

UNVIERSITY OF VICTORIA

Manual Eclipse CDT Mac OS Leopard

Installation & Demonstration Guide

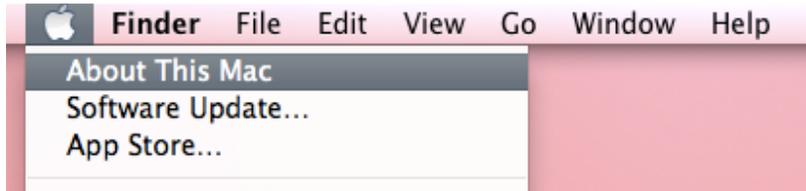
Przemek Lach

9/3/2013

This guide shows how to use install Eclipse and C-Compiler and how to test the setup by creating a simple 'hello world' program. NOTE: This guide is based on Mac OS Leopard (64Bit) and Eclipse Kepler (4.2).

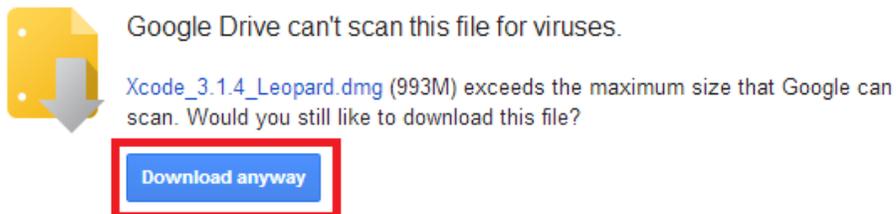
Mac OS Version

This tutorial is designed to work for Mac OS Leopard. To find out what version you have open 'About This Mac' and look at the version number under the Apple sign. If the version numbers starts with **10.5** then you've come to the right place. Note the version number in the image below is not correct and is only here for demonstrative purposes.



Installing the Compiler

1. The C-Compiler for Snow Leopard comes with Command Line Tools. Command Lines Tools come with Xcode. Xcode is an IDE for Mac OS and must be downloaded and installed in order to get the Command Line Tools. Note this is a large down. To download Xcode click on the following link:
<https://drive.google.com/a/rigiresearch.com/uc?id=0B7bIXSWNFM5RREhndVhIYUtGcEU&export=download>
2. On the page click Download button.

