PHYS-4007/5007: Computational Physics

Tutorial #1
Using Linux for the First Time

1 Getting Started with Linux.

The information of logging in on the Linux side of the computers in Brown Hall 264 can be found on your **Linux Computer Account Information** sheet passed out in class. Please refer to that sheet to see how to boot to Linux from the Windows machines in Brown Hall 264.

If that terminal icon is not present, follow the instructions on the **Linux Computer Account Information** to start the Linux terminal software. To permanently "pin" the terminal icon so that it always appears on your screen, go to the terminal icon on the left-hand column of icons, right-click your mouse, and select "Lock to Launcher." From now on, every time you log in, the terminal icon will be present on that launcher bar.

As explained on the **Linux Computer Account Information** sheet, one should have made three directories in your login directory using

> mkdir fortran

> mkdir python

> mkdir tex

Note that on the lines above the '>' represents the Linux (referred to as Unix from this point forward) prompt — **do not type this in!** ('>' is used here since each person's prompt is based upon the current directory one is in.) Now do a directory listing using

> ls

this gives a directory listing of the current directory (*i.e.*, "folder" in the Microsoft world). Now enter the following four command in succession

> ls –px

> ls –pxa

> ls –l

> ls –la

The letters after the '–' are called 'flags' in Unix: '–p' appends characters to the file/directory names ('/' for directory, '—' for link, etc.); '–x' lists the entries by lines instead of columns; '–l' shows the long listing of the files which shows what type of file it is, the protections set for the file, it's size in bytes, etc.; and '–a' shows hidden files (those whose file name starts with a dot ('.') as well as normal files). Note that one can include more than one 'flag' keyword after the '–' as shown above. One can get detailed information about any Unix command using the 'man' (for manual) command:

> man ls

See a list of other common Unix commands in Subsection D of Appendix A of the course notes.

2 Other Commonly Used Commands

For a list of common Unix commands see subsection D of Appendix A of the course notes. All common *Unix* commands are in lowercase – remember, Unix distinguishes cases, *e.g.*, ls is not the same as Ls, lS, or LS. We have already used the ls (list source) and the mkdir (make directory) commands, now we will explore some other common Unix commands.

cd (change directory):

The command cd directory means change the current working directory to 'directory'. The current working directory may be thought of as the directory you are in, *i.e.*, your current position in the file-system tree. To change to the directory you have just made, type

> cd fortran

Type ls at the Unix prompt (*i.e.*, >) to see the contents of this subdirectory (which should be empty).

The current directory (.)

In Unix, (.) means the current directory, so typing

> cd .

(NOTE: there is a space between cd and the dot)

means stay where you are (the fortran directory).

This may not seem very useful at first, but using (.) as the name of the current directory will save a lot of typing, as we shall see later in the tutorial.

The parent directory (..)

(..) means the parent of the current directory, so typing

> cd ..

will take you one directory up the hierarchy (back to your login or home directory). Try it now.

NOTE: typing cd with no argument always returns you to your home directory. This is very useful if you are lost in the file system.

pwd (print working directory)

Pathnames enable you to work out where you are in relation to the whole file-system. For example, to find out the absolute pathname of your home-directory, type cd to get back to your home-directory and then type

> pwd

The full pathname will look something like this:

/home/username

which means that 'username' (your home directory) is in the sub-directory 'home' (the group directory), which is in the top-level root directory called " / " . For instance, my home directory is '/home/lutter'.

~ (your home directory)

Home (also called login) directories can also be referred to by the tilde ~ character. It can be used to specify paths starting at your home directory. So typing

> ls ~/fortran

will list the contents of your fortran directory, no matter where you currently are in the file system. What do you think

> ls ~

would list? What do you think

> ls ~/..

would list? Try it and find out.

3 Using the *emacs* Editor

It's a good idea to keep your Unix files organized so they are easily found when you need them. This is why I had you created the fortran, python, and tex directories above. At this point cd (for change directory) into any of the subdirectory (directories below your login directory are called 'subdirectories') that you created, for instance

> cd fortran

Let's now create a Fortran file using emacs:

> emacs mycode.f &

note that the '&' sign put this job in background so that you can still access Linux in the terminal window. There are one of two ways we can input the text presented below. First, we could just go to the Tutorial web page linked to the course web page and click on the first 'mycode.f' link; then select, copy and paste into the file opened in your emacs session (make sure that the Fortran command lines start in column 7 in the file). Or we can just type the lines presented below into the emacs GUI (Graphic User Interface) window (*note:* do not include the first line of the sequence of numbers):

```
123456789T123456...
      PROGRAM MYCODE
C
C This is my first Fortran code.
C
      INTEGER I, J, I_INDEX(100)
      REAL PI, LLCOEFF(100)
      REAL*8  DPI, SSCOEFF(100), BBOUT(100)
C
      PI = 3.14159
```

```
      DPI = 3.14159265359D0
C
C Calculate integer and single precision arrays.
C
      DO 52 I = 1, 100
         I_INDEX(I) = I*2
         LLCOEFF(I) = FLOAT(I**2) * SIN(PI*FLOAT(I)/10.)
  52  CONTINUE
C
C Calculate double precision arrays.  This DO loop has an error.
C
      DO 1005 J = 1, 100
         SSCOEFF(J) = COS(DPI*DFLOAT(J)/8.0D0)
         BBOUT(J) = SSCOEFF(J) * DEXP(-DFLOAT(J)/300.D0)
  52  CONTINUE
C
C Output this data to the screen in ordered columns.
C
C This next DO loop has an error in it.
C
      DO 2344 I = 1, 200
         WRITE(*, 2340) I, I_INDEX(I), LLCOEFF(I), SSCOEFF(I),
     1      BBOUT(I)
C
C This FORMAT statement has two errors in it,
C one minor and one severe.
C
 2340    FORMAT(2X,2I5,2X,F8.4,2X,1PE12.2)
 2344 CONTINUE
C
      STOP
      END
```

After you have finished entering this text, go to the top of the emacs GUI and press the "Save" button. Then go to the 'File' pulldown menu at the top-left of the GUI and quit the emacs GUI.

In the code above note that I purposely included a few programming errors, some you will see when compiling and others when you run the code. You will need to fix these in order for the code to run successfully. We will continue this in our next tutorial.

4 Finishing Up

Next week we will be working with this Fortran file, so mount a USB Flash Drive into one of the USB slots and copy this file to your USB stick once it is mounted. If you are not sure how to do this, Dr. L will assist you. Following this, eject your USB stick and log off of your account.