

**School of Computer Science**

**Faculty of Science**

**COMP-2650: Computer Architecture I: Digital Design**

**Fall 2020**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lab# | Date | Title | Due Date | Grade Release Date |
| Lab 01 | Sept 21-23, 2020 | **L01: Programming Environment Setup** | Sept. 30, 2020  Wednesday Midnight [AoE](https://www.timeanddate.com/time/zones/aoe) | Oct. 07, 2020 |

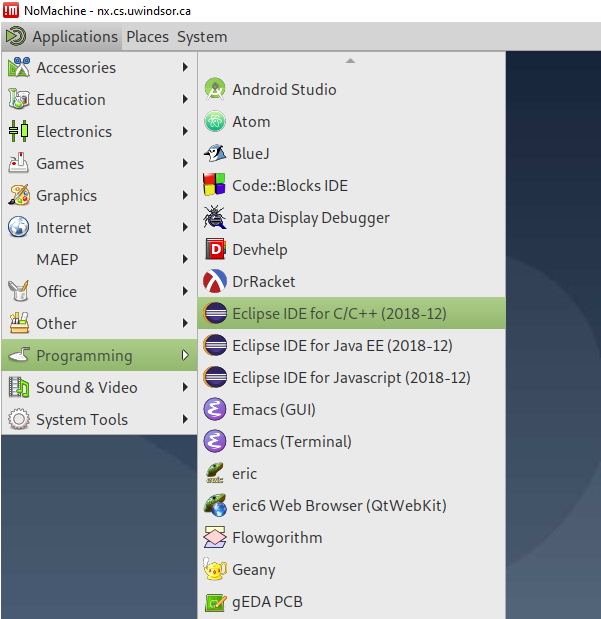
The objectives of the first lab will be for you to set up a programming environment, specifically to have an integrated development environment (IDE) installed and functioning on your local drive in the lab or on your own desktop or laptop computer. Also, you will be developing on the program specification of the lab assignments that you will be gradually completing throughout this term.

**Step 1. Environment Setup**

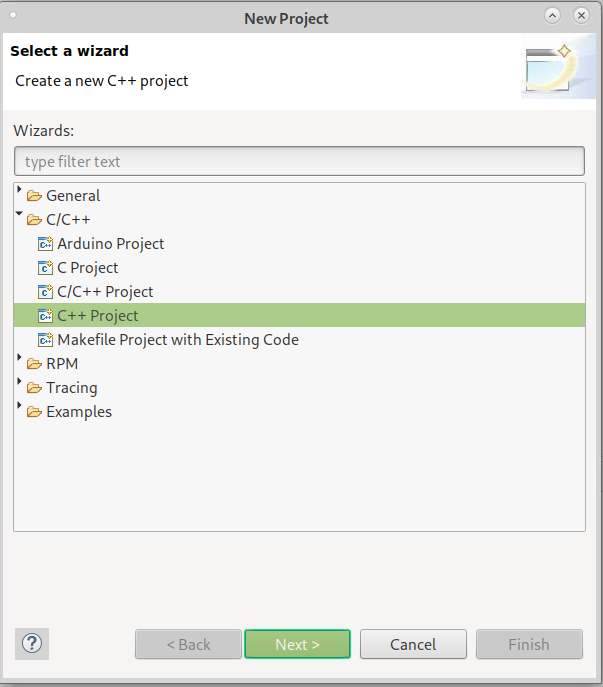
In [computer engineering](https://en.wikipedia.org/wiki/Computer_engineering), there are hardware description languages (HDL) such as VHDL and Verilog that are specialized [computer language](https://en.wikipedia.org/wiki/Computer_language) used to describe the structure and behavior of [electronic circuits](https://en.wikipedia.org/wiki/Electronic_circuit), and most commonly, [digital logic](https://en.wikipedia.org/wiki/Digital_logic) circuits. However, in computer science and this course mainly, we rely on schematic-based design methods that create a representation of functionality. We literally use pen and paper to draw circuits and gates. Therefore, we do not cover any HDL herein. We instead simulate the circuit's functionality via writing programs using a general-purpose programing language to make sure that our designed circuit is working correctly.

We will use C/C++ as the programming language. C++ is the extension to the C and has more features. We are not using those features in this course and, therefore, C++ is similar to C for us. Although we create C++ projects in Eclipse, we only use C language. This way you will see some limitations of C language and will be motivated to move on to C++ language in future.