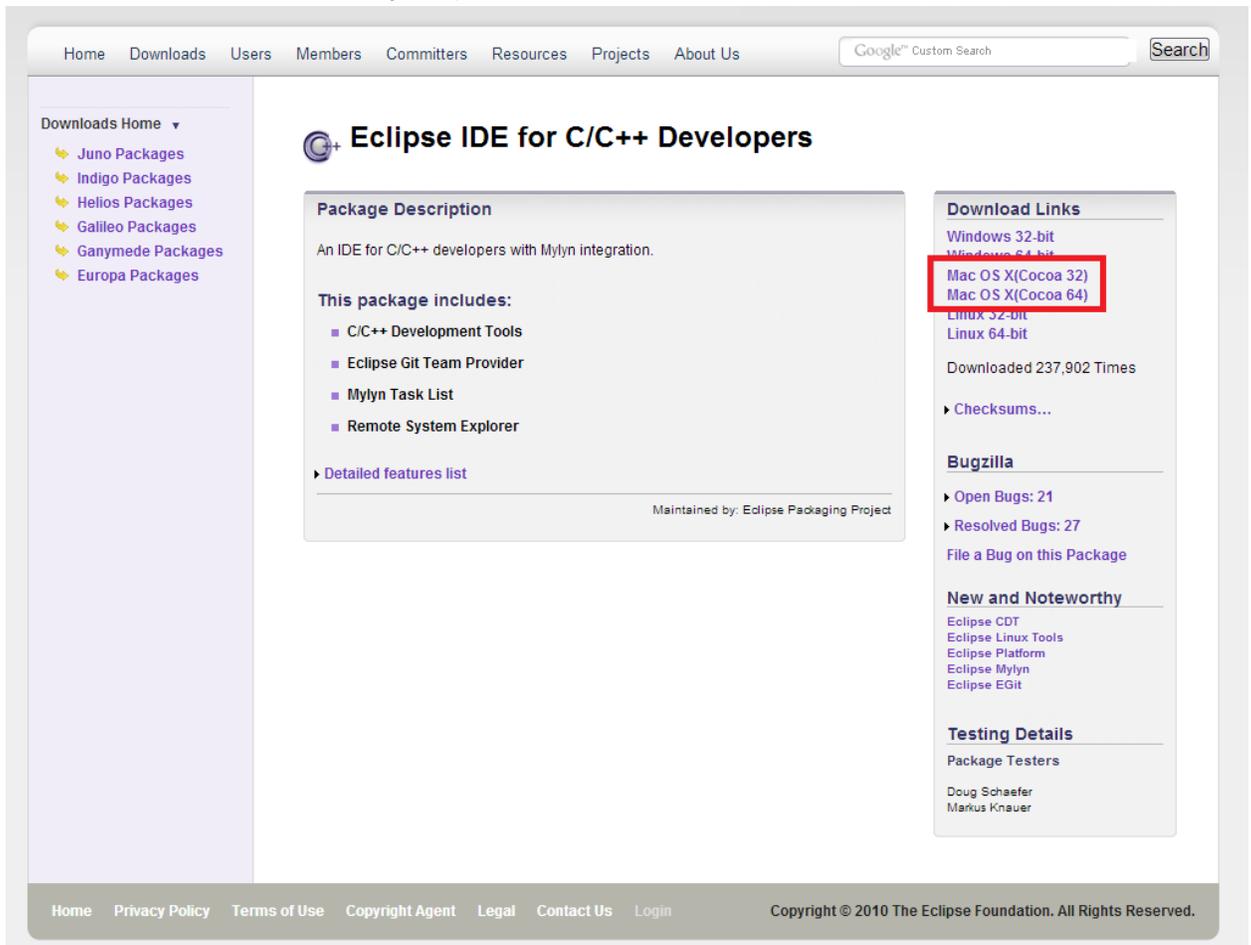


3. Once the file is downloaded double click on it and following the instructions provided by the installer.

4. Once Xcode installation is complete you have installed Command Line Tools and you are done.

Installing Eclipse

1. Ensure that you have the Java Runtime Environment (JRE) installed. This is required for Eclipse to run. If you plan on using a 64Bit version of Eclipse then you need a 64Bit JRE. If you plan on using a 32Bit version of Eclipse then you need a 32Bit JRE. In this demonstration we are using the 64Bit version.
2. Navigate to:
<http://www.eclipse.org/downloads/packages/eclipse-ide-cc-developers/keplerr>
and download the version for your platform.

