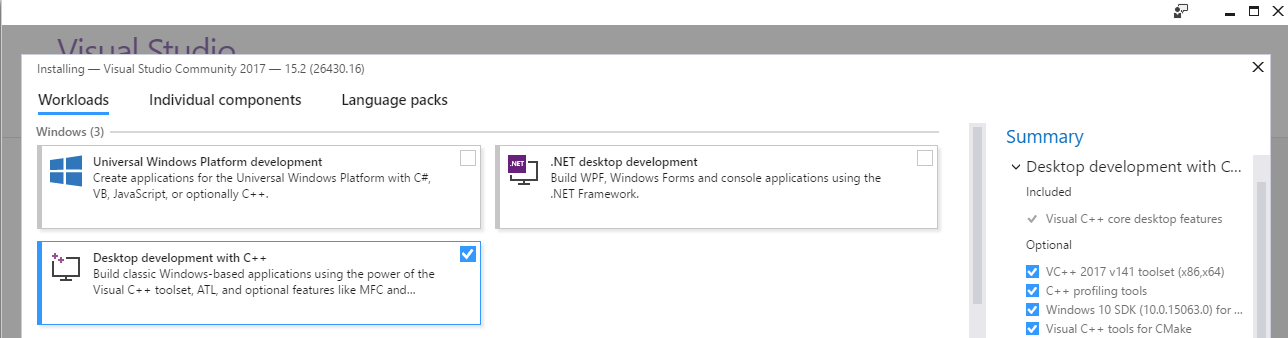
**ECCS 1611 – Programming 1 – Written by Dr. Estell and Dr. Stephany Coffman-Wolph**

**Lab 1 – Introduction to C++ Programming using Visual Studio 2019**

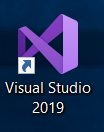
Today’s lab is meant as an introduction, both to the development system that you will be using this semester as well as to the “second chances” laboratory format used in Programming 1. Please note that the purpose of these laboratory sessions is to help you obtain hands-on experience with the course material in a supportive setting – if you have any questions, please raise your hand and either the instructor or one of the teaching assistants will come over to help.

1. Create a C++<my name> or an ECCS 1611 folder on the desktop for the storing of all of your laboratory projects. Create a C++<my name> or an ECCS 1611 folder on your Google Drive – share this folder with your instructor (Dr. Stephany, email: s-coffman-wolph@onu.edu).
2. Make sure that you have downloaded a copy of the file “ECCS 1611 Lab 01 checksheet” for this week’s assignment.
3. If you are using a laptop and need to download Visual Studio, please visit the following site and select "Visual Studio Community 2019": <https://www.visualstudio.com/downloads/> Please note that the download is approximately 2.2 GB, so it’s going to take a considerable amount of time!

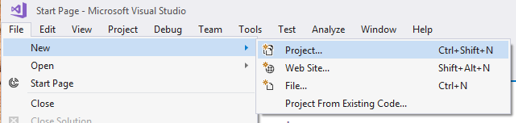
When installing (which will take a while), make sure to check the box for “Desktop development with C++”.