For IDE, we use Eclipse as the programming environment throughout the lab manual. You are however free to adopt any programing environment such as NetBeans or text editors like nano or vim. Eclipse is free and open-source. Go to the [download page](https://www.eclipse.org/downloads/) and grab the binary for your platform. When installing Eclipse, remember to choose Eclipse IDE for C/C++ Developers. You can also connect to computer systems available in the school to work remotely using this [link](https://help.cs.uwindsor.ca/en/servers/remote-access/no-machine/). Therein, Eclipse have already been installed and ready to use. Make sure you can start up a Hello World project in C++ and run it in Eclipse. Here's a quick test for you to try and create a new C++ project using remote computers in the school using Eclipse:

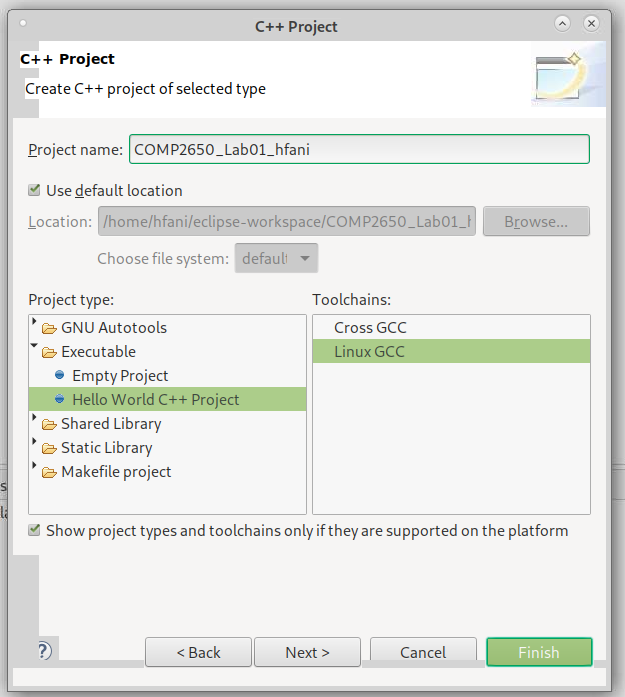


Now, you can create a C++ project:

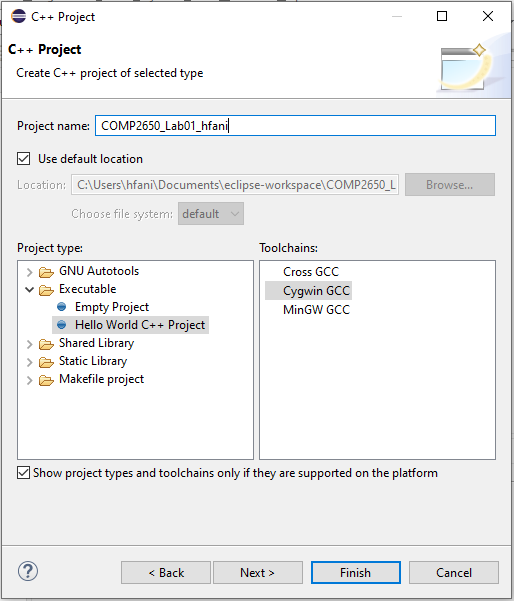


Next, depending on the operating in your computer, you select the compiler. Do NOT select the Cross GCC.

On Linux:



On Windows:



**Step2. Writing the First Program**

Now, let’s write a quick program that accepts two Boolean values and returns the AND of them. Recall from math, the truth table for AND operation, also shown as ‘∧’ or ‘&’ is as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x | y | x AND y | x ∧ y | x & y |
| false | false | false | false | false |
| false | true | false | false | false |
| true | false | false | false | false |
| true | true | true | true | true |

We know that C++ language has a type for boolean values, called bool, that accept false and true. However, C language does not have it. So, we can use the standard type int for our program and use 0 for false and 1 for true values.

01 **int** **main**(**void**) {

02

03 **int** x;

04 **int** y;

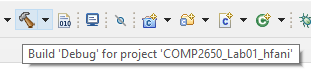
05 scanf("%d", &x);

06 scanf("%d", &y);

07 printf("%d AND %d is %d", x, y, x & y);

08 **return** 0;

09 }

Let’s build the project and run the program by clicking the build icon  to see it’s working fine according to the above truth table. Depending on what computer system and folder you created the program, you see the build messages similar to the followings in the Console tab:

12:27:03 \*\*\*\* Incremental Build of configuration Debug for project COMP2650\_Lab01\_hfani \*\*\*\*

make all

Building file: ../src/COMP2650\_Lab01\_hfani.cpp

Invoking: Cygwin C++ Compiler

g++ -O0 -g3 -Wall -c -fmessage-length=0 -MMD -MP -MF"src/COMP2650\_Lab01\_hfani.d" -MT"src/COMP2650\_Lab01\_hfani.o" -o "src/COMP2650\_Lab01\_hfani.o" "../src/COMP2650\_Lab01\_hfani.cpp"

../src/COMP2650\_Lab01\_hfani.cpp: In function 'int main(int, char\*\*)':

../src/COMP2650\_Lab01\_hfani.cpp:5:19: error: 'scanf' was not declared in this scope

scanf("%d", &x);

^

../src/COMP2650\_Lab01\_hfani.cpp:7:42: error: 'printf' was not declared in this scope

printf("%d AND %d is %d", x, y, x & y);

^

make: \*\*\* [src/subdir.mk:20: src/COMP2650\_Lab01\_hfani.o] Error 1

"make all" terminated with exit code 2. Build might be incomplete.

