

# CSC 1300 LAB 1

Fall 2024

**IMPORTANT! Make sure to read this document completely so you do not forget any steps and so you will see how you will be graded.**

## STEP ONE: SET UP YOUR LAPTOP!

If you haven't already, make sure you have followed the directions in the "1300\_GETTING YOUR COMPUTER SET UP" document. This zip file can be found under **Content** and then **Module 0: Start Here** and then **Course Requirements & Getting Your Computer Set Up**, and then **1300\_GETTING YOUR COMPUTER SET UP**.

There is also a video available for Windows users: <https://www.youtube.com/watch?v=hLo6vrsCIT0&t=2s>

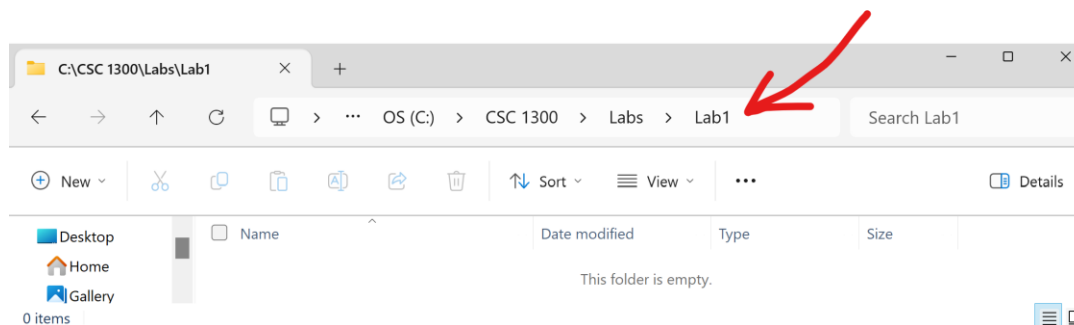
## STEP ONE: SET UP YOUR FILES!

If you haven't already, make sure to set up your files by downloading the zip file from ilearn.

This zip file can be found under **Content** and then **Module 0: Start Here** and then **Course Requirements & Getting Your Computer Set Up**, and then **Set Up Your Folders on Your Computer**.

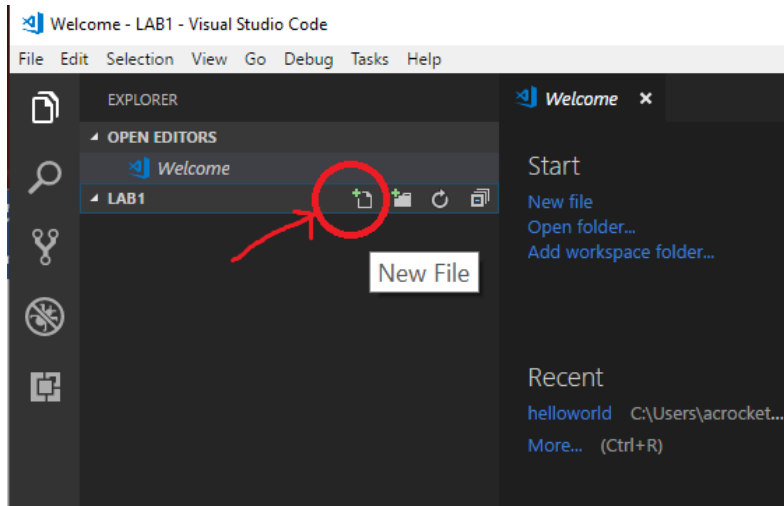
## STEP TWO: WRITE, COMPILE, & RUN YOUR FIRST PROGRAM!

1. Make sure you have set up your folders (STEP TWO above) and so you should have a CSC 1300 > Labs > Lab1 folder as you see in the screenshot below.



2. Open **Visual Studio Code**.
3. Click on **File** and then select **Open Folder**. Select the **Lab1** folder.

4. Click on the New File icon to create a new file.



5. Type **lab1.cpp** to create a file with the name lab1.cpp.
6. Type the following lines of code exactly the way they are below in your lab1.cpp source file but **beside "Author", type your first and last name**.

