



CS-1004, Introduction to Program-
ming for Non-Majors, C-Term 2015

Setting up Python 3.4, numpy, and matplotlib on your Macintosh or Linux computer

Hugh C. Lauer[©]
Adjunct Professor
Worcester Polytechnic Institute

Programming assignments in CS-1004 will be in the programming language *Python* — specifically, version 3.4 of *Python*. In addition, you will need several *Python* packages, including one called *numpy* (meaning “Numerical Python”) and one called *matplotlib*, a *Python* version of the popular *Matlab* system. The first part of this document provides instructions for installing *Python 3.4* and packages such as *numpy* and *matplotlib* on Macintosh platforms. [Click here](#) to skip forward to the Linux section of this document.¹

In general, it is expected that assignments will be compatible among Windows, Macintosh, and Linux systems, provided that they all use compatible versions of *Python* and *numpy*.

Note: There are two different, incompatible versions of *Python* in general use around the world — *Python 2.7* and *Python 3.4*. Significant changes to the *Python* language were made between *Python 2.x* and *Python 3.y* (for all values of x and y). The *Python 3* language is cleaner, more self-consistent, and more user-friendly. Programs written for versions of *Python 2* will not necessarily run on *Python 3* installations; if they do run, they may get different answers to the same problem.

That being said, a lot of legacy *Python 2* code is still in use, and new *Python 2.7* code is still being written and distributed by organizations that have not yet upgraded to *Python 3*. Not all *Python 2* packages have been ported to *Python 3*.

Note 2: Both Mac OS X and Linux come with *Python 2.7* already installed for their own needs. That version of *Python* must not be deleted or disturbed when installing *Python 3.4* and its packages.

[©] Copyright 2015, Hugh C. Lauer. All rights reserved. Permission is given for use in courses at Worcester Polytechnic Institute, Worcester, Massachusetts.

¹ If you have a Windows computer or laptop, please refer to this documents instead:– [docx](#), [pdf](#)

Installing Python 3.4 on Macintosh^{2,3}

Installing *Python 3.4* on Mac OS X presents some challenges:–

- Mac OS X has a very robust protection mechanism against malware, particularly the inadvertent installation of programs of unknown provenance. This protection must be overridden in order to install *Python 3.4*.
- Packages such as *numpy 1.9.1* and *matplotlib 1.4.2* are delivered in forms that require installation via commands in a *Terminal* window for installation.
- The *IDLE* tool of *Python 3.4* has a dependency on *Tcl/Tk*, an open-source toolkit for building user interfaces. However, *Python 3.4* requires a more recent version than is already installed on Mac OS X.

First, download the following two installers to a convenient folder or directory by clicking on these links:– [ActiveTcl 8.5.17.0](#), and [python-3.4.2-macosx10.6](#). Alternatively, you may browse to

<http://www.cs.wpi.edu/~cs1004/c15/>

and download the files from there. Note that installations on the Macintosh require administrator privileges.

ActiveTcl 8.5.17.0

Open the **ActiveTcl 8.5.17.0** file that you just downloaded. You will be asked to confirm your action, and then it displays the following dialog box:–

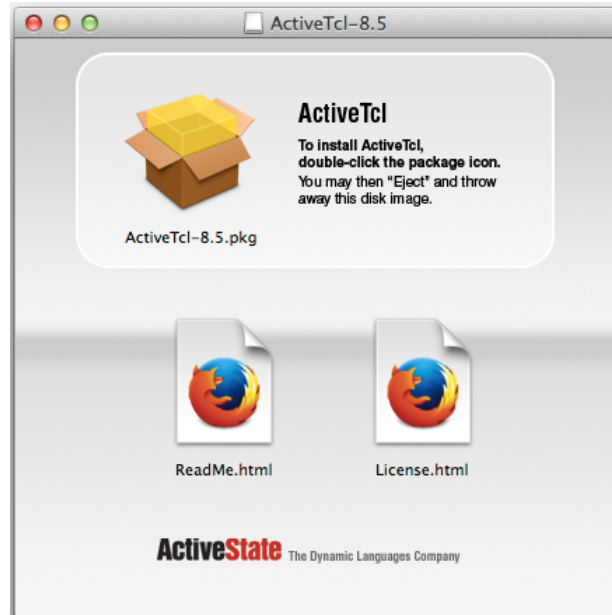


Figure 1

² It is useful to print out the relevant section of this document. If you read it on-screen, the dialog boxes of the installation tend to obscure the instructions of the document, just when you need them the most!

³ These instructions have been tested on MacOS X 10.9.4. There may be slight differences with other versions of MacOS.

Double-click on the icon that looks like a shipping carton. Under most circumstances, you will next be presented with the following dialog box of the Macintosh *Gatekeeper*.—

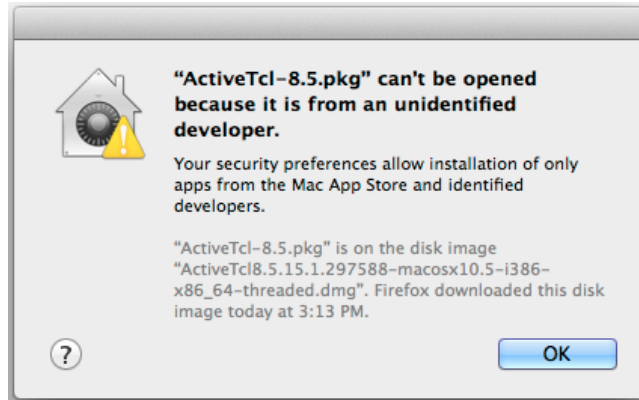


Figure 2

Click *OK*, and then open *System Preferences* under the *Apple* menu. In the *System Preferences* folder, open the *Security and Privacy* icon, shown circled in Figure 3 to open the *Security & Privacy* control panel of Figure 4.



Figure 3

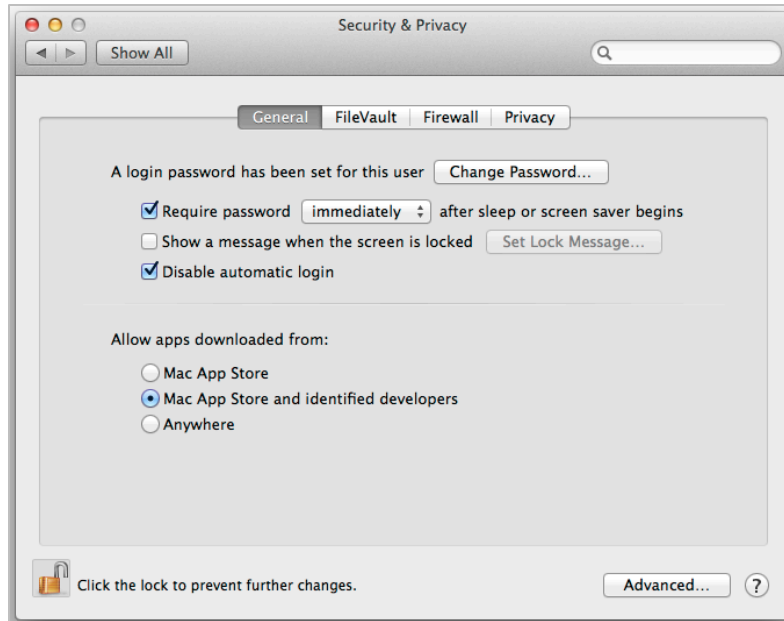


Figure 4

In the *Security & Privacy* panel, first click the padlock in the lower left to unlock the settings. (The padlock is shown already unlocked in the figure. In normal cases, it would be locked.) You probably need to supply the *Administrator's* login and password at this point. Next, select *Anywhere* for the option under *Allow apps downloaded from*.

