

B. Installing Necessary Software

B.1. For Microsoft Windows

Windows 10:

In August of 2016 Microsoft released an update for Windows 10 that allows you to install a complete development environment, including the tools used in this book, fairly easily.

1. To begin, enable Windows Subsystem for Linux (WSL):
 - Click the start button on the taskbar and search for “Turn windows features on or off”. This should show you a box like Figure B.1. Place a check in the box beside “Windows Subsystem for Linux”, then click OK.
 - Restart your computer (this is necessary before going on).
2. Install the “Ubuntu” app for WSL:
 - Click the start button and search for “Microsoft Store”. Open the Microsoft Store app. (See Figure B.2.)
 - Within the Store app, search for “Ubuntu”.
 - Select the orange Ubuntu app whose name is just “Ubuntu”, with no version number.
 - Click the “Get” button to install this app. (See Figure B.3.)
3. Start the “Ubuntu” app and configure it:
 - From the Start Menu, click the “Ubuntu” icon, as shown in Figure B.4.
 - The app will open (see Figure B.5) and begin setting itself up. This might take several minutes.

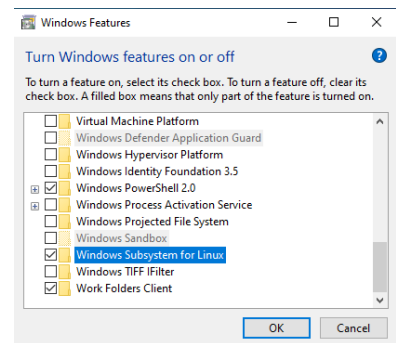


Figure B.1: Turning on “Windows Subsystem for Linux”

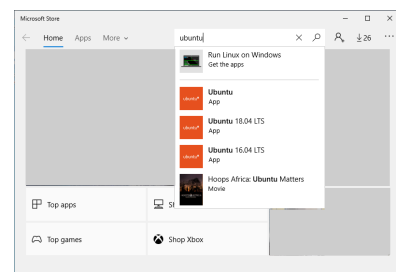


Figure B.2: Searching for “Ubuntu” in the Microsoft Store

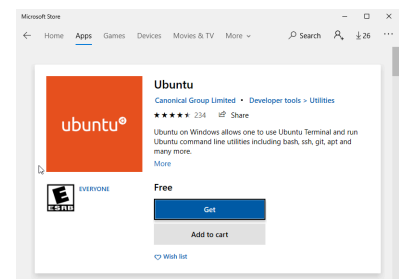


Figure B.3: Click “Get” to install the Ubuntu app.

- Near the end of this process, the app will ask you to supply a user name and a password for use within the app (see Figure B.6).

4. Updating and installing software in the app:

Type the following commands in the app's window:

```
sudo apt update
sudo apt -y upgrade
sudo apt -y install g++ nano gnuplot-x11
sudo apt -y reinstall gnome-icon-theme
```

The first command will ask for the user name and password you entered in the preceding step. If any of the commands asks you about restarting services, answer "yes".

This will install the specific software (*g++*, *nano*, and *gnuplot*) used in this book.

5. Next, type the following command:

```
echo export DISPLAY=localhost:0 >> $HOME/.bashrc
```

6. In order to use this version of *gnuplot* you'll need to install one more piece of Windows software, called an "X server". This allows the tools in the development environment (installed in the steps above) to display graphics on your screen. To install it, download and install VcXsrv from here:

<https://sourceforge.net/projects/vcxsrv/>

7. After you've installed VcXsrv, click the Start button, and type "xlaunch" and press the Enter key. A window like Figure B.7 should appear. Keep clicking "Next" until you get to the dialog box shown in Figure B.8.

Click "Save Configuration" and save the configuration as "config.xlaunch" on your desktop.

Now hold down the "Windows" key and type R, press "Enter", then type:

```
shell:startup
```

and press the Enter key. This will open up your startup folder. Drag the "config.xlaunch" icon from your desktop into this folder.

8. Now restart your computer. You should be able to get a command window by clicking "Ubuntu" in the Start Menu. All the work in this book can be done in this window.