**DO NOT COPY & PASTE! YOU MAY GET SYNTAX ERRORS BECAUSE OF HIDDEN CHARACTERS!**

```
/*
    Title:      lab1.cpp
    Author:
    Date:       January 2023
    Purpose:    Convert Celsius to Fahrenheit
*/

#include <iostream>
using namespace std;

int main()
{
    double fahrenheit, celsius;
    cout << "\n\nWhat is the temperature in Celsius? ";
    cin >> celsius;
    fahrenheit = celsius * (9.0/5.0) + 32;
    cout << "\nRESULT: " << celsius << " degrees Celsius is ";
    cout << fahrenheit << " degrees Fahrenheit.\n\n";
    return 0;
}
```

### Sample Output

User input is highlighted in **yellow**.

```
What is the temperature in Celsius? -4.8
RESULT: -4.8 degrees Celsius is 23.36 degrees Fahrenheit.
```

7. Now, open either the **Command Prompt** (on WINDOWS) or the **Terminal** (on MAC). Do not use an IDE for this assignment.

8. Using the appropriate DOS or LINUX commands, navigate to the location where your program is located on your computer in your Command Prompt/Terminal.
- You are likely already in your C: drive and so you now need to navigate to your Documents/CSC 1300/Lab/Lab1 folder. You can do this with the `cd` command, which stands for "change directory".
    - Type `cd Documents`
    - Type `cd "CSC 1300"` (note you need the "" quotes because of the space)
    - Type `cd Labs`
    - Type `cd Lab1`

### Helpful Hints

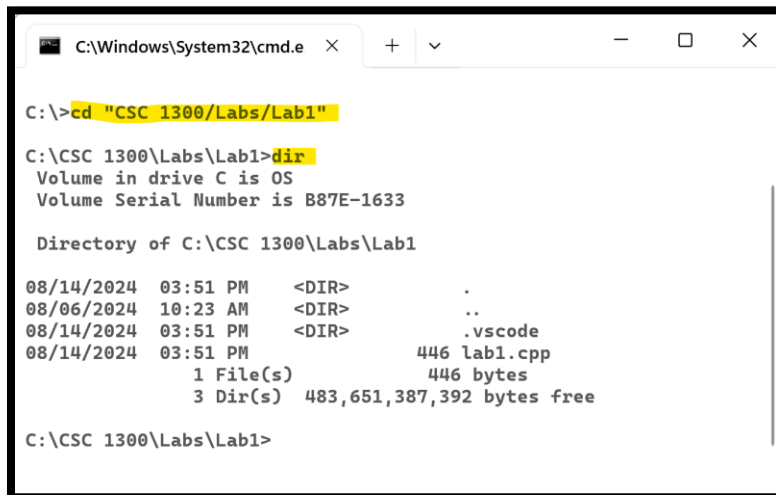
You may also change directly into the LAB1 folder from the C: drive by typing `cd "Documents/CSC 1300/Labs/Lab1"`.  
Also, if you ever need to back up a directory, instead of going forward, you can type `cd ..` and that will bring you back to the parent directory.

9. You should now be inside the Lab1 folder in the Command Prompt/Terminal and this should be the folder that contains your source file named `lab1.cpp`. To make sure, you can type:

WINDOWS- `dir`

MAC- `ls`

and this will list out the files and directories in your current folder. I can see by doing this that `lab1.cpp` is inside this directory.



```
C:\Windows\System32\cmd.e x + v - □ X

C:\>cd "CSC 1300/Labs/Lab1"
C:\CSC 1300\Labs\Lab1>dir
Volume in drive C is OS
Volume Serial Number is B87E-1633

Directory of C:\CSC 1300\Labs\Lab1

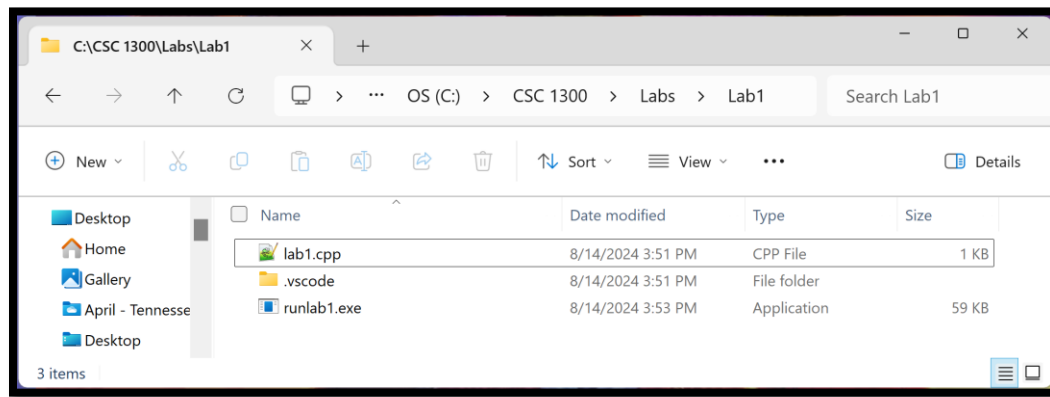
08/14/2024 03:51 PM <DIR>      .
08/06/2024 10:23 AM <DIR>      ..
08/14/2024 03:51 PM <DIR>      .vscode
08/14/2024 03:51 PM          446 lab1.cpp
                   1 File(s)      446 bytes
                   3 Dir(s)  483,651,387,392 bytes free

C:\CSC 1300\Labs\Lab1>
```