Finally, go back to Figure 1 and double-click again on the package icon to install *ActiveTcl 8.5.17.0*. At this point, the installation of that package should proceed normally.

Installing Python 3.4

Next, you will install *Python 3.4*. Open the **python-3.4.2-macosx10.6.pkg** file that you downloaded on Page 2 above. This should bring up a dialog resembling Figure 5 below.



Figure 5

The message in the dialog box is scrollable and includes information about specific versions of *Tcl/Tk*. We have already addressed this issue with the installation of *ActiveTcl 8.5.17.0*.

If the installer complains that there is already a version of *Python 3* installed on your computer, please seek help or contact the Professor. You must have *Python 3.4.2* for this course. Earlier versions should be upgraded.

Click *Continue* to bring up another dialog:–



Figure 6

This is a warning that the Macintosh *Gatekeeper* policy does not, by default, trust the *Python 3.4.2* installer. We have already addressed that issue in Figure 4 above. Click *Continue* to go forward with the installation. The next dialog is from *Python* itself:–

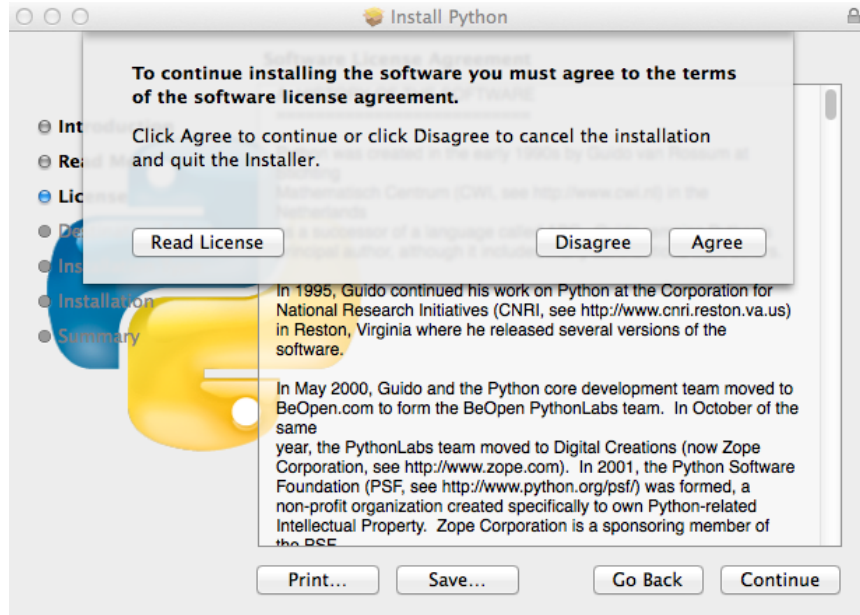


Figure 7

Click *Agree* to continue the installation. The installation itself will take several minutes. After it has completed, you should find a folder named *Python 3.4* in your *Applications* folder, as shown circled in solid black below.



Figure 8

Open the *Python 3.4* folder to show the next figure:–

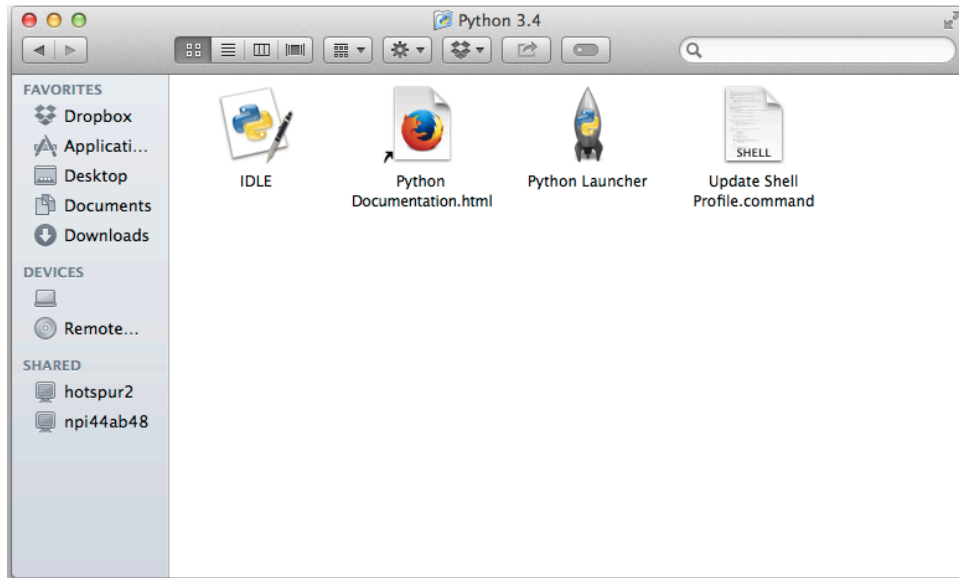


Figure 9

Create an *Alias* for *IDLE* and put it in a convenient place so that you can access it easily — for example, your desktop. Double-click this alias to bring up the following window:–

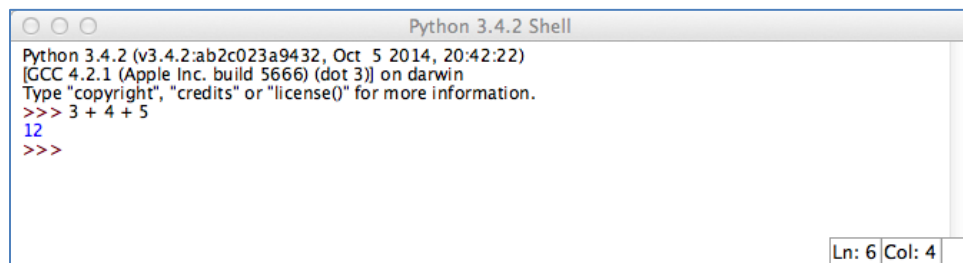


Figure 10

This is *IDLE*, the *Python* command prompt and graphical user interface. This is where we will start all programs and projects in this course. For now, simply type any *Python* statement or expression after the “>>>” prompt. For example, in Figure 10, the expression $2 + 3 + 4$ was typed and *Python* responded with the value 9. Continue testing by typing out the code on pages 10-11 of the textbook, just to make sure that your installation works as expected.

