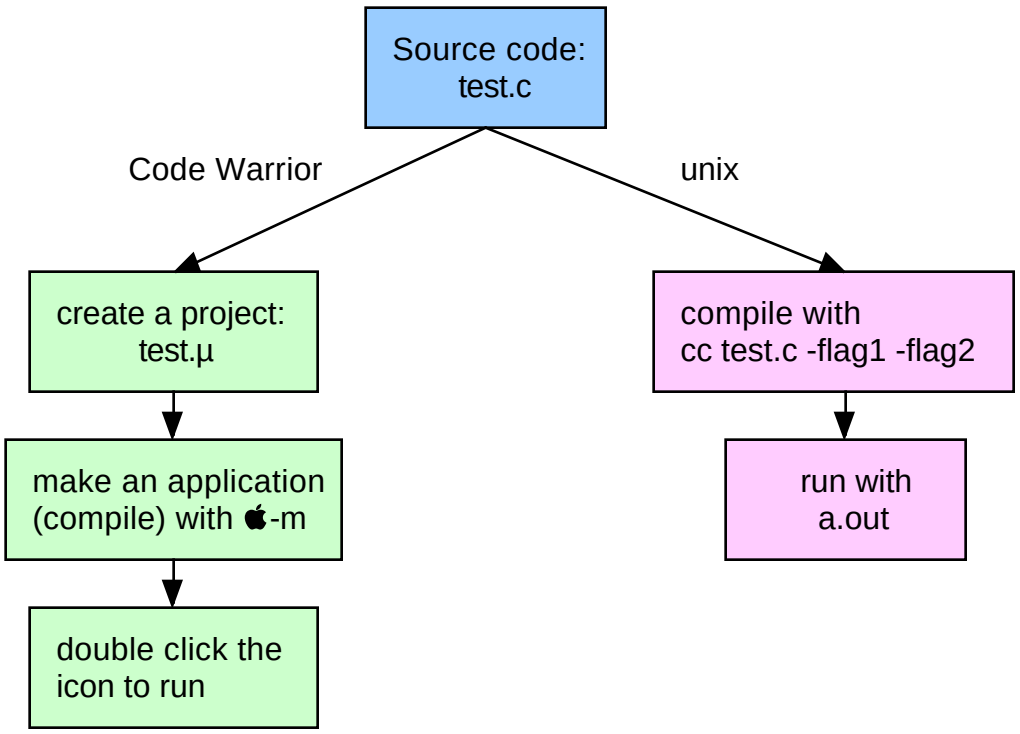
APPENDIX A - PROGRAMMING ENVIRONMENT

In a command-line oriented operating system such as **unix**, the compilation command for a code might run something like

cc code.c -flag1 -flag2 -flag3 ....

where the flags might include links to libraries, optimization levels *etc*. Keeping track of flags and other settings is a minor bother, unless MakeFiles are used.

Code Warrior, which is the C/C++ compiler used in this course, provides an integrated environment for keeping track of flags, for linking to reference sources, for editing files *etc*. Compared to unix, **Code Warrior** requires one extra procedure, namely the creation of a project. However, once the project is created, the tasks of editing, linking, *etc*. become much simpler, especially if one is constantly renewing a code.

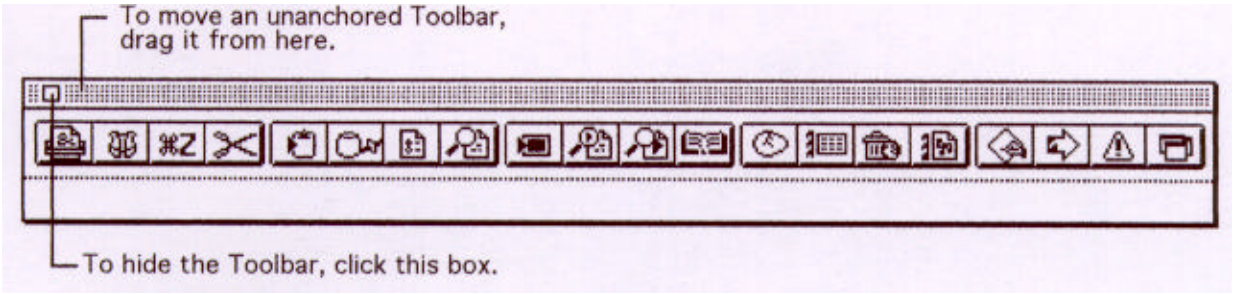


What does the **Code Warrior** environment look like? First, open the **CodeWarrior** application, whose icon looks like (for **CW9**)