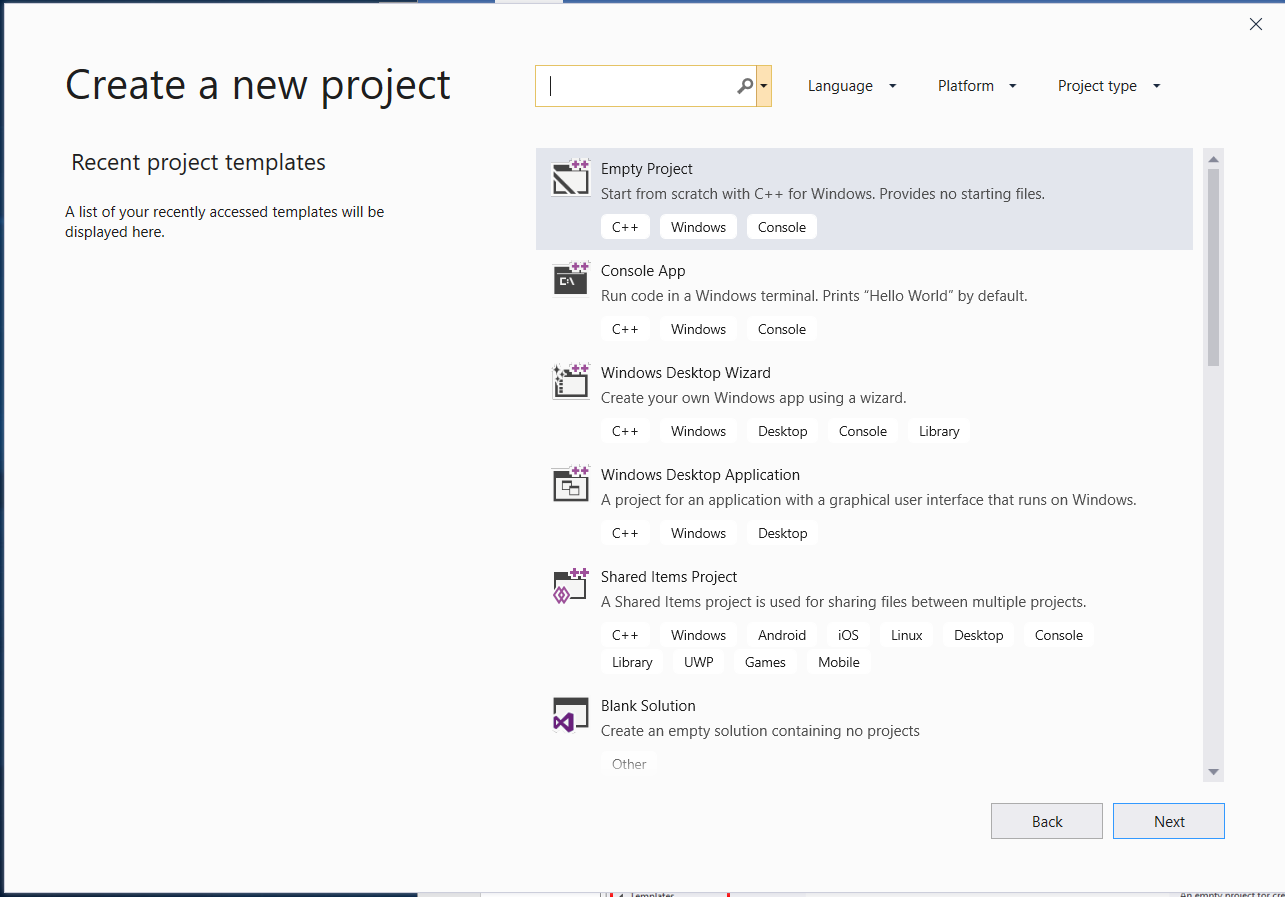


1. Launch Microsoft Visual Studio, either by double-clicking on the appropriate desktop icon or by going to the Start button, expanding the “Visual Studio” folder, and clicking on the entry for “Microsoft Visual Studio 2019”.

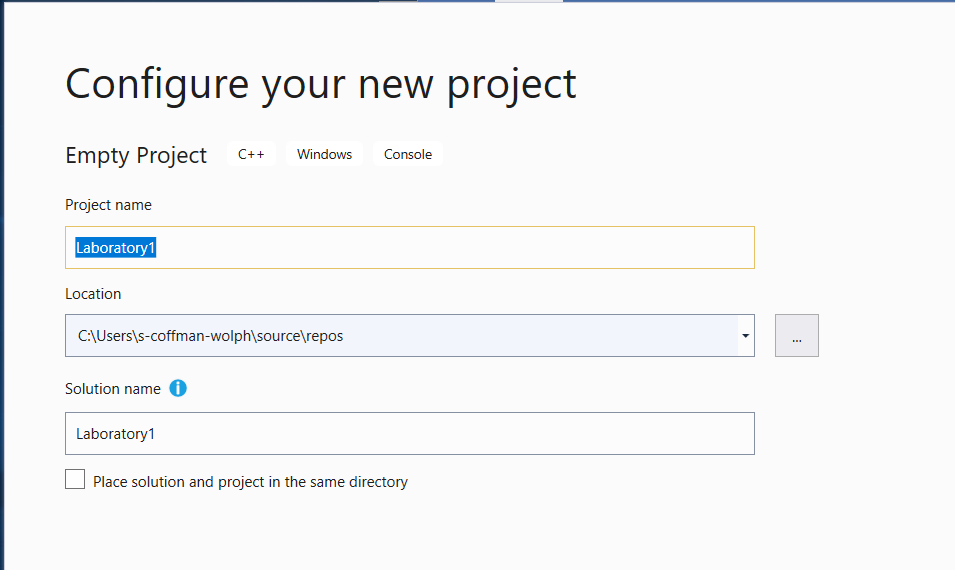


1. If asked to sign in, you should already have a Microsoft account by virtue of being an ONU student (and hopefully this will be the case; if not, then go through the “Sign up” process). Please use your ONU email address and password. You will (eventually) find yourself on the Start Page for Visual Studio.
2. Create a new project by clicking on the “New Project…” link displayed on the Start Page (image on the next page) or by selecting File > New > Project… from the menu. The first time you run Visual Studio it might just skip to the “Create a new project…” dialog.