Note: *IDLE* is essentially the same across *Windows*, *Macintosh*, and *Linux* platforms. It is used the same way in all three.

Reminder: After you have completed the entire installation of *ActiveTcl 8.5.17.0* and *Python 3.4.2*, return to the *Security & Privacy* control panel of Figure 4 and restore the *Allow apps downloaded from* option to its original value (*Mac App Store and identified developers*). Then click on the padlock to *lock* it, thereby preventing further changes.

Installing *matplotlib*, *numpy*, and other packages

One of the many benefits of *Python* is the vast number of third-party packages that can be downloaded and used by your *Python* programs. Many of these are open-source and free. For this course, we will use at least the following:–

- *matplotlib* (a package for creating 2D plots and graphs similar to *Matlab*),
- *numpy* (meaning “Numerical Python,” a package for efficient handling of large arrays of numerical data), and
- *graphics.py*, a simple tool written in *Python 3* and created by the textbook author for making simple drawings.

Installing *matplotlib*, *numpy*, and other packages on a Macintosh requires you to type Linux-like or Unix-like commands in a *Terminal* window. If you are comfortable doing this — including the use of the **sudo** command — skip forward to [here](#). If not, it is a good idea to learn about this now.

Administrator privileges

First, give yourself *Administrator* permission at the command line level, so that you can use the **sudo** command with your own password. To get such permission, click on your user name in the upper right corner of the Macintosh desktop. This will open a pull-down menu of user names authorized to use your computer. Select the bottom item, entitled *Users and Groups Preferences*, to bring up a control panel similar to Figure 11 below.

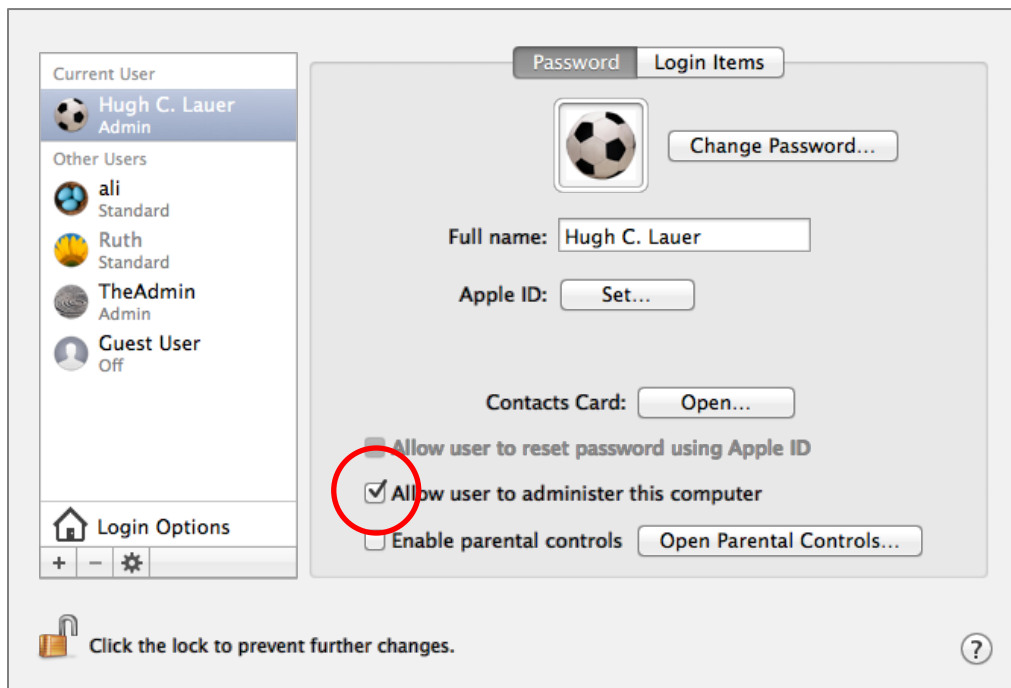


Figure 11

First, click on the padlock in the lower left corner to allow you to edit these settings. You will probably have to supply the *Administrator* user name and password.

Next, select your own user name from the left column. Check the circled option in Figure 11 that says “Allow user to administer this computer.” Finally, click on the padlock in the lower left to prevent further (possibly accidental) changes.

Mac OS X will require you to reboot the computer to make these changes. **Do this now!**

After you have rebooted and logged in again, go back to Figure 8 and open the *Utilities* folder, which is indicated by the dashed circle in the figure. This will bring up the window of Figure 12. Select *Terminal* (highlighted with a red circle) and open it.

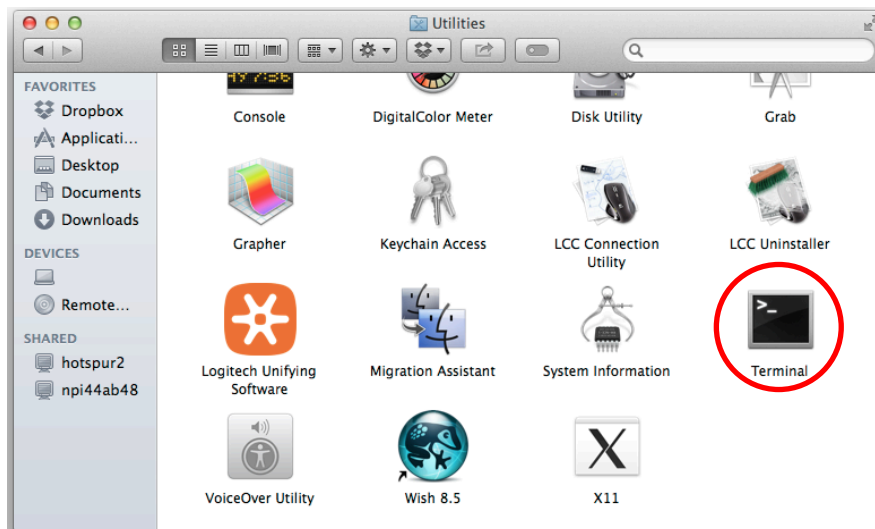


Figure 12

This will bring up a terminal window (also known as a *command shell* in Unix and Linux terminology). In this window, you will type *commands*, each one of which starts with a command name and is followed by zero or more *operands*, which control what the command does. A *command* may work silently and then type out its results in the same window, or it may engage in a textual conversation with you, requiring you to respond, or it may open its own window with its own graphical user interface.

An important command is **sudo**, which means **do** the command that follows on the rest of the line with **su** privileges (i.e., *super-user* privileges, also known as *administrator* privileges). By checking the box in Figure 11, you have authorized yourself to use the **sudo** command. You need to supply your own password when you use **sudo** (not the administrator’s password).