Once the application is open, there is a menu with titles

File Edit Search Project Tools Window

as well as a toolbar:



The set-up is much like word-processing software, in which you can use either the menu or the toolbar for commands, such as
•set the **preferences** for debugging, optimization, even specifying the processor
•provide links to libraries and other resources, such as custom icons
•provide links to reference resources.

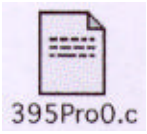
Your first project

The easiest way to explain the development environment of **Code Warrior** is to work through an example. In this project, we will simply write the words "Welcome to Computational Physics" on the screen. Writing a line of text to the screen is a straightforward task. The following piece of C code will do the trick:

```
#include <stdio.h>
void main(void)
{
  printf("Welcome to Computational Physics");
}
```

Our first step is to write the five lines of code above using a simple text editor. You can use **vi** on the SGIs, or click on an icon like

to open a text editor on the Macs. Once you have entered the 5 lines of code, save your text file as 395Pro0.c (or whatever you want to call it), and close the file. You now have a text file containing the code:

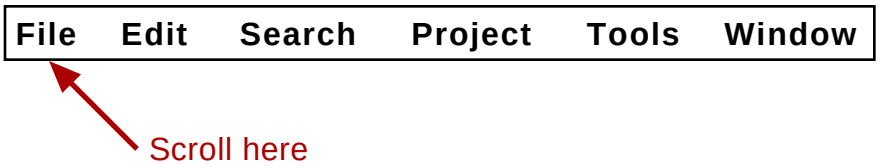


If you are using **unix**, then you can compile and run the code as it is.

In a "console" window, or dumb terminal, this code will just write the string of text at the current prompt. However, in a graphical environment like the Mac OS, one doesn't see lines of text written at arbitrary places on the screen. The text and other output may appear in windows. One aspect of a **Code Warrior** project deals with the opening and closing of windows.

With **Code Warrior**, you now create a project. First, open **Code Warrior** with the same icon as you used before:

Go to the menu at the top of the screen, and scroll down from **File** to open a **New Project** (or click the **New Project** button in the toolbar):



This action generates a box, allowing you to **name** the project, and select the **project stationary**. What does this mean? First, under

Name the project as:

you name the file where you will keep all of the project info, and through which you will edit, compile, etc, your source code. All projects must end in **.µ** so name the test project **395Pro0.µ** after the source code **395Pro0.c** (you can type a **µ** through the **alt-m** combination of keys):

Name the project as:

395Pro0.µ

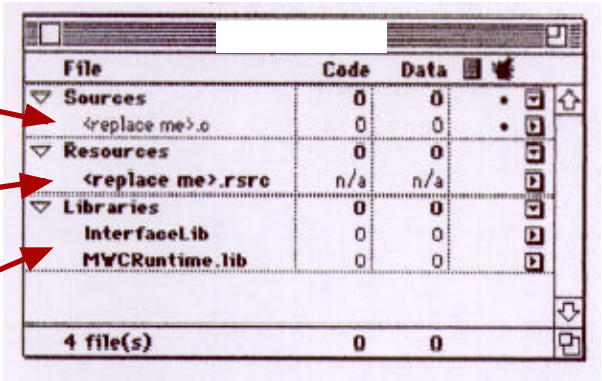
You also have a choice of **project stationary**, which includes a set of libraries that will be linked to the project. Choose **MacOS PPC** which contains everything we need. Don't choose the empty folder, since it has no links to libraries. Click on **Save** to save the project (make sure that you are happy with the open folder to which the file is saved).