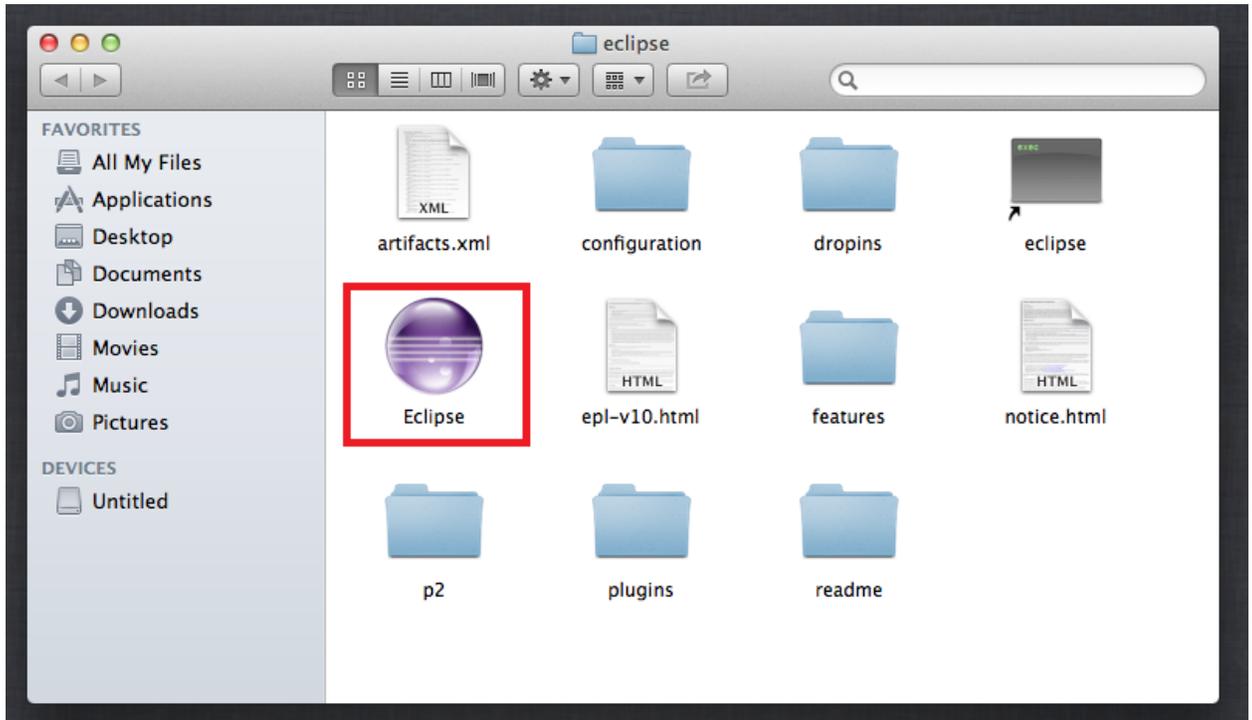


The screenshot shows the Eclipse IDE for C/C++ Developers download page. The page has a navigation bar at the top with links for Home, Downloads, Users, Members, Committers, Resources, Projects, and About Us. A search bar is also present. On the left, there is a sidebar with a 'Downloads Home' dropdown menu containing links for Juno, Indigo, Helios, Galileo, Ganymede, and Europa packages. The main content area is titled 'Eclipse IDE for C/C++ Developers' and includes a 'Package Description' section stating it is an IDE for C/C++ developers with Mylyn integration. Below this, it lists features included in the package: C/C++ Development Tools, Eclipse Git Team Provider, Mylyn Task List, and Remote System Explorer. A 'Download Links' section on the right lists various operating systems and architectures, with 'Mac OS X(Cocoa 32)' and 'Mac OS X(Cocoa 64)' highlighted in a red box. Other sections include 'Bugzilla' with open and resolved bug counts, 'New and Noteworthy' with links to related projects, and 'Testing Details' with package testers listed.

3. Extract the downloaded file where it is most convenient. There is no installer for Eclipse so wherever you unzip is where Eclipse will exist.

Demo Using Eclipse

1. Open the *eclipse* folder by double clicking on eclipse folder (where eclipse was downloaded). The contents of this folder will look like:



2. At this point you may get a warning that Eclipse can't be opened because it is from an unidentified developer.

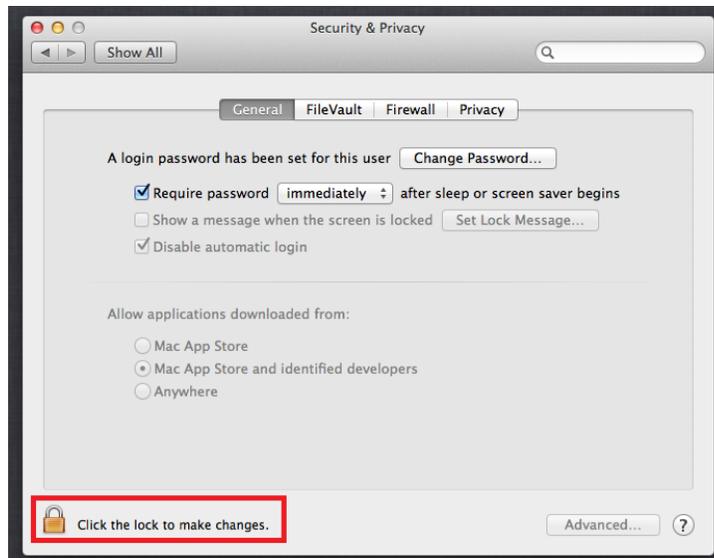


If this is the case then you have to allow the installation of third party applications in Mac OS.

- a. Open System Preferences and click on Security & Privacy



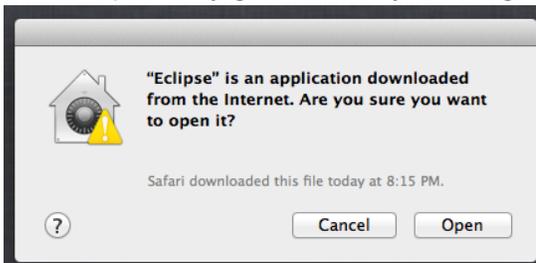
- b. On the next window ensure that editing is enabled. On the bottom left see that the padlock is open. If it is not click on the padlock to enable the editing of settings on this window.



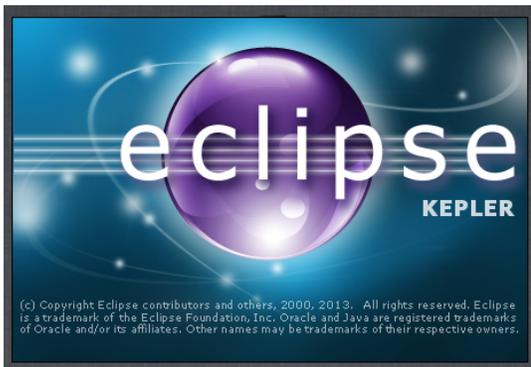
- c. Now under the "Allow applications download from:" section change the selection to "Anywhere". At this point you will get a second confirmation drop down. Click "Allow From Anywhere". Now you are done. Go back and double click on the Eclipse icon again.



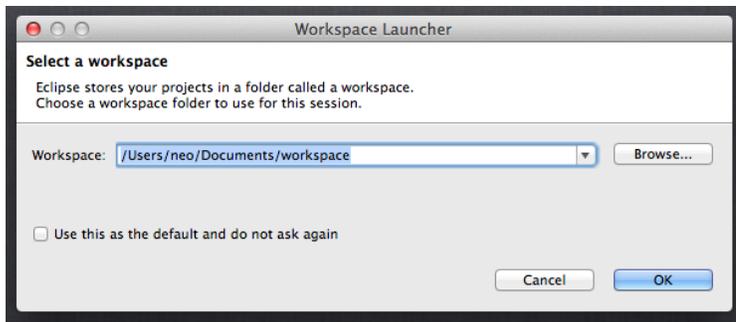
3. You will probably get a security warning about starting Eclipse. Click "Open".