Installing *numpy* and *matplotlib*

Click on the following links to download two files to a convenient folder:–

- [numpy-1.9.1-cp34-cp34m-macosx_10_9_x86_64.whl](#)
- [matplotlib-1.4.2-cp34-macosx_10_9_x86_64.whl](#)

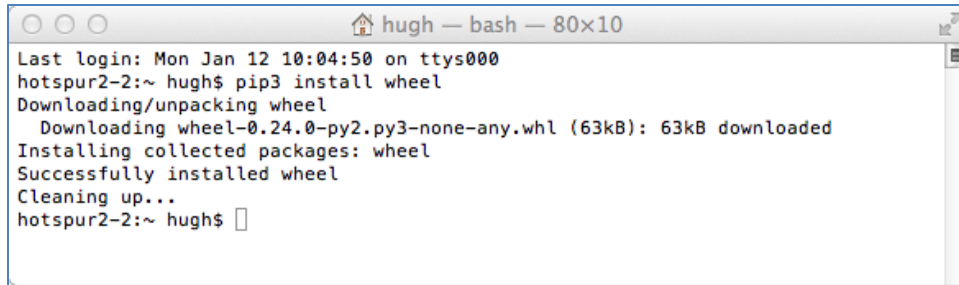
Before you can install *numpy 1.9.1*, you first need to install a *Python* utility that can interpret “wheel” files — i.e., files with the **.whl** suffix.⁴ To do this, open a *Terminal* window and type the command

```
pip3 install wheel5
```

This is illustrated in Figure 13 starting in line 2:–

⁴ The origin of the term **wheel** for this purpose is a mystery.

⁵ **pip** denotes the *Python Installation Program*. **pip3** is the version of **pip** for *Python 3.x*.



```
hugh — bash — 80x10
Last login: Mon Jan 12 10:04:50 on ttys000
hotspur2-2:~ hugh$ pip3 install wheel
Downloading/unpacking wheel
  Downloading wheel-0.24.0-py3-none-any.whl (63kB): 63kB downloaded
Installing collected packages: wheel
Successfully installed wheel
Cleaning up...
hotspur2-2:~ hugh$
```

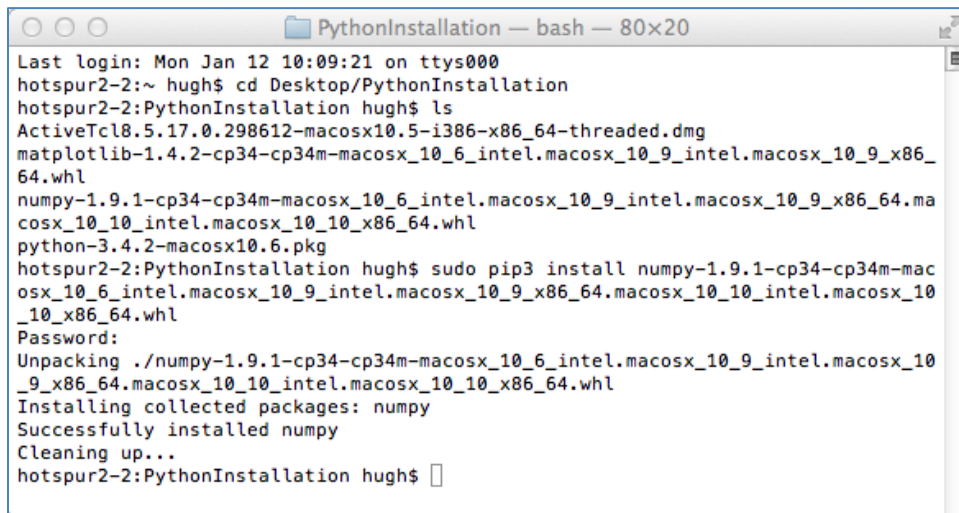
Figure 13

Next, use the **cd** command to change the working directory of the *Terminal* to the folder where you downloaded your files. This is shown on line 2 of Figure 14. List the directory so that you can get the exact names of the file downloaded files. Note that they are *very* long.

Install *numpy* by typing the following command to the *Terminal* window and pasting the file name where the placeholder is shown, so that the command is all on one line:—

```
sudo pip3 install <paste numpy file name here>
```

The entire command should be pasted onto one line, even though it will overflow the line length of the terminal window. This command will install *numpy* and all of its dependencies, as shown in Figure 14 below.



```
PythonInstallation — bash — 80x20
Last login: Mon Jan 12 10:09:21 on ttys000
hotspur2-2:~ hugh$ cd Desktop/PythonInstallation
hotspur2-2:PythonInstallation hugh$ ls
ActiveTcl8.5.17.0.298612-macosx10.5-i386-x86_64-threaded.dmg
matplotlib-1.4.2-cp34-cp34m-macosx_10_6_intel.macosx_10_9_intel.macosx_10_9_x86_64.whl
numpy-1.9.1-cp34-cp34m-macosx_10_6_intel.macosx_10_9_intel.macosx_10_9_x86_64.macosx_10_10_intel.macosx_10_10_x86_64.whl
python-3.4.2-macosx10.6.pkg
hotspur2-2:PythonInstallation hugh$ sudo pip3 install numpy-1.9.1-cp34-cp34m-macosx_10_6_intel.macosx_10_9_intel.macosx_10_9_x86_64.macosx_10_10_intel.macosx_10_10_x86_64.whl
Password:
Unpacking ./numpy-1.9.1-cp34-cp34m-macosx_10_6_intel.macosx_10_9_intel.macosx_10_9_x86_64.macosx_10_10_intel.macosx_10_10_x86_64.whl
Installing collected packages: numpy
Successfully installed numpy
Cleaning up...
hotspur2-2:PythonInstallation hugh$
```

Figure 14

You can test your installation of *numpy* by opening an *IDLE* window, as in Figure 10. Type or paste the following commands into *IDLE*, one line at a time, *exactly* as written:—

```

import numpy as np
np.__version__ 6
a = np.arange(10)
a
b = np.arange(1, 9, 2)
b
c = np.eye(3)
c
d = np.diag(np.array([1, 2, 3, 4]))
d

```

The result should resemble Figure 15 below:–

```

Python 3.4.2 Shell
Python 3.4.2 (v3.4.2:ab2c023a9432, Oct 5 2014, 20:42:22)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>> import numpy as np
>>> np.__version__
'1.9.1'
>>> a = np.arange(10)
>>> a
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
>>>
>>> b = np.arange(1, 9, 2)
>>> b
array([1, 3, 5, 7])
>>>
>>> c = np.eye(3)
>>> c
array([[ 1.,  0.,  0.],
       [ 0.,  1.,  0.],
       [ 0.,  0.,  1.]])
>>>
>>> d = np.diag(np.array([1, 2, 3, 4]))
>>> d
array([[1, 0, 0, 0],
       [0, 2, 0, 0],
       [0, 0, 3, 0],
       [0, 0, 0, 4]])
>>>
Ln: 26 Col: 21

