

The following guide is provided by Dr. Sean Szumlanski, with updates from

- <https://linuxize.com/post/how-to-install-gcc-compiler-on-ubuntu-18-04/>

Getting Started:

First, install a good coding text editor. Some top options include:

- Atom: <https://atom.io/>
- Sublime: <https://www.sublimetext.com/>
- Notepad++: <https://notepad-plus-plus.org/>

Important Note: You do not want to use Notepad this semester. You also cannot use WordPad or Microsoft Word, as those programs add formatting annotations to files (instead of saving them as plain text files).

For Windows Users Only: Make sure Windows is displaying file extensions. This is super important to avoid accidentally naming a file something like “hello.c.c” instead of “hello.c,” which can happen if you have file extensions hidden. The following link has instructions for setting this up:

<https://www.thewindowsclub.com/show-file-extensions-in-windows>

Command Line Compiler Setup:

This section has separate instructions for Windows and Mac users. Mac users should skip to the section titled “Instructions for Mac Users”.

Instructions for Windows Users

1. Set up a Linux bash shell in Windows 10.

- These instructions should work for up-to-date Windows systems:
<https://www.howtogeek.com/249966/how-to-install-and-use-the-linux-bash-shell-on-windows-10/>
- If your system is a bit out of date, you might need these instructions instead:
<https://www.laptopmag.com/articles/use-bash-shell-windows-10-3>.

After you log into your new Ubuntu-based bash environment for the first time, you’ll want to install the gcc compiler like so:

```
sudo apt-get update
```

```
sudo apt install build-essential
```

2. Testing Your Setup:

Next, you need to test that you can compile and run a simple “Hello, world!” program at the command line.

First, write up a simple “Hello, world!” program, and save it to your desktop in a file called Hello.c. How to create your first program is covered in the Week 1 module, but you can also type the following into the code text editor you installed:

```
#include <stdio.h>
int main(void)
{
    printf ("Hello World!\n");
    return 0;
}
```

2. Next, open up your bash prompt again. (This process was described in the setup instructions above. You most likely need to run the program called “Ubuntu” from your Windows Start menu.)

3. When you start the Linux command line program in Windows, it’s operating by default in a weird directory that is not your Windows desktop directory. To reach your desktop at the command line, you need to use the cd command. “cd” stands for “change directory,” as in, “please move to a different directory/folder on my computer).

Type the following command (with careful attention to correct capitalization), and hit [enter]:

```
cd /mnt/c/Users/YOUR_WINDOWS_USERNAME/Desktop
```

When typing that command, you should replace YOUR_WINDOWS_USERNAME with your actual Windows username. On my system, for example, I typed the following: cd /mnt/c/Users/seansz/Desktop

If that doesn’t work for you, you might not be typing the correct username. If that’s the case, or if you can’t remember your Windows username, try typing the following without hitting [enter]: cd /mnt/c/Users/ After you’ve typed all that (without hitting [enter]), hit your [tab] key twice in a row, rapidly. That’ll display the full list of options you have to type next. One of the options that displays will be your username.

If you’re typing your username correctly but you still can’t reach your desktop directory, it might also be that there’s no desktop directory under your username at all. That could happen if you’re syncing all your desktop files with OneDrive. If that’s the case, try typing the following without hitting [enter]: cd /mnt/c/Users/YOUR_WINDOWS_USERNAME/ From there, hit the [tab] key twice in a row, rapidly, to see a list of all the folders in that directory.

If there’s a directory called “OneDrive” (and no directory called “Desktop”), you probably need to type the following to reach your desktop directory:

```
cd /mnt/c/Users/YOUR_WINDOWS_USERNAME/OneDrive/Desktop
```

4. Once you’ve reached your desktop directory, you can compile and run the Hello.c program you wrote by typing the following two commands. The gcc command is used to compile Java source files. The ./ command is used to run your program.

```
gcc Hello.c
```

```
./a.out
```

“a.out” is the default name for compiled programs. If you would like to choose your own name, use the -o compiler tag:

```
gcc hello.c -o hello
```

This tells the compiler to name the executable hello instead of a.out. To run it use the new name:

```
./hello
```

(Optional) Abbreviating the cd Command:

You're going to have to type that long `cd /mnt/c/Users/username/Desktop` command to reach your desktop every single time you open the command line program in Windows. If you'd rather type something shorter, like `cd desktop`, follow the instructions in this section to set that up:

a. Close the command line window and re-open it.

b. Before typing anything else, type the following command to create a permanent shortcut to your desktop folder (which we will use in step (c), below): `ln -s`

`/mnt/c/Users/YOUR_WINDOWS_USERNAME/Desktop desktop` When typing that command, you should replace `YOUR_WINDOWS_USERNAME` with your actual Windows username.

On my system, for example, I typed the following: `ln -s /mnt/c/Users/seansz/Desktop desktop` You could also replace the word "desktop" at the very end of that command with anything else, like "desk" or "windows."

c. That's it! That effectively turns "desktop" into an alias for

`/mnt/c/Users/YOUR_USERNAME/Desktop`, so now when you open the bash prompt in Windows, all you have to do to reach your desktop folder is type following: `cd desktop`

Instructions for Mac Users

1. If you're on a Mac, you may already have a C compiler installed by default. To check, open the command line by launching the "Terminal" program found in `/Applications/Utilities/`, and then type the following command: `gcc -version`

If you have a C compiler installed, that command should produce some output telling you what version of the compiler you're working with (e.g., you might see: `gcc 7.4.0`). In that case, you're ready to skip to step (2), below.

If you do not have a C compiler installed, you'll most likely see some sort of command not found error. The easiest method to install the compiler is to download Xcode from the app store.

Next, check that you can compile and run a simple "Hello, world!" program at the command line.

How to create your first program is covered in the Week 1 module, but you can also type the following into the code text editor you installed:

```
#include <stdio.h>
int main(void)
{
    printf ("Hello World!\n");
    return 0;
}
```

Save this to your desktop in a file called `Hello.c`

Next, open the command line by launching the “Terminal” program found in /Applications/Utilities/, and then type the following command to navigate to your desktop folder: `cd ~/Desktop` Note: The ~ above refers to your home directory: /Users/YOUR_MAC_USERNAME c.

Once you’ve reached your desktop directory, you can compile and run the Hello.c program you wrote by typing the following two commands:

. The gcc command is used to compile C source files. The ./ command is used to run your program.

```
gcc Hello.c
```

```
./a.out
```

A.out is the default name for compiled programs. If you would like to choose your own name, use the -o compiler tag:

```
gcc hello.c -o hello
```

This tells the compiler to name the executable hello instead of a.out. To run it use the new name:

```
./hello
```