4. After you double-click on Eclipse, the following screen will appear indicating that eclipse has started.



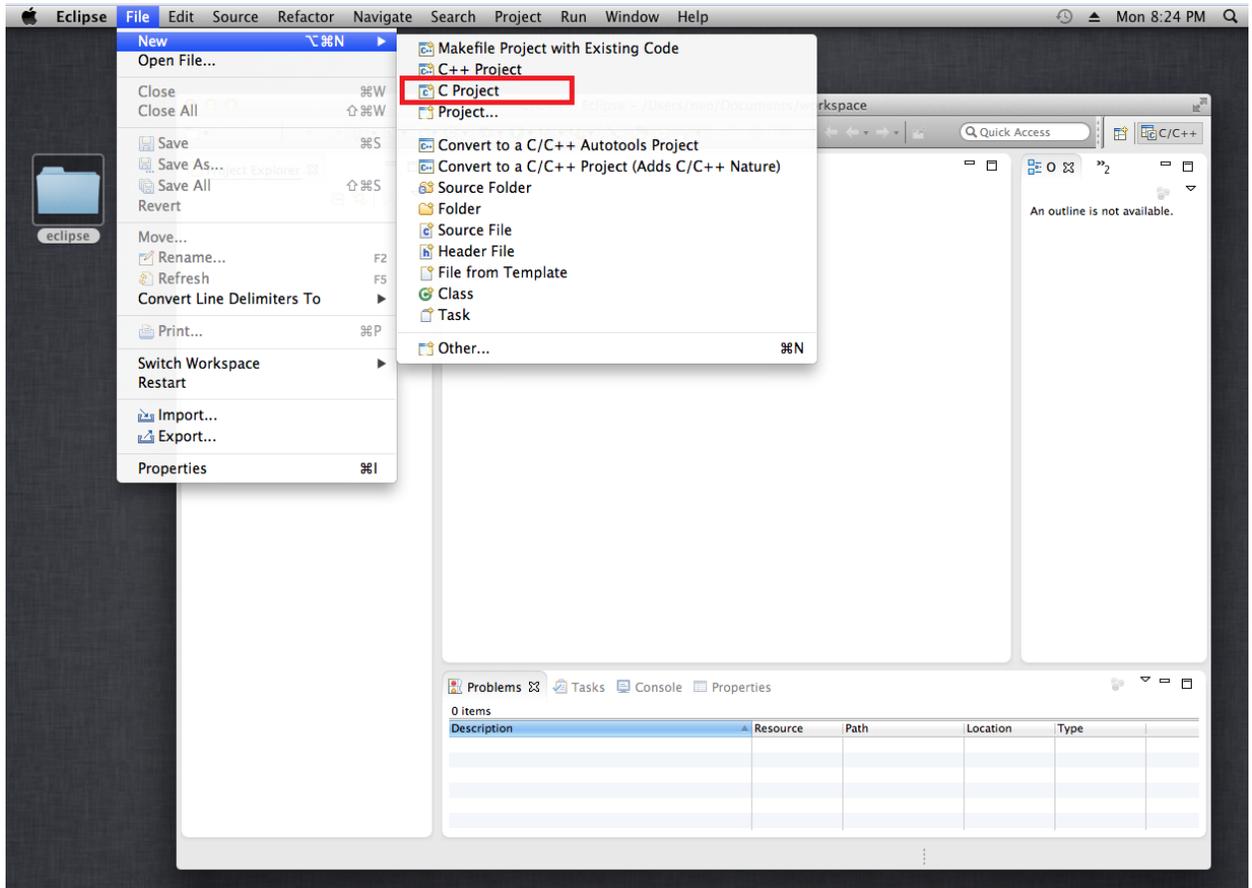
5. Immediately after that Eclipse will ask you for your workspace location. It defaults to a place within your personal settings. It is up to you to choose a new location or use the default workspace. Then, click OK.



