How to handle multiple mains in your Program

Problem at Hand

Some apprenticeship exercises show projects in which multiple files each have their *main*. For instance, a project would have *tools.c* / *tools.h* files which implement a few key functions. These functions would then be used by a *main.c* file which provides a *main* function leveraging them to offer an interactive experience to the user. Another file, e.g. *tests.c*, would then provide another *main* used to run the same functions against tests defined elsewhere, e.g. in this file or in another pair of files such as *tests.c* / *tests.h*.

The end result is that if you compile all the files in your project, GCC will complain that you have several *main* functions thus making it impossible for it to determine where your program is expected to start running.

Linux Command Line Solution

As you've seen in the videos, the command line solution is simple. You need to make sure that you ask GCC to only compile a subset of files featuring a single *main*.

- If you want to use your *tools.c/.h* function in an interactive manner, you would compile them along with the *main* function from *main.c*. e.g.

  gcc -o myprog.exe main.c tools.c

- On the other hand, if you want these functions to be used by the *main* meant to run them through tests, you would use the one in *tests.c* e.g.

  gcc -o mytests.exe tests.c tools.c

Code::Block IDE Solution

There is a way to tell Code::Block that, for a give project with multiple files which might each have a main, you want to generate an executable based on a subset of these files. The idea is to modify the "build target". You see toward the top of your IDE a drop down box which generally shows the default build target "debug".

The following steps will show you how to add another build target to allow you to just build based on the files relevant to testing – e.g. tests / tools – then another build target allowing you to build the files relevant to running your programming interactively – e.g. main / tools

- open our project
- go to “project” → “Properties”
- go to tab “build targets”
- use button “Add”
- provide a name e.g. “testing”
- select the files which are part of this build in the “build target files:” area
    - For testing, you’ll pick the file with the *main* which runs all the tests & the files involved in implementing the functions you want to test e.g. *tools.c/.h*
    - You may do the same to prepare a new target for when you want to run your functions via the main allowing you to have an interactive menu
- use button “ok”
- You should now be able to use the dropdown menu “Build Target:” to select “testing” then build & run your project