Lesson 1: Intro to C++ and Eclipse

3D Game Programming With C++

Digital Media Academy (Summer 2011)

Written by: Andrew Uzilov (andrew.uzilov@gmail.com)
Feel free to contact me with any questions.

Welcome to the awesome world of C++ programming!!

Why learn C++ ?

- Produces blazing fast programs like no other high-level programming language can!
- Industry standard in gaming or pretty much any software where performance matters.
- Tons of libraries (useful code written by other people) – for example, Panda3D!
- Feature-rich, powerful language; if you know it, other programming languages are much easier to learn.

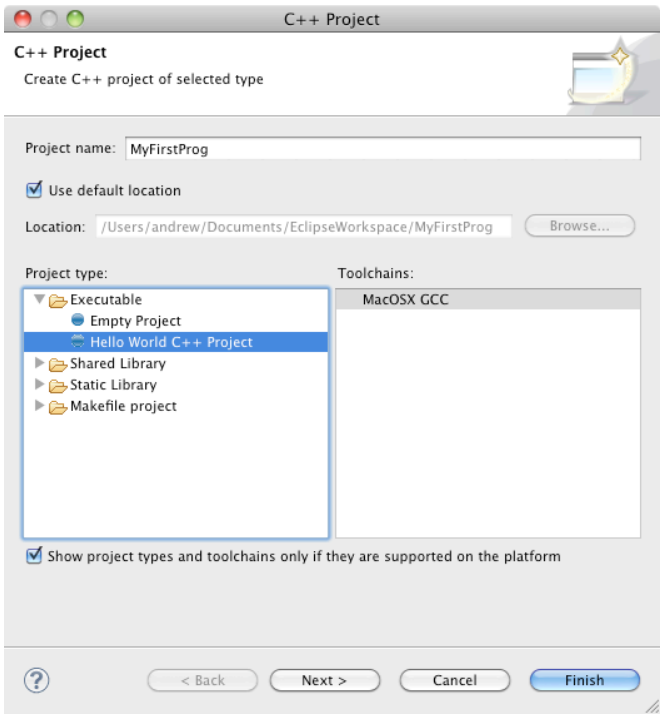
THE BASICS

- High-level programming languages – they tell another person what you want the computer to do.
- C++ → compiler → machine code (OS- and hardware-dependent). Machine code is stored in an executable binary file.

INTRO TO ECLIPSE

You will be editing your code in the **Eclipse Integrated Development Environment (IDE)**. Let's fire up Eclipse and create our first program, which will be automatically generated from an Eclipse template.

In Eclipse, go through the menus File → New → C++ Project and you will get a window like this:



Fill out the "Project name" field with `MyFirstProg` and make sure you choose "Hello World C++ Project", then click "Finish".

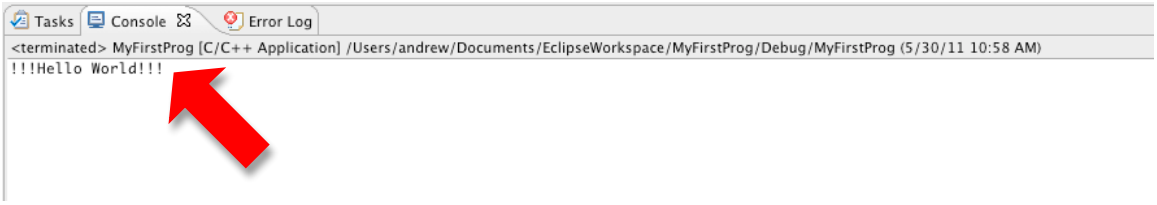
Your first program will appear in the editor window in the middle; it will look like this:

```
//============================================================================
// Name        : MyFirstProg.cpp
// Author      : Andrew
// Version     :
// Copyright   : Your copyright notice
// Description : Hello World in C++, Ansi-style
//============================================================================

#include <iostream>
using namespace std;

int main() {
	cout << "!!!Hello World!!!" << endl; // prints !!!Hello World!!!
	return 0;
}
```

Compile and run the program by clicking the Run button ( ) at the top of the Eclipse workspace. The program output will be text, and it will appear in the **Console** tab at the bottom. It should look like this:



Let's go over the program and learn what each part of it does! The exact details right now are not important; we'll go over them more thoroughly in later lessons. Here are the key points (terms in **bold** are the most important):

- The `#include` line means our code will make use of a separate **library** of code called `iostream`. It contains stuff we need to print text to the **console**. `#include` literally means: "copy everything from that file/library and paste it into this place in my file."

- `main()` is a **function**, which is a block of code that has a name.

- `cout` is a special object called a **stream**. It comes from the `iostream` library. We use the << syntax to send **strings** (letters in double quotes) to the stream. Everything that goes to `cout` comes out on the console.

- `endl` causes the line that is being created by `cout` and << to end (it stands for "end line") and appear in the console. It should appear at the end of every line you want to print.

Comments are parts of the program that aren't compiled. They have no effect on your program. They are just there to tell you stuff about the program in an easy-to-read way. There are two kinds of comments:

```
/* These kinds of comments can go across multiple lines.
   They begin with a front slash and a star, and end with
   a star and front slash. */

// These kinds of comments are one line only.
// They begin with two forward slashes and end
// when the line ends.
```

Programs in C++ can be broken down into statements. A statement is like a sentence in the English language. Here is an example:

```
int damage = modifier * 35;  // what do you think this does?
```

Just like sentences in English usually end with a period, statements in C++ usually end with a semicolon.

Exercise 1.1: Your first C++ program.

Modify `MyFirstProg` so that instead of "!!!Hello World!!!" it prints your name and where you're from to the console, all on one line.

Exercise 1.2: Printing multiple lines.

Start a new "Hello World" C++ project and call it this: `Exercise_1_2`

In that project, write a program that prints out exactly four lines to the console, like this:

```
Welcome to the wonderful world of C++!!
Prepare to be challenged and amazed.

Are you READY for it?
```

Save the program, run it, and show it to your instructor before moving on to the next lesson.