```

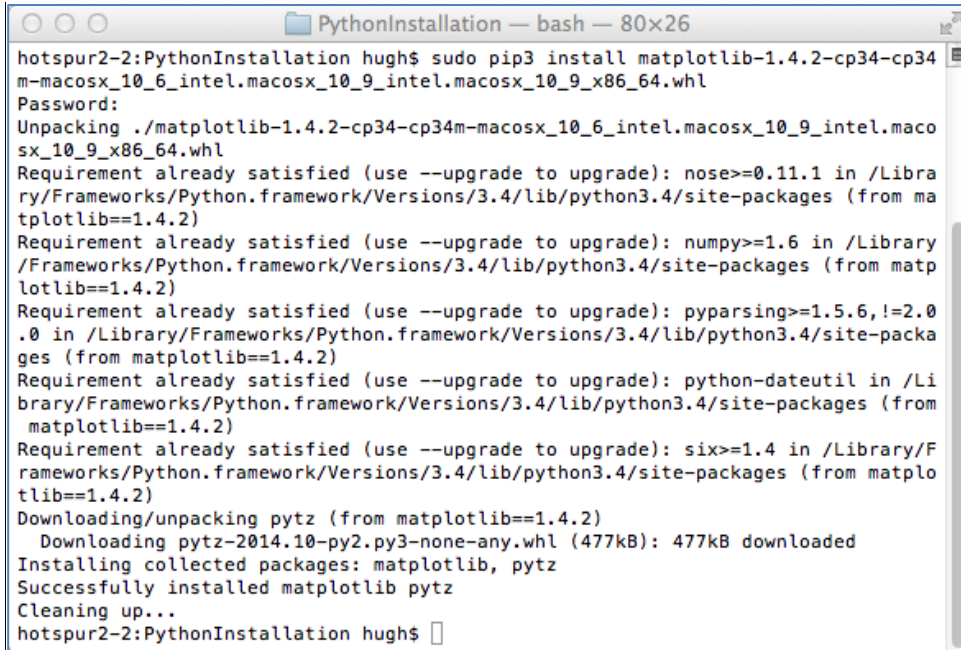
Figure 15

Installation of *matplotlib* is done the same way. Copy the name of the *matplotlib* installation file into the following command into the *Terminal* window, all on one line:–

```
sudo pip3 install <paste matplotlib file name here>
```

This is shown in Figure 16 below. Note that, *matplotlib* has several dependencies; the **.whl** file instructs the **pip3 install** command to retrieve them from the network and install them, also.

⁶ Note that the word “**version**” is preceded by *two* underscore characters and followed by *two* more underscore characters. Note also that the word “**arange**” in several lines has only one ‘**r**’.



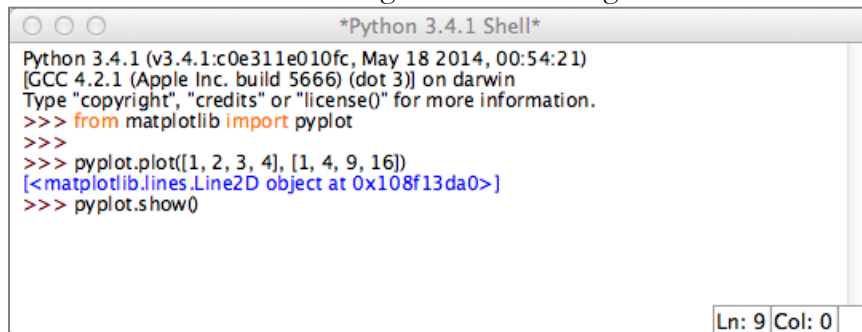
```
hotspur2-2:PythonInstallation hugh$ sudo pip3 install matplotlib-1.4.2-cp34-cp34m-macosx_10_6_intel.macosx_10_9_intel.macosx_10_9_x86_64.whl
Password:
Unpacking ./matplotlib-1.4.2-cp34-cp34m-macosx_10_6_intel.macosx_10_9_intel.macosx_10_9_x86_64.whl
Requirement already satisfied (use --upgrade to upgrade): nose>=0.11.1 in /Library/Frameworks/Python.framework/Versions/3.4/lib/python3.4/site-packages (from matplotlib==1.4.2)
Requirement already satisfied (use --upgrade to upgrade): numpy>=1.6 in /Library/Frameworks/Python.framework/Versions/3.4/lib/python3.4/site-packages (from matplotlib==1.4.2)
Requirement already satisfied (use --upgrade to upgrade): pyparsing>=1.5.6,!=2.0.0 in /Library/Frameworks/Python.framework/Versions/3.4/lib/python3.4/site-packages (from matplotlib==1.4.2)
Requirement already satisfied (use --upgrade to upgrade): python-dateutil in /Library/Frameworks/Python.framework/Versions/3.4/lib/python3.4/site-packages (from matplotlib==1.4.2)
Requirement already satisfied (use --upgrade to upgrade): six>=1.4 in /Library/Frameworks/Python.framework/Versions/3.4/lib/python3.4/site-packages (from matplotlib==1.4.2)
Downloading/unpacking pytz (from matplotlib==1.4.2)
  Downloading pytz-2014.10-py2.py3-none-any.whl (477kB)
Installing collected packages: matplotlib, pytz
Successfully installed matplotlib pytz
Cleaning up...
hotspur2-2:PythonInstallation hugh$
```

Figure 16

To test your *matplotlib* installation, type or paste the following commands into IDLE, one line at a time, exactly as written:–

```
from matplotlib import pyplot
pyplot.plot([1, 2, 3, 4], [1, 4, 9, 16])
pyplot.show()
```

The IDLE window should look something like the following:–



```
*Python 3.4.1 Shell*
Python 3.4.1 (v3.4.1:c0e311e010fc, May 18 2014, 00:54:21)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>> from matplotlib import pyplot
>>>
>>> pyplot.plot([1, 2, 3, 4], [1, 4, 9, 16])
<matplotlib.lines.Line2D object at 0x108f13da0>
>>> pyplot.show()
Ln: 9 Col: 0
```

Figure 17

After you type the **ENTER** key following the last line, the following window should appear:–