The dialog box now closes, and the **project window** opens. An abreviated form of the window looks like (in Code Warrior 9, somewhat different in Code Warrior 11):

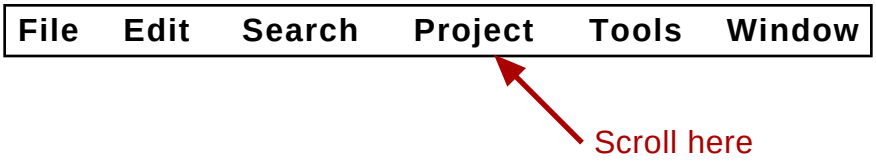
Don't edit this, replace it with your source code

Don't edit this, replace it with your resources

These are your libraries, including windows



At this point, the project has no information about your source code (*e.g.* **395Pro0.c**) or resource files (e.g., custom icons). The **<replace me>** files are just space holders - don't edit them, just remove them AFTER you have added the files you want. To add files, scroll down from the **Project** menu at the top of the screen:



When you open **Add Files**, you have two boxes with (perhaps!) some file names in them. The top box lists the files available in the folder that you have selected. Click on the ones you want to add to the project, then click on the **Add** button. Similarly, select from the bottom box those files that you want to remove from the project, then click on the **Remove** button. When you have things set up the way you want, click **Done**.

After removal of the **<replace me>** files, your project window should have the following files:

Sources
- **395Pro0.c**

Resources
- **whatever.rsrc**

Mac Libraries
- **InterfaceLib**
- **MathLib**
- **MWCRuntime.Lib**

and perhaps the following, depending upon the version of Code Warrior that's mounted on your machine:

ANSI Libraries
- **ANSI C++.PPC.Lib**
- **ANSI C.PPC.Lib**
- **SIOUX.PPC.Lib**

Most of these items are self-explanatory. We will comment briefly on resource files a little later. The **SIOUX.PPC.LIB** sets up the window onto which your code will write its message.

Whew! It seems like a lot of work to set up the project. But now life is simpler:
To edit the source code, just click on it.
To compile and link the source code:
- •select **Make** from the **Project** menu, or
- •type ***command* - m**, or
- •click on the **Make** button in the toolbar.

To run the application that you have made:

- •select **Run** from the **Project** menu, or
- •type ***command* - r**, or
- •click on the **Run** button in the toolbar.

Once you have the project set up, you can quit at any time, and there will be an icon in your folder corresponding to **395Pro0.**μ:

Click on this icon when you want to open the project.

Once you have successfully compiled the source code, there will be an icon in your folder corresponding to the application. If you are working in the Computational Physics Lab, the icon looks like

However, if you are working elsewhere, a generic icon may appear instead. Click on this icon when you want to run the application. The name on the icon is supplied by the **Preferences** in the **Edit** menu of the project. The icon itself it supplied by the resource files.

Now you can run the application. If the code is correctly written, a window will open and your message will appear in it. Once you're tired of looking at it, close the window with ***command* - q**, as usual.

Resource files

Generating resource files for the Mac - windows, dialog boxes, icons - is beyond what we can handle in this course. Interested students should get themselves a copy of the software **ResEdit** and read *Zen and the Art of Resource Editing* by D. Schneider and H. Hansen (Hayden Books, 1995) 4th edition. A **ResEdit** icon looks like:

Code Warrior Help

For help with C/C++, QuickDraw etc., scroll down the **Search** menu or click on the open book in the toolbar.

General references

M. P. Allen and D. J. Tildesley, *Computer Simulations of Liquids* (Oxford, 1987).

K. Binder and D. W. Heermann, *Monte Carlo Simulation in Statistical Physics* (Springer-Verlag, 1988).

D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning* (Addison-Wesley, Reading, Mass., 1989).

H. Gould and J. Tobochnik, *An Introduction to Computer Simulation Methods* (Addison-Wesley, Reading, Mass.,1988).

S. Koonin and D. Meredith, *Computational Physics* (Addison-Wesley, Reading, Mass., 1990).

B. Müller and J. Reinhardt, *Neural Networks* (Springer-Verlag, Berlin, 1990).

W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery *Numerical Recipes in C*, 2nd edition (Cambridge, 1992) .

D. C. Rapaport, *The Art of Molecular Dynamics Simulation* (Cambridge, 1995).