

cis1.5, spring 2009, CodeBlocks “how to”

1. Start up CodeBlocks

There are two ways to start up the **CodeBlocks** application:

(a) If you already have an CodeBlocks project file on your computer that looks like this:



hello.cbp

then double-click on it and CodeBlocks will start up.

(b) On the Desktop or in the Dock, find the CodeBlocks icon that looks like this:

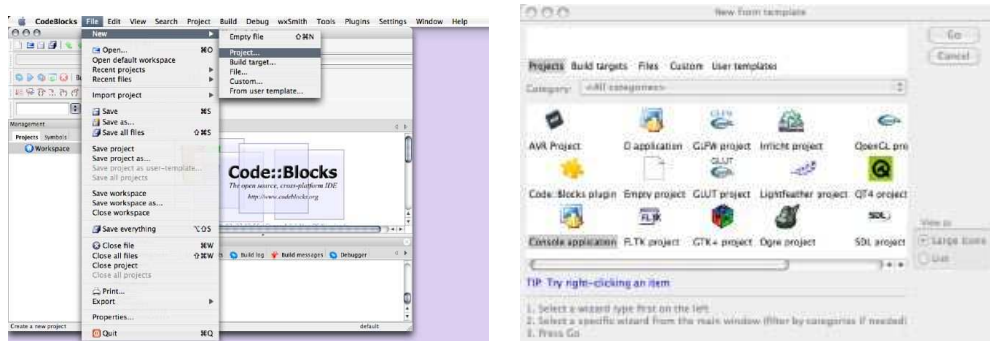


CodeBlocks

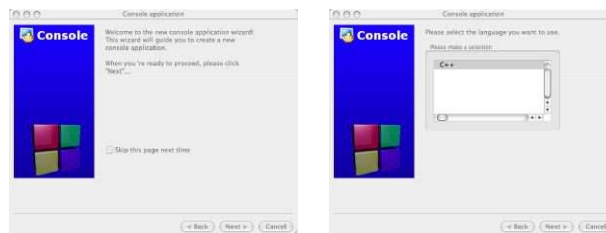
then double-click on it and CodeBlocks will start up.

2. Open a project

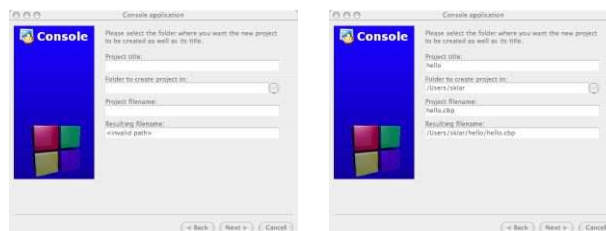
Click on **File – New Project** to create a new project. Then select **Console Application**, in the “New from template” window that pops up after selecting “New Project”. Then click on **Go**.



Then, in the **Console application** window that comes up, select the **C++** language and click on **Next>**. Note that C++ may be the only language from which to choose.



Then, enter the name of the project in the text field under **Project title**. Note that the **Project filename** and **Resulting filename** fields will fill in automatically when the project name is entered. If you want to change the folder where the project is located, then edit the **Folder to create project in** field or click on the small circle to the right of that field to browse the folders on your system and select one for the project.

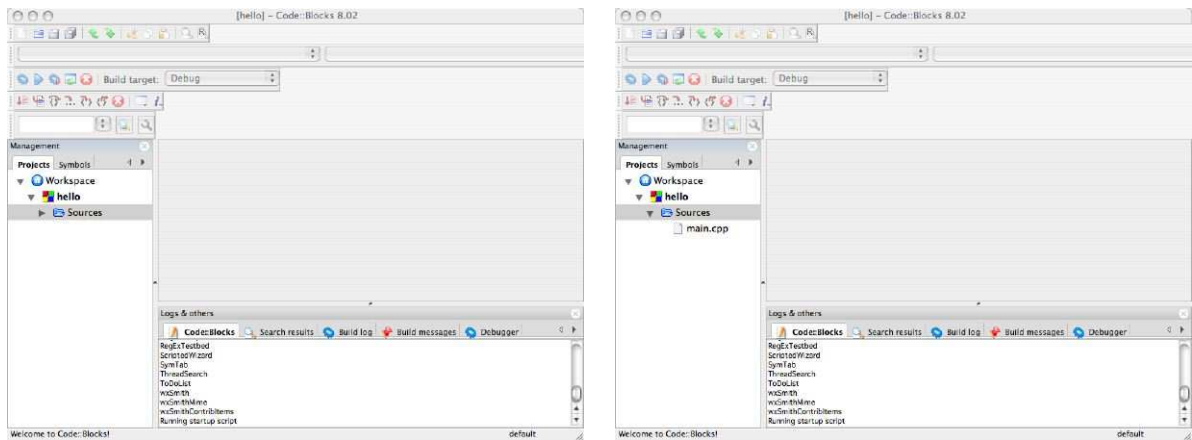


Finally, click on **Finish** when the window below appears. Note that you do not need to change anything in this window.