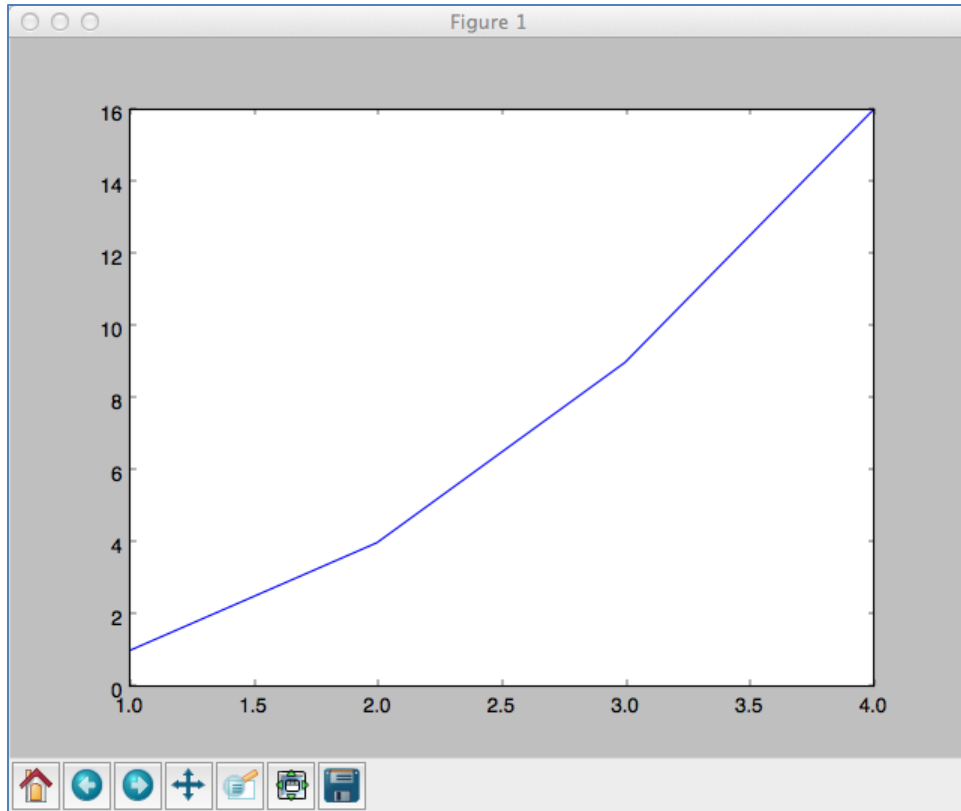


Figure 18

To close this window, click on the “close” button in the upper right.

For a second test with a more interesting plot, copy and paste the following code, line-by-line, into the IDLE window:–

```

from pylab import *

t = arange(0.0, 2.0, 0.01)
s = sin(2*pi*t)
plot(t, s)

xlabel('time (s)')
ylabel('voltage (mV)')
title('About as simple as it gets, folks')
grid(True)
savefig("test.png")
show()

```

This tests not only *matplotlib* but also *numpy* (in the background). The result should be a window like this:–

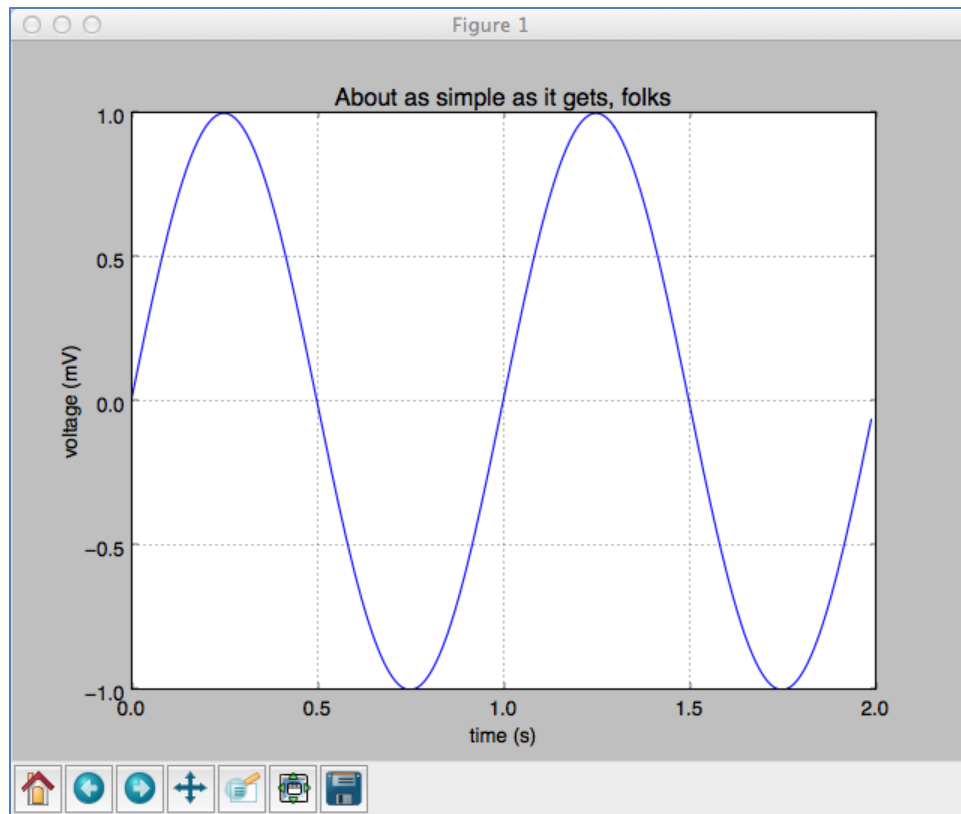


Figure 19

Congratulations! You now have a working version of *matplotlib* installed.

Cleaning up

Remember to go back to the *Security and Privacy* settings *Figure 4* to restore the setting to *Allow apps downloaded from only the Mac App Store and identified developers*.

In addition, note that **dmg** files in Apple parlance are really virtual disk images. Opening the files to install *ActiveTcl 8.5.17.0* and *Python 3.4* causes two disk icons to be left on your desktop. You can remove these by right-clicking and selecting *Eject* for each one.

Installing Graphics.py

To install *graphics.py*, click on this link — [graphics.py](#) — and download the file to the folder where you keep your *Python* programs. Follow the instructions on p.488 of the textbook.

