**Wave Robotics Software Fall Training Syllabus**

[www.beginnerscpp.com](http://www.beginnerscpp.com), Tutorials are “talk & type”, can skip compiler related ones.

1. Introduction to programming (IDEOne)
	1. Open The Door demonstration
	2. ‘Hello World’ (**http://ideone.com/YpnMlx**)
	3. How to read compiler errors
	4. Print lots of strings with ‘endl’
2. Variables, If/Else (IDEOne)
	1. Like ‘x’ and ‘y’ in math
	2. int and double (**http://ideone.com/qhuMQK**)
	3. Direct assignment
	4. If/else (**http://ideone.com/UWLyEs**)
	5. Keyboard shortcuts: copy, cut, paste, shift arrow, etc
3. Variables, Simple Math, Into to Functions (IDEOne)
	1. Add, subtract, multiply, divide
	2. Comparisons: greater than, less then, equal to ([**http://ideone.com/QKVgee**](http://ideone.com/QKVgee))
	3. Exercise
		1. Print the first 15 numbers in the Fibonacci sequence, starting with 0 and 1, and do it using variables and math (0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377)
		2. Next, add code that will print one word (like "even") if the Fibonacci number is even, and another word if it is odd.

Remember that a number is even if it can be divided by 2 with no remainders, use "%" to give you the number of remainders

For example: "8 % 2" results in 0, so 10 is even. But "13 % 2" results in 1, because 2 goes into 13 6 times with 1 left over

* 1. What is a function? How does the compiler know a function exists (prototypes)?
		1. Code that needs to be called a lot, or code that the main program really doesn’t care what gets done.
	2. Simple functions, no parameters and no return values (**http://ideone.com/IRkPpW**)
1. Functions with Parameters and Return Values (IDEOne)
	1. What is a parameter? ‘x’ in y=mx+b (**http://ideone.com/W0BPnN**)
	2. What is a return value? ‘y’ in y=mx+b
	3. Sensor voltage conversion example (**http://ideone.com/CT02O3**)
2. Intro to Classes (IDEOne)
	1. It’s like a super variable. It can contain variables and functions.
	2. All of the information inside a classes is related.
	3. Example: Student information. First name, last name, Grade, Subject taking, grade for each subject, GPA
3. More Classes (IDEOne)
	1. Analog Sensor Class: IO Point, GetRawValue(), GetConvertedAngle()
4. Class Interactions (IDEOne)
	1. Analog Sensor, Motor output.
	2. Turn motor on until angle is reached, then turn motor off.
5. Class Interaction continued (IDEOne) / Review Week
	1. More work on Sensor/Motor control as needed
	2. Work on any aspects still needing review
6. RobotBuilder (Eclipse)
	1. Intro to the Eclipse Environment
	2. Build a robot using RobotBuilder
7. Robot Control (Eclipse)
	1. Run an arm with a joystick
	2. Drive with the joystick
	3. Turn a roller on and off with a button
8. Robot Control continued (Eclipse)
	1. Similar to previous week
	2. Virtual stops on the arm
9. Robot Commands (Eclipse)
	1. Take the arm control and turn it into a command that is activated by a button
	2. Set different angles with different buttons.