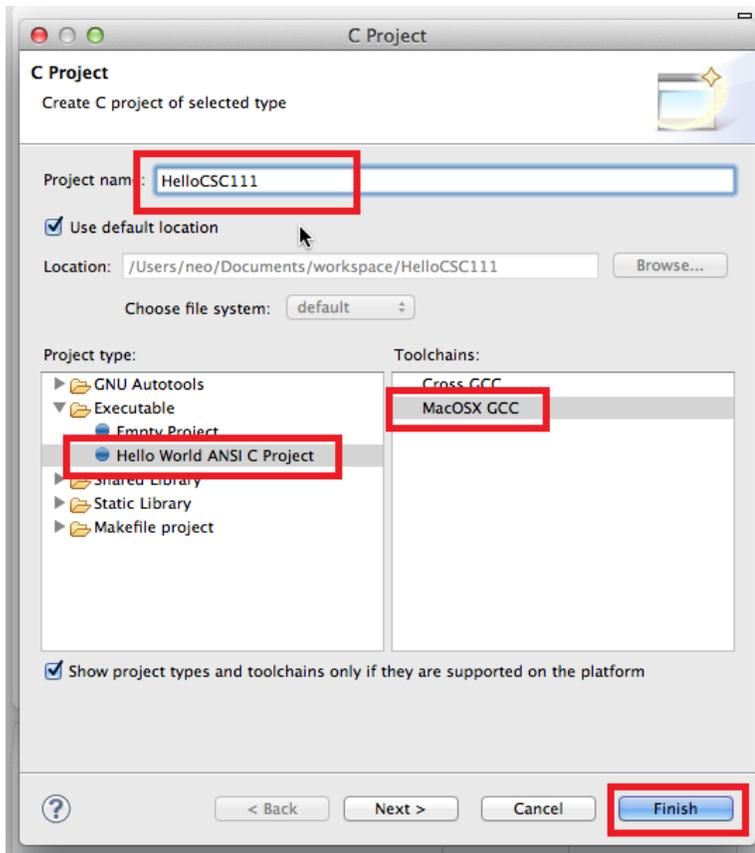
6. Eclipse starts up with a welcome page. Click on the "x" to close the welcome page.



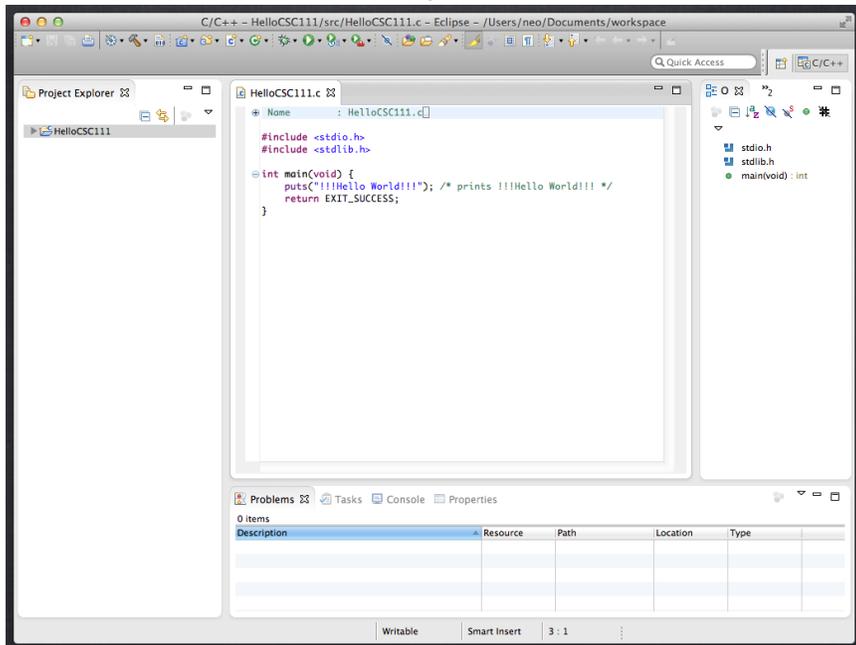
7. Create a new C project via the File menu.



