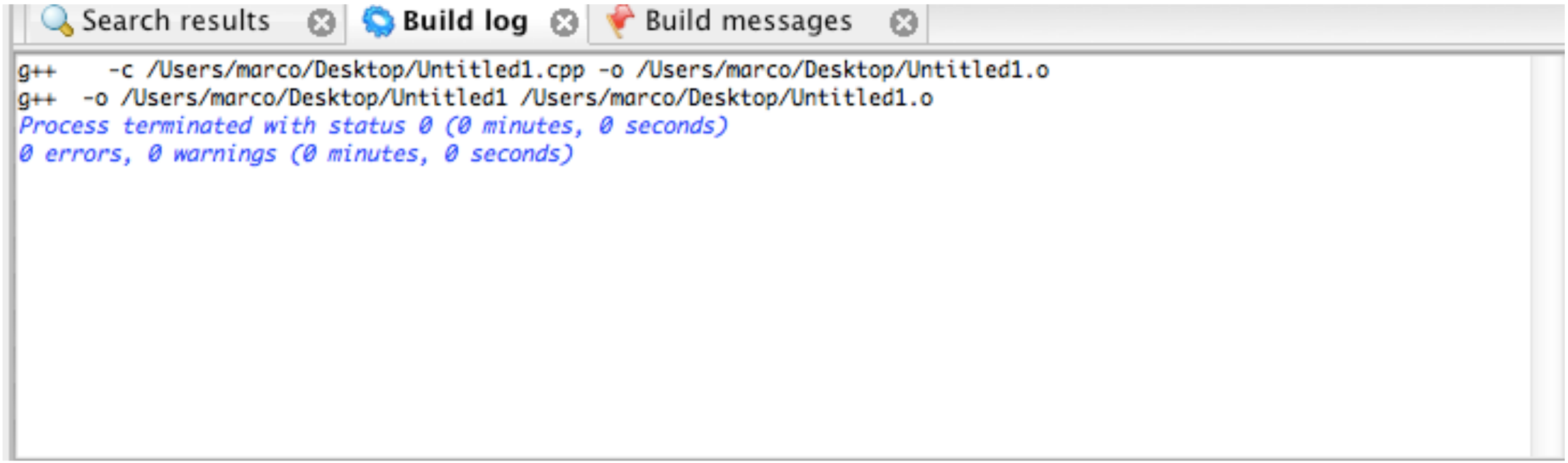
- Build and Run (ou f9): Primeiro executa o build e depois o run, automaticamente.

Compilando um programa



- Utilize a terceira opção para compilar o programa e logo em seguida executá-lo.

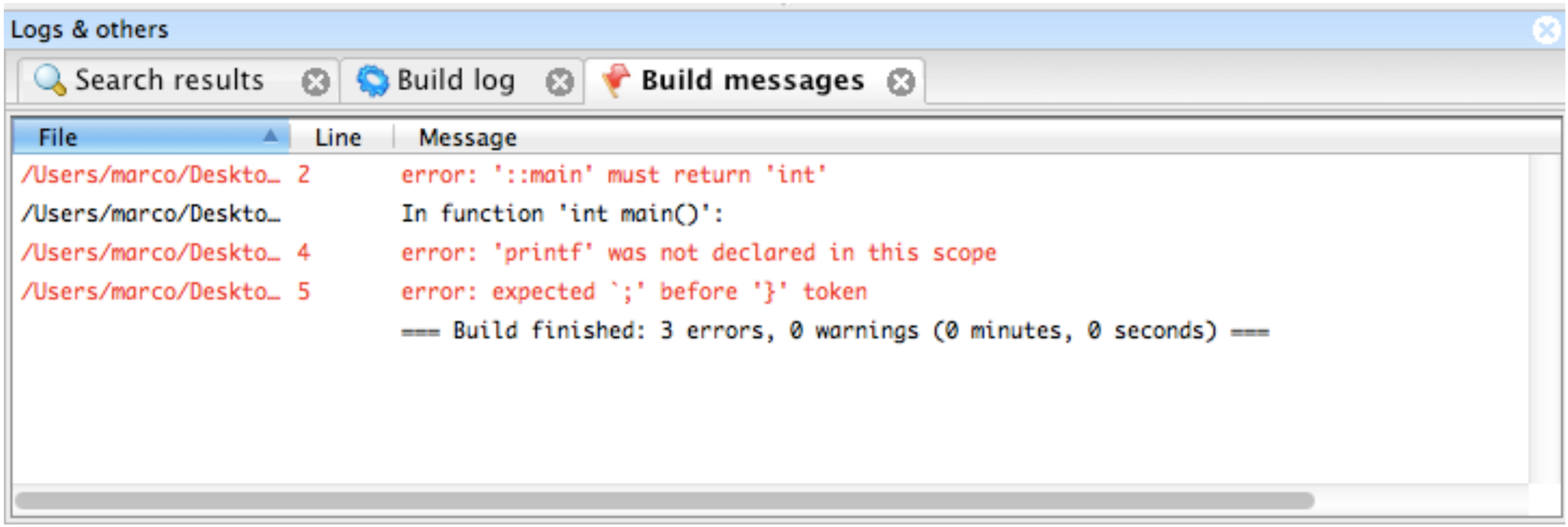
A janela de mensagens (aba Build Log) nos fornecerá informações sobre a compilação do código-fonte.



Frases azuis indicam que não houve erro;

Confira a mensagem “0 errors, 0 warnings”.

A janela de mensagens (aba Build Log) nos fornecerá informações sobre a compilação do código-fonte.



Frases vermelhas indicam que houve erro(s);

O compilador avisa em qual linha do código-fonte houve erro, devemos checá-la e corrigir o erro.

Primeiros programas

- Faremos agora nosso primeiro programa em C++
- Lembre-se durante todo este curso de que programas devem ser feitos com paciência e devem ser testados a cada poucas linhas escritas de código
  - Se fizermos 200 linhas de código antes de testarmos o código, nossa probabilidade de erro é quase 100%

Este é o esqueleto básico de um programa em C++...

```cpp
#include <iostream>// biblioteca para leitura/escrita C++
using namespace std;
// a função main inicia a execução do programa
int main()
{
   return 0; // indica que o programa terminou com sucesso
} // fim da função main
```

Crie este programa no code::blocks.

Não se esqueça que ainda não devemos criar projetos e os arquivos devem ser salvos como .cpp

O programa *Olá, Mundo!* é o mais básico que fazemos para testar qualquer ambiente de programação

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Olá, Mundo!" << endl;

    return 0;
}
```

Crie este código, clique no botão do code::blocks para compilar e rodar o programa.

Teste agora este programa com capacidade de entrada do usuário...

```cpp
#include <iostream>
using namespace std;
int main()
{
    int num;
    cout << "Digite um número: ";
    cin >> num;
    cout << "O número é " << num << endl;
    return 0;
}
```

Faça um programa que contenha este trecho de código e teste o resultado...

```
...

int a = 1;

if(a == 0)
   cout << "a é igual a zero" << endl;

if(a < 0)
   cout << "a é menor que zero" << endl;

if(a > 0)
   cout << "a é maior que zero" << endl;

...
```

Programação de Computadores
Primeiros Passos em C++

Alan R R Freitas / Marco Antonio Carvalho