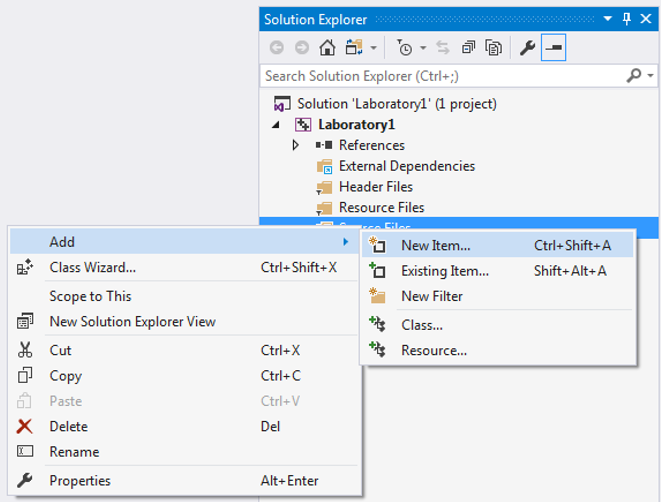




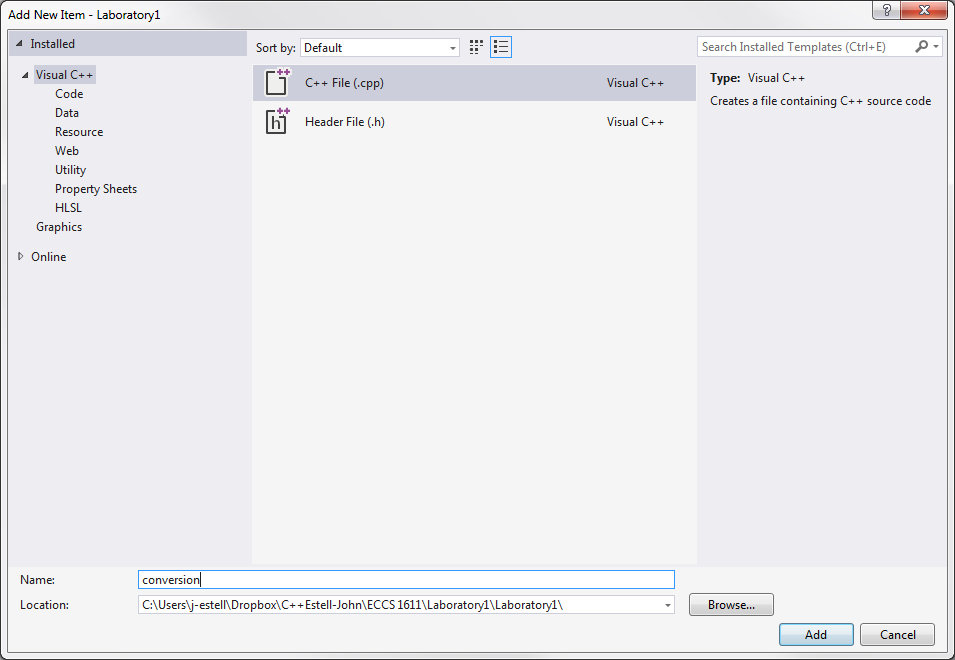
1. In the “Create a New Project” dialog box, first select “Empty Project” as the “Recent project templates” type. Click next.
2. Give “Laboratory1” as the project name. At this time, do **NOT** click on the OK button – we still need to set the location of your project.