Installing Python 3.4 on Linux

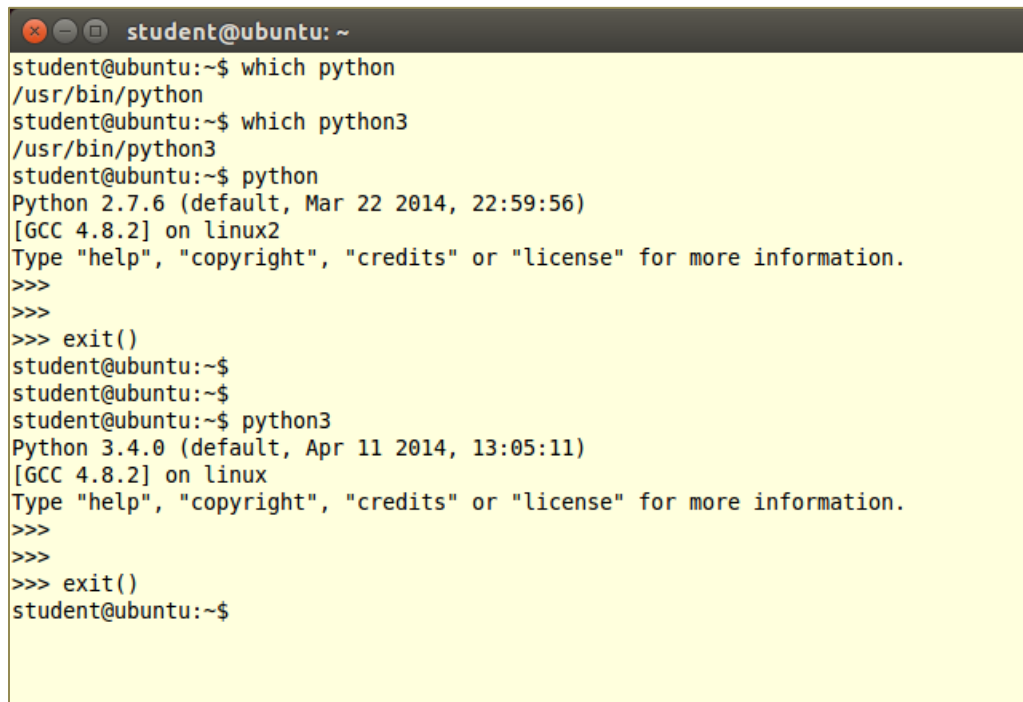
Linux comes in many flavors supported by different organizations with different approaches to software installation. The desktop/laptop versions of Linux typically package the most common applications — including *Python* — in their automated download and installation systems. The following describes the installation on *Ubuntu 14.04*, a popular Linux version in use on the WPI campus, also known by the code name *Trusty Tahr*.^{7,8}

To determine whether or not *Python* is installed in your Linux (for any version or distribution of Linux), open a *Terminal Window* (also called a *Command Shell*) and type the following two commands (on separate lines):—

```
which python
```

```
which python3
```

If either returns the name of an executable file, then execute it to start that version of *Python*. The version number should be the first thing that *Python* says, as illustrated in the following figure for **python** (line 5) and **python3** (line 15):—

A terminal window titled 'student@ubuntu: ~' showing the following commands and their outputs:

```
student@ubuntu:~$ which python
/usr/bin/python
student@ubuntu:~$ which python3
/usr/bin/python3
student@ubuntu:~$ python
Python 2.7.6 (default, Mar 22 2014, 22:59:56)
[GCC 4.8.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>>
>>> exit()
student@ubuntu:~$
student@ubuntu:~$
student@ubuntu:~$ python3
Python 3.4.0 (default, Apr 11 2014, 13:05:11)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>>
>>> exit()
student@ubuntu:~$
```

Figure 20

The first two commands (lines 1 and 3) ask which file contains the programs *python* and *python3*, respectively. In this case, both programs are present in the indicated files. If either had not been present, the command would have reported nothing.

⁷ It is not clear whether up-to-date versions of *Python* are available for download on older releases of *Ubuntu* or other versions of Linux. If you have an older Linux and are not able to upgrade, please see the Professor or Teaching Assistants for advice.

⁸ The WPI Computer Science Department maintains a mirror distribution site for Ubuntu releases at <http://ubuntu.cs.wpi.edu>.

In line 5, the `python` program is run. Its first action is to print a preamble announcing its version and other (possibly useful) information; in this case, it is *Python 2.7.6*. The `exit()` command following the traditional *Python* prompt of “>>>” merely tells *Python 2.7.6* to finish what it is doing and to quit.

In line 14, the `python3` program is run. You can see from its preamble that it is *Python 3.4.0*.⁹

Note: Both Mac OS X and Linux come with Python 2.7 already installed for their own needs. That version of Python must not be deleted or disturbed when installing Python 3.4 and its packages.

Installing IDLE

Figure 20 applies to any version Linux from any distributor to determine whether or not *Python* is installed and, if so, what version it is.

The following is specific to *Ubuntu* Linux. Users of other Linux distributions should carry out similar actions on those distributions. Select and open the *Ubuntu Software Center* icon



in the toolbar on the left of the *Ubuntu* desktop. In the upper right, enter the term “Python3” in the search box. A search should produce the results shown in the figure below.

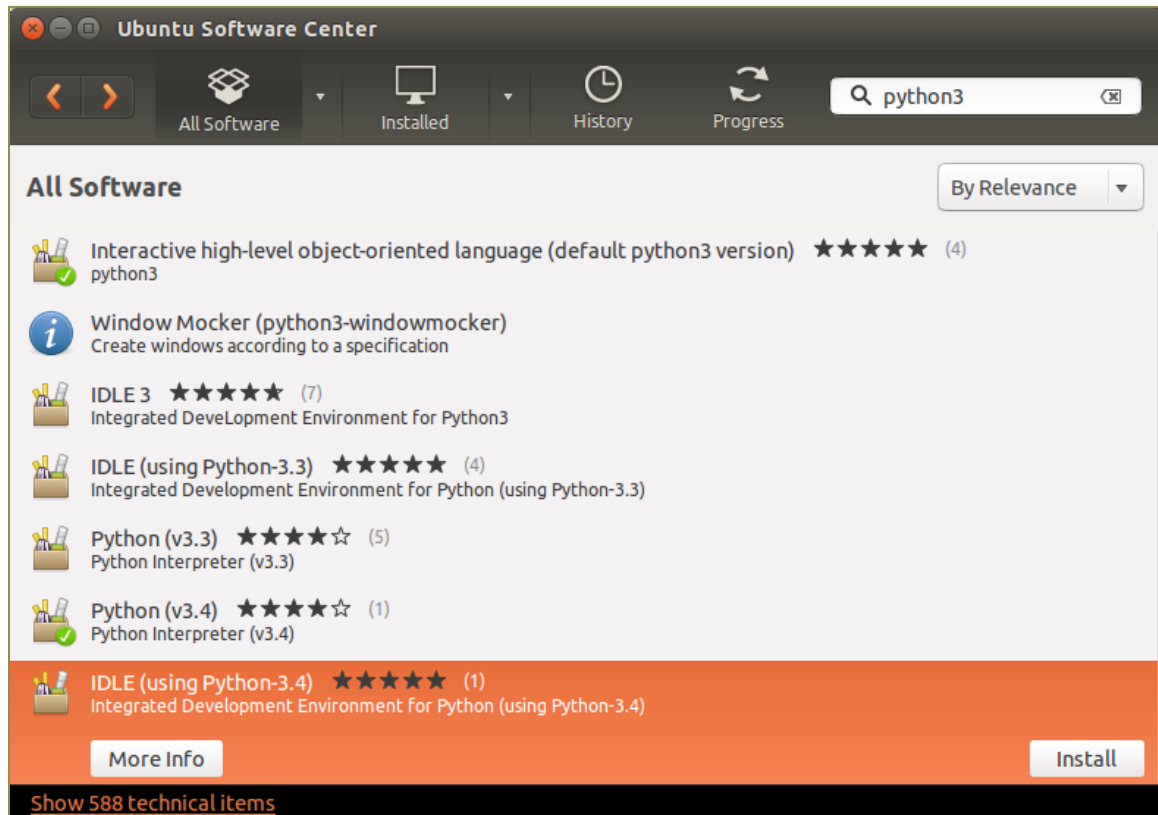


Figure 21

⁹ This is one minor release older than the versions that we have installed on Windows or the Macintosh. However, we will live with that difference for CS-1004.