10. Now you can compile. The compiler command to compile a single file into an executable (runnable) file is:
- ```
g++ sourcefilename.cpp -o executablefilename
```
11. Where `sourcefilename` is the name of the file you want to compile and `executablefilename` can be anything you want it to be. I usually name my executable file the same name as the source file. You can also leave off the `executablefilename` and your executable file will be created as `a.exe` on windows or `a.out` on a Mac.
12. So, to compile this lab assignment, you can type
- ```
g++ lab1.cpp -o runlab1
```

If I typed `dir` now (or `ls`) then I will see both `lab1.cpp` and `runlab1.exe` (Windows) or `runlab1.out` (Mac).

Here is a screen capture of my folder after compiling:



13. Run your program.

MAC command to run program- `./executablefilename`

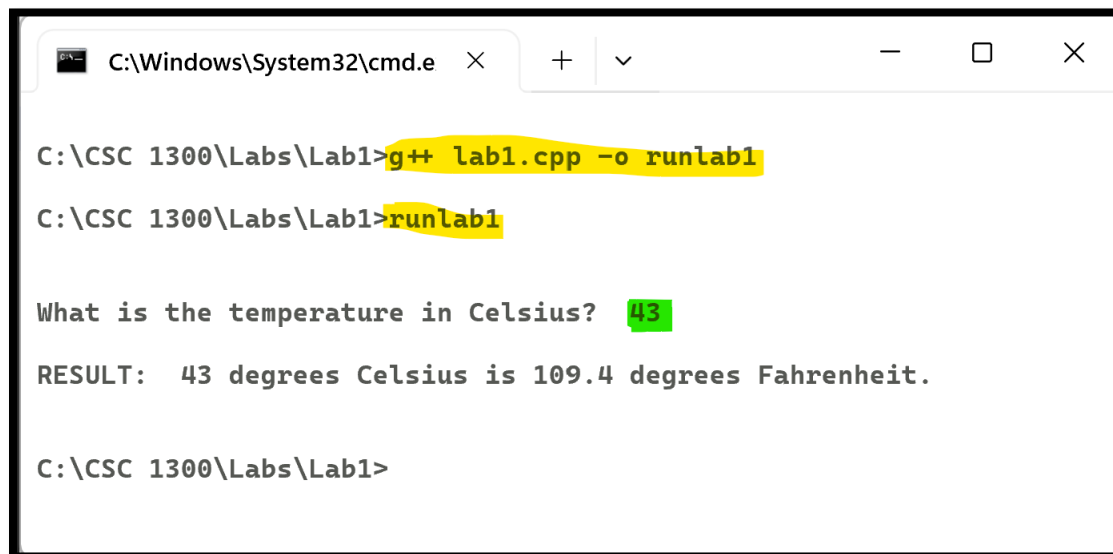
WINDOWS command to run program- `executablefilename`

For this lab, you will replace `executablefilename` with `lab1`:

MAC- `./runlab1`

WINDOWS- `runlab1`

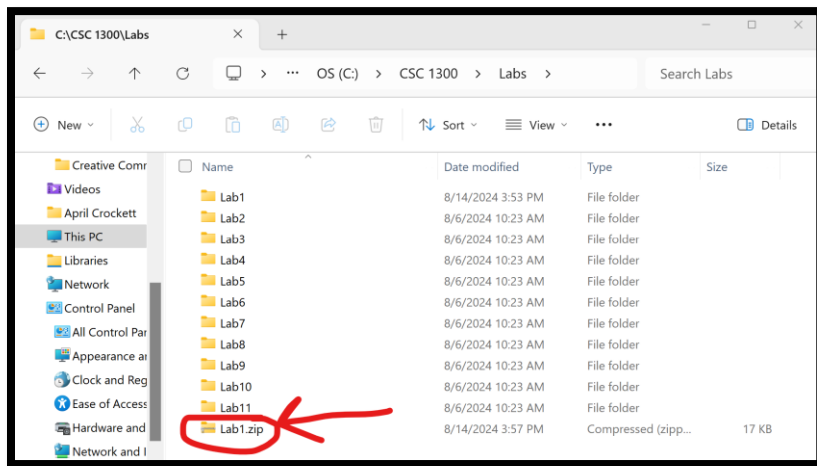
Here is a sample showing how it will look when you compile & run on a windows computer:



14. Zip your **Lab1** folder, which contains **lab1.cpp**. In ALL labs, you will submit your work in a zipped folder.

- WINDOWS** – select the LAB1 folder you want to zip. Then, right-click on the folder, then point to **Send to** and select **Compressed (zipped) folder**. You can find the newly created .zip folder in the same directory.
- MAC** – select the LAB1 folder you want to zip. Then, right-click on the folder and select **Compress Items**. You can find the newly created .zip folder in the same directory.

Below is an image of the folder I want you to submit – it should be the zipped/compressed folder.



## STEP THREE: FILL OUT THE LAB REPORT

You will fill out this lab report for every lab and it is part of your grade. To get credit, you must upload a screenshot of the confirmation page to this lab assignment.

Name your screenshot **lab1ReportProof**.

**Lab Report Link:** [https://tntech.co1.qualtrics.com/jfe/form/SV\\_d6BGc6kzQdSvBmS](https://tntech.co1.qualtrics.com/jfe/form/SV_d6BGc6kzQdSvBmS)

## WHAT TO TURN IN

Upload your **zipped file** containing **lab1.cpp** and **lab1ReportProof.png** to ilearn under **Assignments**, and then **Lab 1** under the “Lab Assignments” category.

# LAB ASSIGNMENT RUBRIC

The following grading rubric will be used for all lab assignments. The maximum number of points you can earn on each assignment is 120 points.

	EXCELLENT	GOOD	FAIR	UNACCEPTABLE
<b>PROGRAM EXECUTION</b>	<b>15 points</b> Program executes correctly with no syntax or runtime errors.	<b>10 points</b> Program compiles and executes but has a minor (easily-fixed) runtime error.	<b>5 points</b> Program doesn't compile due to minor error or errors.	<b>0 points</b> Program does not compile or execute (many errors).
<b>CORRECT OUTPUT</b>	<b>15 points</b> Program displays correct output with no errors.	<b>10 points</b> Output has minor errors.	<b>5 points</b> Output has multiple errors.	<b>0 points</b> Output is incorrect.
<b>DESIGN OF OUTPUT</b>	<b>15 points</b> Program output is perfect with no misspellings and is visually easy-to-read and understand.	<b>10 points</b> Program output is pretty good. There are one or two minor issues (Example: poor use of whitespace)	<b>5 points</b> Program output fair. There are some issues.	<b>0 points</b> Program output is very difficult to read and not user friendly at all.
<b>FOLLOWED SPECIFICATIONS OF ASSIGNMENT</b>	<b>15 points</b> Completely followed all specifications in assignment and used all the correct programming constructs. Programmer followed good programming practices.	<b>10 points</b> Followed specifications in assignment with a minor exception.	<b>5 points</b> Mostly followed specifications. Had a few issues with not following the assignment.	<b>0 points</b> Did not follow the specifications at all or there were multiple differences. Poor programming practices in code
<b>FOLLOWED GOOD PROGRAMMING PRACTICE</b>	<b>15 points</b> Programmer followed good programming practices.	<b>10 points</b>	<b>5 points</b>	<b>0 points</b> Poor programming practices in code.
<b>DOCUMENTATION &amp; READABILITY OF CODE</b>	<b>15 points</b> Program is well commented, and code is readable with appropriate indentions. Program source file or files contain comment block header at top with source code title, author, date, and purpose.	<b>10 points</b> Program readability is good but there are a few issues with indentions or lacking some comments.	<b>5 points</b> Program readability is fair. Maybe lacking comments or inconsistent indentions.	<b>0 points</b> Horrible readability of code. Missing comments and inconsistent indentions.
<b>FILLED OUT LAB REPORT</b>	<b>10 points</b> Filled out lab report and uploaded proof of completion to the lab assignment.			<b>0 points</b> Did not fill out lab report or did not upload proof of completion to the lab assignment.