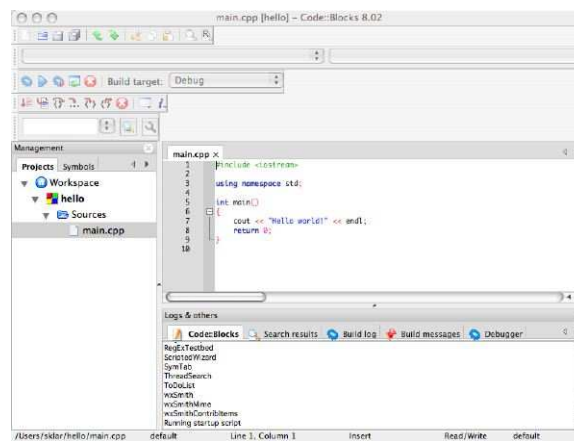


3. Edit the project

When the project window opens, clicking on the **Source** line in the Project Workspace (on the left side of the window) will list the source code files that belong to this project. The default template **main.cpp** will appear.

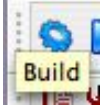


Double-click on the line that says **main.cpp** to open this file in the editing workspace.



4. Compile the project

Find the **Build** icon located in the upper left side of the CodeBlocks window:



and click on it to compile your code.

The **Build log** at the bottom of the editor window will display any status messages:



If there are errors compiling, then the lines where the errors occur will be listed in red in the “Build messages” window, along with corresponding error messages. Correct the errors and click on “Build” again.

If the build is successful, then go on to the next step.

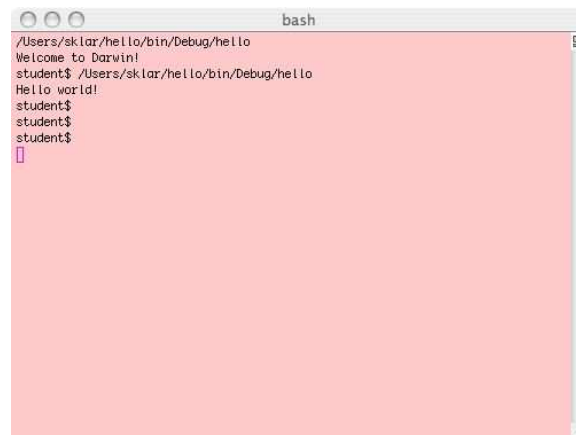
5. Run the project

Find the **Run** icon located in the editor window, next to the “Build” icon:



and click on it to execute your code.

On a Mac, the program will execute in a Terminal window. You will need to click out of CodeBlocks to the Terminal application and then you can see the program output:



In an Windows machine, the program will execute in a Console window that is opened within CodeBlocks.

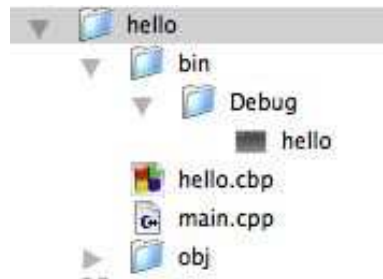
6. File structure

When you use an Integrated Development Environment (IDE) like CodeBlocks, it is important to understand the file structure of the project which is generated automatically by the IDE. As indicated earlier (see step 2), opening a new project creates a directory which, by default, has the same name as the project. In our example, this directory is called `/Users/sklar/hello/`. Inside this directory, there is a subdirectory called **bin/** and inside that directory, there is a file called **hello.cbp**, the main C++ source code file, which is called **main.cpp**, and a subdirectory called **Debug**. Inside the **Debug** directory, there is a file called **hello**. This file is the executable, or runnable file that is generated by the compiler. You could execute this file from the Terminal application command line, as we have done in class.

The overall project file structure is organized like this:

```
~/hello/
|
+----- bin
|
|   hello.cbp           <==== this is your CodeBlocks project file
|   main.cpp           <==== this is your C++ source code
|   +----- Debug
|       |
|       +----- hello   <==== this is your program executable
```

or as shown in the Mac Finder window:



When you save your work, you should save the entire `hello/` project folder.

When you submit assignments, you should ONLY submit the C++ source code, i.e., the **main.cpp** file.