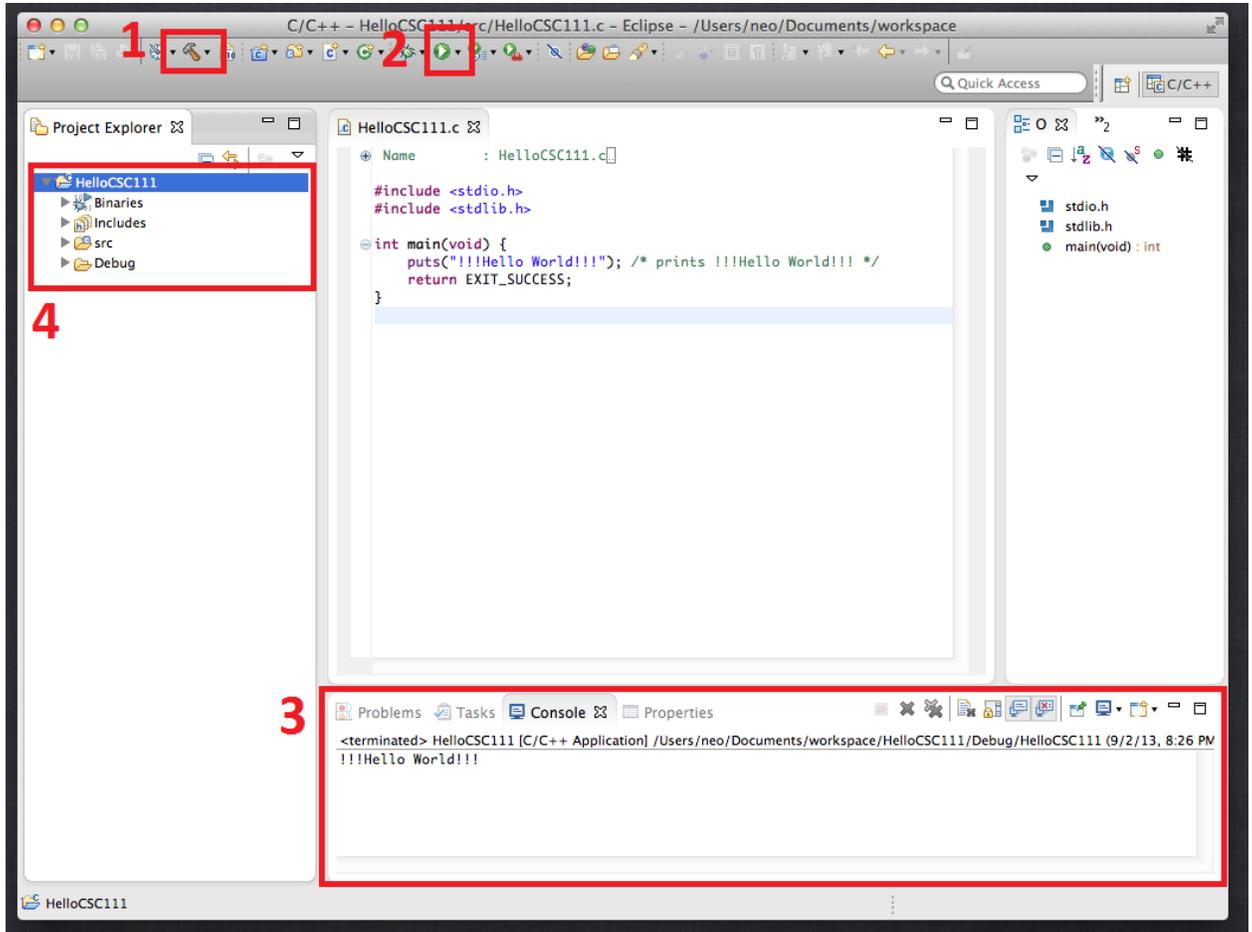
8. Give your project a name; ensure there are no spaces in your project name. Select the Executable and Tool Chain options as highlighted below. This is very important. You must select the right Tool chain or things will not work. Click "Finish".



9. You should now see the following view.

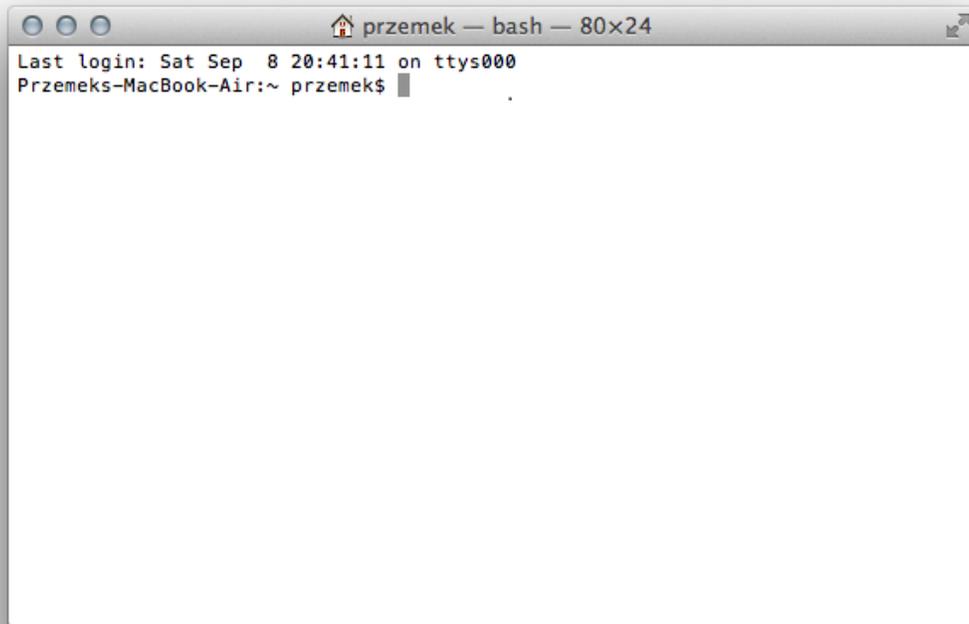


10. Now build and run your project. Expand the tree on the left hand side if you want to see your files. On the bottom of the screen select the "Console" tab where you will see the output of your program.