This shows that the *Ubuntu Software Center* knows about several versions of *Python 3* and several versions of *IDLE*. The last line is highlighted and shows an *Install* button for “IDLE using Python 3.4). Click on this *Install* button to cause IDLE to be downloaded installed.

When the installation has completed, a new icon will have been added to the toolbar on the



left, namely, the IDLE icon. *IDLE* has now been installed.

Testing your Linux installation of Python 3.4

Follow the same instructions as with Figure 10 in order to validate that you can run *Python*.

Installing *numpy* on Linux

In *Ubuntu 14.04*, open the *Ubuntu Software Center* again and search for the term “python3-numpy”. This will bring up one or more results, including the *numpy* package for *Python 3*.

Note: Be sure that you specified “python3” as part of the search term. Otherwise, the search will return information about *numpy* for *Python 2.7*.

Note2: If you use a different version of Linux, the search term in your software installation system should still be “python3-numpy.”

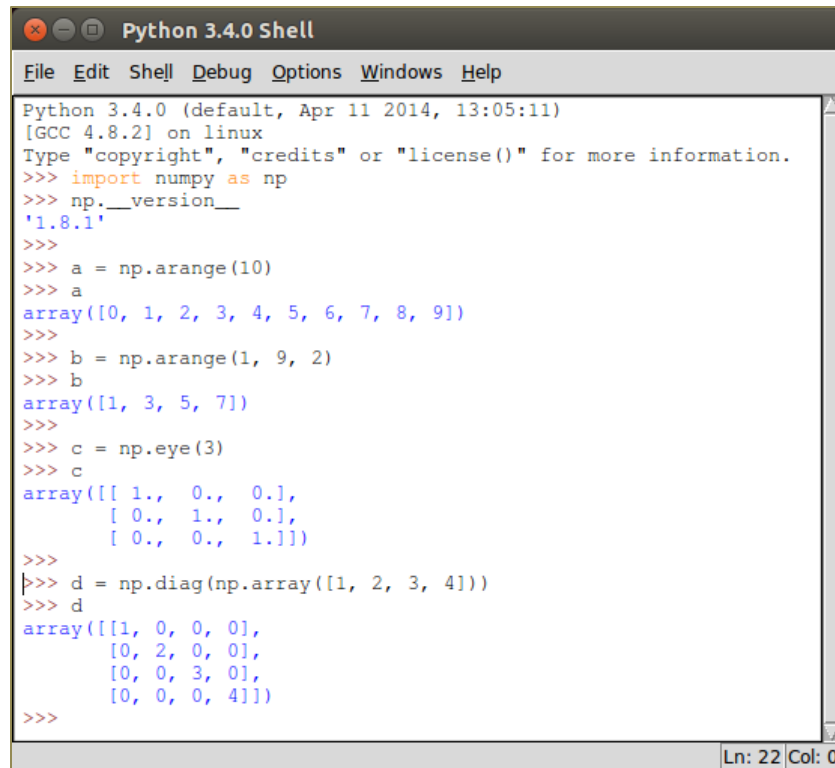
Select on the *More Info* button to bring up the following window:–



Figure 22

Although it does not specify which version of *numpy* is provided, click on the *Install* button to install it. It will automatically install any required dependencies. You do not need to select other options for this course.

Finally, open an *IDLE* window by clicking on the *IDLE* icon and carry out the tests of Figure 15. Results of these tests are shown in Figure 23 below. Fortunately, the version of *numpy* is 1.8.1, as indicated by line 5 of this window.



```
Python 3.4.0 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.0 (default, Apr 11 2014, 13:05:11)
[GCC 4.8.2] on linux
Type "copyright", "credits" or "license()" for more information.
>>> import numpy as np
>>> np.__version__
'1.8.1'
>>>
>>> a = np.arange(10)
>>> a
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
>>>
>>> b = np.arange(1, 9, 2)
>>> b
array([1, 3, 5, 7])
>>>
>>> c = np.eye(3)
>>> c
array([[ 1.,  0.,  0.],
       [ 0.,  1.,  0.],
       [ 0.,  0.,  1.]])
>>>
>>> d = np.diag(np.array([1, 2, 3, 4]))
>>> d
array([[1, 0, 0, 0],
       [0, 2, 0, 0],
       [0, 0, 3, 0],
       [0, 0, 0, 4]])
>>>
```

Figure 23

Installing *matplotlib* on Linux

To install *matplotlib* on your version of Linux, repeat the same steps as you used for *numpy* but search instead for “python3-matplotlib.” In *Ubuntu*, you would see the following:–



Figure 24

Install this, just as you installed *numpy*. To test your installation, carry out the tests of Figure 17 and of Figure 19. This should produce the *IDLE* window of Figure 25 plots shown in those figures

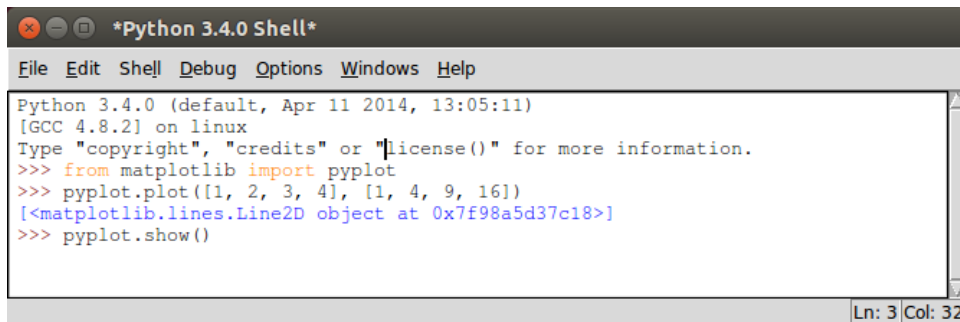


Figure 25

Conclusion

Congratulations! You have successfully used *Python 3.4* as already installed and have added *IDLE*, *numpy*, and *matplotlib*.

If you are brave enough (and savvy enough) to use another distribution of *Linux*, good luck. The guidance here should help you get started. The official *Python* download page is at

<https://www.python.org/download/>

Source “tarballs” are available to compile for your own environment. Alternatively, a list of other *Python* distributors is also provided. Good luck!