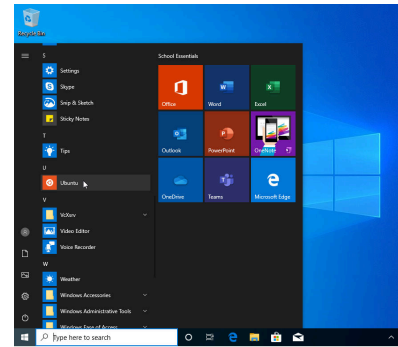


Figure B.4: Starting the "Ubuntu" app

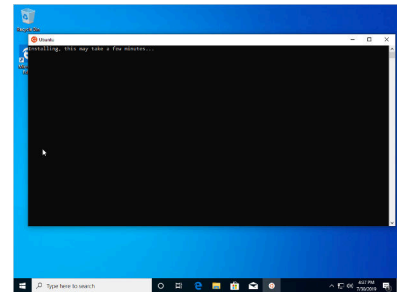


Figure B.5: "Ubuntu" app beginning configuration

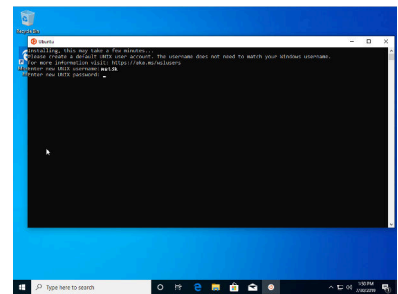


Figure B.6: Creating a user name and password for the app

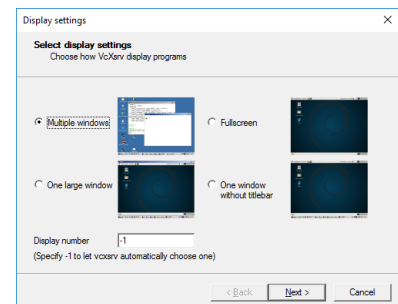


Figure B.7: Running *xlaunch*.

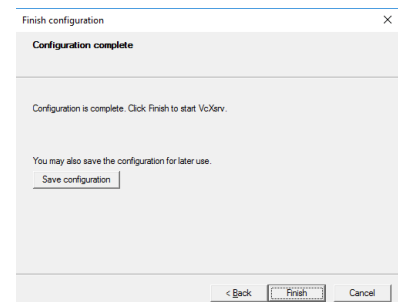


Figure B.8: Saving *xlaunch* configuration.

Windows 7 and 8:

For Windows 7 and 8 we've created a bundle of useful free Windows software that you can download and install on your own computer. The bundle includes *g++*, *nano*, and *gnuplot*, among other tools.

You can download the bundle from the following address:

<http://faculty.virginia.edu/comp-phys/phys1660/2015/software/phys1660-bundle-setup.exe>

The downloaded file will be called `phys1660-bundle.exe`. Run it to install the software. Once installed, you should see a new icon for *MSys* on your desktop. Double-click this to open a command window. From the command window, you can use the *g++*, *nano*, and *gnuplot* commands described in this book.

B.2. For Linux

On Debian, Ubuntu, Mint and similar distributions, type:

```
sudo apt update
sudo apt -y upgrade
sudo apt -y install g++ nano gnuplot
```

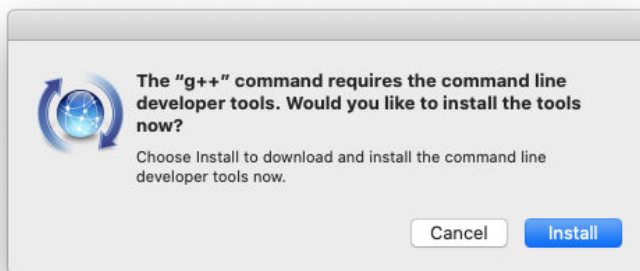
On Fedora, CentOS, Red Hat and similar distributions, type:

```
sudo yum install gcc-c++ nano gnuplot
```

B.3. For Apple MacOS

For Mac users, Apple includes many of the tools you'll need, but they might need to be "activated". You'll also need *gnuplot* and *Xquartz*, upon which *gnuplot* depends.

1. To get a command window, click any blank spot on your desktop background, then go to the "Go" menu at the top of the screen and select Utilities->Terminal.
2. Inside the terminal window, type `g++`. The first time you do this you'll see a message like the one below.



Click "Install" to install the command line developer tools. Now you should be able to use the `g++` command as we do in this book.

3. In order to use *gnuplot* under OS X, you'll also need to install two more things. The first is *XQuartz*, which you can get here:
<http://xquartz.macosforge.org/landing/>
4. **IMPORTANT:** After you've installed *XQuartz*, you must log out of your computer and log back in to complete the installation. (If you

don't do this, *gnuplot* may not install or work correctly.)

5. The last thing to install is *gnuplot* itself, which you can get here:

https://csml-wiki.northwestern.edu/index.php/Binary_versions_of_Gnuplot_for_OS_X

Download the current version of *gnuplot* from the site above.

6. **IMPORTANT:** After you've downloaded the file, hold down the Ctrl key while clicking on it. If you don't hold down the Ctrl key, the computer might refuse to run the installer. Then proceed to install the package as usual.

You should now be able to use the *g++*, *nano*, and *gnuplot* in your terminal window.