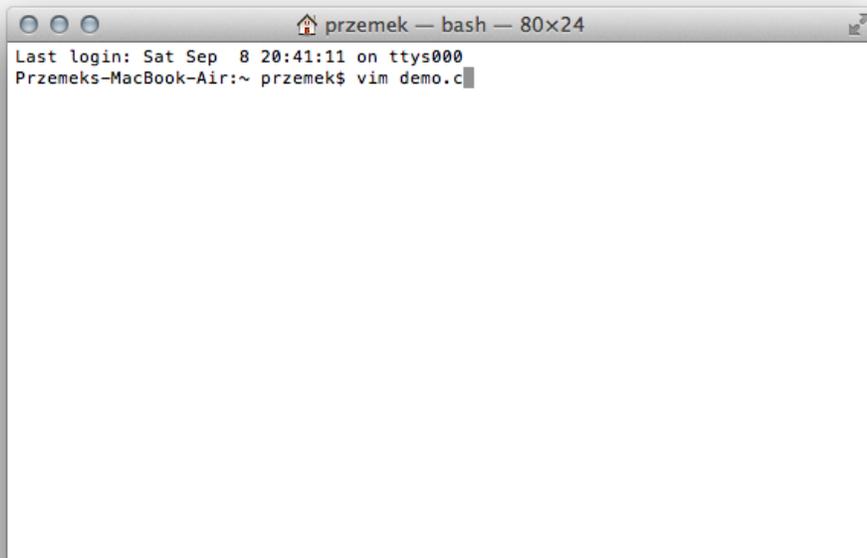
Demo Using Command Line

1. Start the Terminal



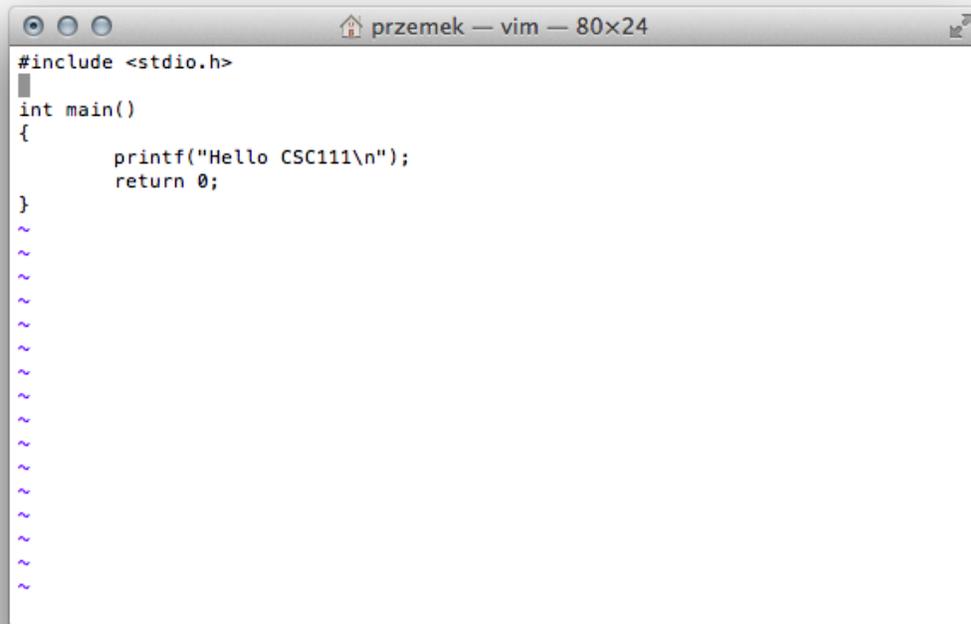
```
przemek — bash — 80x24
Last login: Sat Sep  8 20:41:11 on ttys000
Przemeks-MacBook-Air:~ przemeks$
```