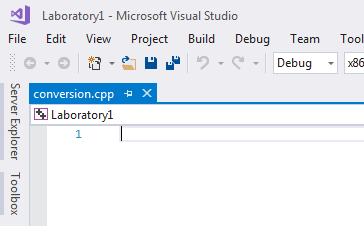
1. For the location of the project, click on the “…” button to display the “Project Location” popup dialog box. Select your folder created in step 1 for the storing of all your laboratory projects. Make sure that folder is selected, then click on the “Select Folder” button.
2. In either the upper right or upper left corner there is now a “Solution Explorer” box containing your project components. Right-click on the “Source Files” entry to display a pop-up menu. Select “Add”; this will display another pop-up menu. Select “New Item…” (This can also be done through using CTRL-SHIFT-A).



1. The “Add New Item” dialog box will now appear. In the templates box that is centered, select “C++ File (.cpp)”. In the “Name” text field at the bottom of the dialog box, enter “conversion” – this will be the name of our source code file. Finally, click on the “Add” button.



1. You should now have a tab labeled “conversion.cpp” containing a blank area for you to enter your code. For this laboratory assignment, the program for you to enter **EXACTLY AS SHOWN** is provided on the next page. Please use the “Tab” key for indentation. You don’t need to understand the code at this time! And, if you do understand code and perhaps see something wrong, **don’t** fix it – you’ll understand why later.



Notes:

* On the first line, please replace <my name goes here> with your name; do not include the angle brackets.
* Also on the first line, please replace <today’s date goes here> with today’s date; again, do not include the angle brackets.

Example: the first line of the program as written by Dr. Estell would appear as follows:

// conversion.cpp – John K. Estell – 29 August 2019

// conversion.cpp - <my name goes here> - <today's date goes here>

// Program will accept input in the form: number letter (e.g. 32 F)

// and will convert the specified temperature from Fahrenheit to Celsius,

// or vice versa, depending on input (either C or F).

#include <iostream> // import library that allows us to use cin and cout

using namespace std; // considered "magic" until later...

const char FAHRENHEIT = 'F';

const char CELSIUS = 'C';

int main( void ) {

// enter "magic formula" for formatting floating-point output values

cout.setf( ios::fixed );

cout.setf( ios::showpoint );

cout.precision( 1 );

// variable declarations

double inputTemperature;

double convertedTemperature;

char temperatureScale;

char convertedScale;

// get the temperature to be converted

cout << "Please enter temperature - indicate scale with C or F: ";

cin >> inputTemperature >> temperatureScale;

// determine if legal scale was entered

if ( temperatureScale != FAHRENHEIT && temperatureScale != CELSIUS ) {

// bailing out of program due to illegal input

cout << "Sorry - temperature scale must be either C or F.\n";

cout << "Press 'Enter' to continue...";

cin.ignore();

cin.get();

return 1;

}

// determine what conversion formula to use

if ( temperatureScale == FAHRENHEIT ) {

convertedTemperature = 5.0 / 9.0 \* inputTemperature - 32.0;

convertedScale = CELSIUS;

}

else { // input was in Celsius

convertedTemperature = 9.0 / 5.0 \* inputTemperature + 32.0;

convertedScale = FAHRENHEIT;

}

// present result and terminate program execution

cout << convertedTemperature << " " << convertedScale << endl;

cout << "Press 'Enter' to continue...";

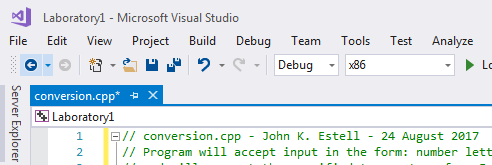
cin.ignore();

cin.get();

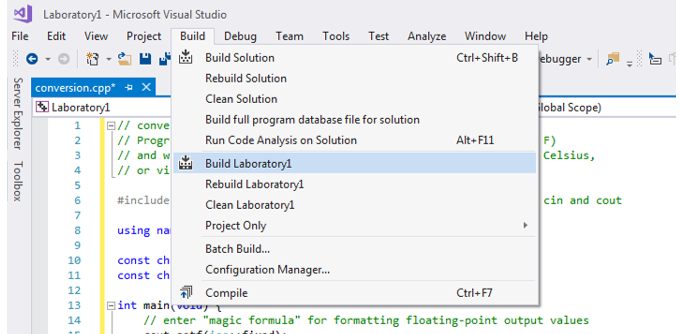
return 0;

}

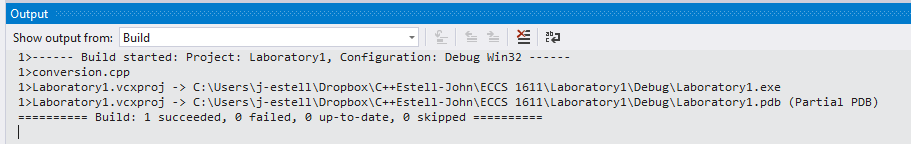
1. Save your program by clicking on the “save file” icon in the tool bar:



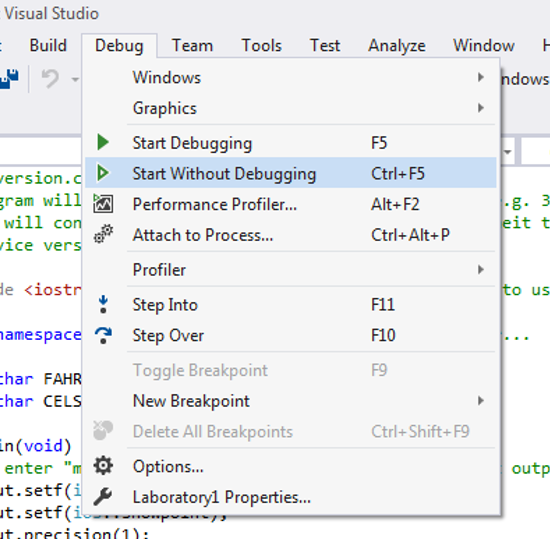
1. The next step is to “compile” or “build” the program; this converts the text that you entered into a format that the computer can use to execute your program. Please open the “Build” menu and select the option “Build Laboratory1”.



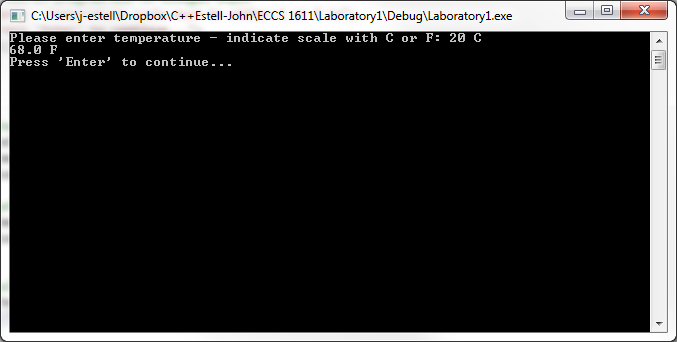
1. The Output box at the bottom of the application will indicate the status of the compilation process, first by displaying “Build started…”, then (hopefully!) the word “succeeded” appears. If there are any errors, then messages appropriate to those errors will appear in the Output box displayed at the bottom of your screen. If this does happen, in this case you have a typo somewhere in your entered code.



1. To run the program, open the “Debug” menu and select the “Start Without Debugging” option.



1. A console (i.e. text-based) window will now appear and you should see the prompt displayed by the program for entering the temperature to be converted. For your first test of the program, use “20 C” (without the quotes) as your input, then press the “Enter” key. You should get “68.0 F” as your answer, as shown below.



1. Now press the ‘Enter’ key as directed – this will dismiss the console window.
2. At this time, bring up the “ECCS 1611 Lab 01 checksheet”. There are a total of six data values provided with which you are to test your program. For each datum, please record the resultant output in the provided space under the “First Attempt” column. Use the word “error” to represent the error message output.
3. If you have an incorrect result for one or more inputs (**and in this case you will, as one error has been deliberately placed in the code you typed in**), find and correct this error. When you believe that you have fixed the problem, rerun the program just for those inputs with incorrect outputs and place the resultant output under the “Additional Attempts” column. Do not worry if your second attempt is also incorrect – just keep working with you until you solve the problem. **Only your final answer is considered for grading purposes**. If you are having trouble, please ask the instructor or lab TA for some help or hints!
4. When you have completed running your program using all the test data and gotten all the correct answers, create a Screenshot (or screenshots). We will use this to check for correctness and mark your assignment accordingly.
5. In addition to requests to execute your program with specified test data, you will occasionally be asked to provide responses to questions embedded within the laboratory write-up. At this time, please answer the following question in the space provided on your checksheet: what do you believe is the reason that the instructor selected the particular input values specified in Step 18 on the checksheet?

1. When you are finished with the lab, copy all the files from the desktop into your Google Drive folder you created in step 1 of lab.
2. Final Step: **Please turn in your code (i.e., a file called** conversion.cpp**), checksheet, and screenshots on Moodle.** Create a zip file with all of these items in it and upload.