12:27:03 Build Failed. 3 errors, 0 warnings. (took 419ms)

As you can see, the build failed for our program due to 2 errors in finding scanf and printf functions. Because the linker could not find the declaration for these functions. Recall from the C++ program language course, we know that these two functions are in the standard I/O library in stdio.h. Let’s include this library and built the program again:

01 **#include** <stdio.h>

02 **int** **main**(**void**) {

03

04 **int** x;

05 **int** y;

06 scanf("%d", &x);

07 scanf("%d", &y);

08 printf("%d AND %d is %d", x, y, x & y);

09 **return** 0;

10 }

12:28:47 \*\*\*\* Incremental Build of configuration Debug for project COMP2650\_Lab01\_hfani \*\*\*\*

make all

Building file: ../src/COMP2650\_Lab01\_hfani.cpp

Invoking: Cygwin C++ Compiler

g++ -O0 -g3 -Wall -c -fmessage-length=0 -MMD -MP -MF"src/COMP2650\_Lab01\_hfani.d" -MT"src/COMP2650\_Lab01\_hfani.o" -o "src/COMP2650\_Lab01\_hfani.o" "../src/COMP2650\_Lab01\_hfani.cpp"

Finished building: ../src/COMP2650\_Lab01\_hfani.cpp

Building target: COMP2650\_Lab01\_hfani.exe

Invoking: Cygwin C++ Linker

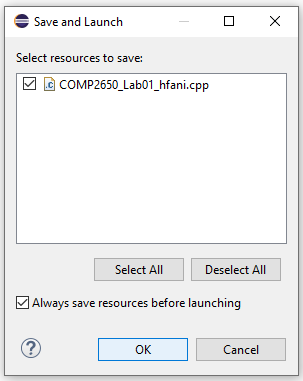
g++ -o "COMP2650\_Lab01\_hfani.exe" ./src/COMP2650\_Lab01\_hfani.o

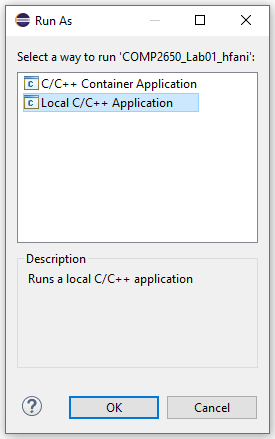
Finished building target: COMP2650\_Lab01\_hfani.exe

12:28:48 Build Finished. 0 errors, 0 warnings. (took 1s.179ms)

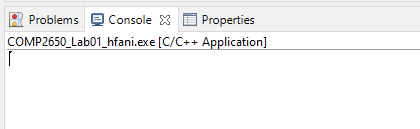
The last message shows that the project has been built successfully with no error.

Now, let’s run the program by clicking the run icon in the toolbar . If this is the first run of the program, you see two dialogs in order to select the program file and the type of application. Select the options as follow:

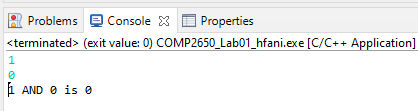




In the Console tab, our program is running and waiting for the inputs, instructed by lines #06 and #07 of the program:



When entering the boolean values, the program outputs the AND result:



Do run the program and try different inputs, e.g., 1 AND 1, 0 AND 0, 0 AND 1, and check whether the program correctly calculates the AND operation.

**Lab Assignment**

You should complete the above program under the name of a project COMP2650\_Lab01\_{UWinID} that firstly outputs a menu of four commands as follows:

Enter the command number:

1. AND
2. OR
3. NOT
4. Exit

Based on the chosen number of commands by the user, the program should then ask for the input(s). For instance, if a user selects (1), the program should accept two inputs as follows:

x =

y =

When the user enters the two boolean values (0 or 1), the program should apply the AND command on the input x and y and print the result and comes back to the main menu.

However, if the user selects (3), the program should ask for one input only:

x =

When the user enters one boolean value, the program then applies the NOT operation on the input and prints out the result and comes back to the main menu. If the user selects (4), the program ends.

**Deliverables**

You will prepare and submit the program in one single zip file COMP2650\_Lab01\_{UWinID}.zip containing the following two items:

1. The entire project folder COMP2650\_Lab01\_{UWinID}
2. The result of the four commands in the file COMP2650\_Lab01\_Results\_{UWinID}.jpg. Simply make a screenshot of the results and save it.
3. A lab report document in the PDF file COMP2650\_Lab01\_Report\_{UWinID}.pdf. It should include:
4. Your name, UWinID, and student number
5. The description of the program that you attached, along with any prerequisites that are needed to build and run the program. *Please note that if your program cannot be built and run on our computer systems, you will lose marks.*

In sum, your final zip file for the submission includes 1 folder (entire project folder), 1 image (results snapshot) and 1 pdf (report). *Please follow the naming convention as you lose marks otherwise.* Instead of {UWinID}, use your own UWindsor account name, e.g., mine is [hfani@uwindsor.ca](mailto:hfani@uwindsor.ca), so,

COMP2650\_Lab01\_hfani.zip

* COMP2650\_Lab01\_hfani
* COMP2650\_Lab01\_Report\_hfani.pdf
* COMP2650\_Lab01\_Results\_hfani.jpg