2. We will use the vim text editor to create our hello world. Simply type 'vim demo.c' and press 'Enter'.



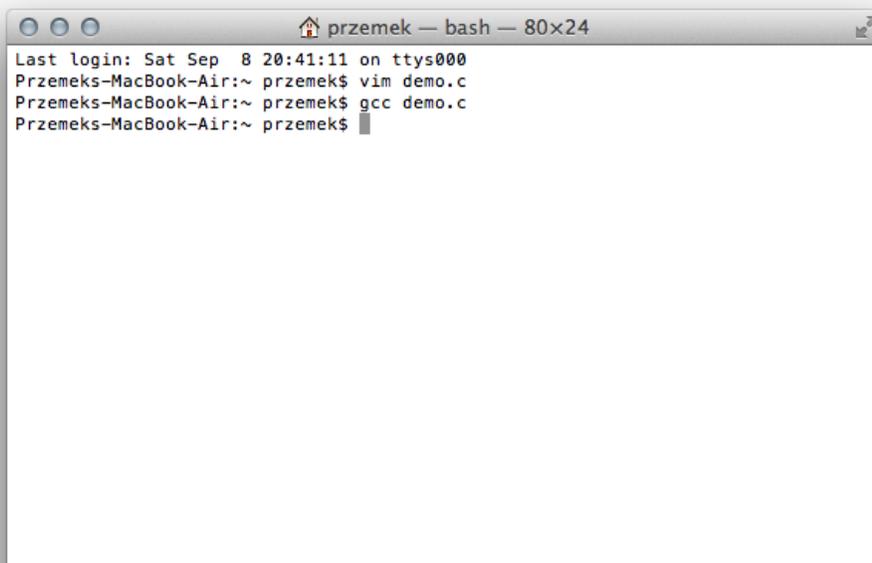
```
przemek — bash — 80x24
Last login: Sat Sep  8 20:41:11 on ttys000
Przemeks-MacBook-Air:~ przemeks$ vim demo.c
```

3. Press the 'i' character to turn on edit mode. Type the program as it appears below



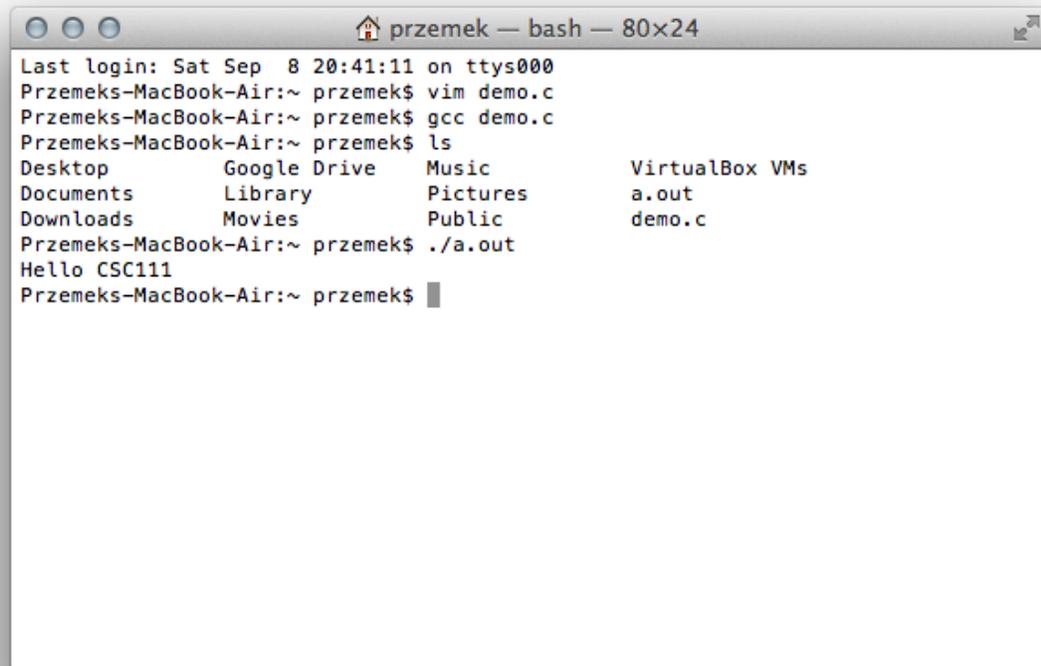
```
przemek — vim — 80x24
#include <stdio.h>
int main()
{
    printf("Hello CSC111\n");
    return 0;
}
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
```

4. Press 'Shift+:', type 'wq' and press 'Enter'. This will save your work and exit the vim program. You should now be back in Terminal. Compile the program by typing 'gcc demo.c'. If you entered your program correctly you should get no errors.



```
przemek — bash — 80x24
Last login: Sat Sep  8 20:41:11 on ttys000
Przemeks-MacBook-Air:~ przemek$ vim demo.c
Przemeks-MacBook-Air:~ przemek$ gcc demo.c
Przemeks-MacBook-Air:~ przemek$
```

5. Now you can type the 'ls' command which will show all the files in the current directory. If your program compiled successfully you will see a 'a.out' file. This is your compiled program. Type './a.out' to run the program. You should see your message.



```
przemek — bash — 80x24
Last login: Sat Sep  8 20:41:11 on ttys000
Przemeks-MacBook-Air:~ przemek$ vim demo.c
Przemeks-MacBook-Air:~ przemek$ gcc demo.c
Przemeks-MacBook-Air:~ przemek$ ls
Desktop          Google Drive    Music           VirtualBox VMs
Documents        Library         Pictures        a.out
Downloads        Movies          Public          demo.c
Przemeks-MacBook-Air:~ przemek$ ./a.out
Hello CSC111
Przemeks-MacBook-Air:~ przemek$
```