

MCB 5472 Assignment #1: Introduction to the terminal and Perl

January 22, 2014

Jonathan Klassen (jonathan.klassen@uconn.edu – please put “MCB 5472” in your email title)

The purpose of assignment #1 is for you to use the terminal to perform basic file manipulations and to introduce you to the first very basic functions of Perl (reading and writing files). You are encouraged to use whatever resources you can find, including (but not limited to): the lecture materials, the assigned texts, other texts, on-line resources and your peers and colleagues.

(1) [2 marks]: Read “Unix and Perl Primer for Biologists” sections U1-U28 and do the accompanying exercises.

“Unix and Perl Primer for Biologists” is the designated text for this course and can be found here: http://korflab.ucdavis.edu/Unix_and_Perl/unix_and_perl_v3.1.1.pdf. These chapters will teach you the basic principles of working in LINUX using the terminal and are crucial for you to accomplish all that follows.

The text is written assuming that you use a Mac, but every command will work using any UNIX-derived operating system. (Recall that Mac OS is at its heart UNIX.) Therefore everything in this section will be essential for using the UNIX cluster. **Windows users:** Windows is not UNIX-based and so these commands will not work in this environment (although parallel commands do exist in DOS, if you are old enough to remember). Therefore you have 3 options: (i) use a Mac in the computer lab; (ii) download sshclient and do all of your work on the cluster (<ftp://ftp.uconn.edu/restricted/ssh/>; I also a copy on a USB key); sshclient then functions as your terminal; (iii) use Linux; I recommend installing Biolinux (<http://nebc.nerc.ac.uk/tools/bio-linux>) to either run off of a USB stick or to boot in parallel to your Windows operating system. Biolinux comes with many bioinformatics tools preinstalled and so can save you some headaches. Please talk to me if you choose this option (I have dual boot BioLinux machines for the last 4 years and have been happy with them).

Before the start of next class: Email Jonathan a screenshot (use the “Print Screen” key on your keyboard and paste the picture in some program that makes documents, e.g., Word, Powerpoint) showing the list of files in the directory in which you created the files generated during “Unix and Perl Primer for Biologists” sections U1-U28. You may use any computer and operating system you like to do this.

(2) [2 marks]: Log into the Biotechnology Center cluster.

Before the first class, you should have received an email from Peter with instructions to apply for an account on the Biotechnology Center’s cluster. Having done so, you should have received an email listing your account login information and how to log in using ssh in any terminal and change your password.

Logging on with ssh is half the battle; the other half is transferring files for the server to work with. If you work in Windows and installed sshclient a file transfer program installed alongside the ssh terminal. I recommend FileZilla for Mac and Linux users. As you become more comfortable with the Terminal, you can also use the very powerful “rsync” and “scp” commands.

Before the start of next class: Copy any file that you would like into your home directly on the Biotechnology Center cluster. Use the terminal to list the files in your server root directory and email Jonathan a screenshot of this result.

(3) Introduction to Perl.

This section will have you generate a series of scripts of increasing complexity using some of the basic programming concepts we covered in the introductory lecture. You may use any text editor you like to write these scripts. On the server, I recommend `nano` or `vi`. On a Mac I recommend TextEdit and in LINUX I recommend Gedit (also available for Windows, but this requires you copying your file onto the server each time you make an edit).

Before the start of the next class: Complete its script, run it storing the output in an appropriate output file, and email both your scripts and their corresponding output files to Jonathan. Ensure that all of your scripts function and are appropriately documented. Name your files according to the following convention:

Script:

`[yourname]_[number].pl`

Output file:

`[yourname]_[number].out`

Replace `[yourname]` with your actual name and `[number]` with the exercise number.

(3a) [1 mark]: Write a script that prints “Hello World” to the screen.

(3b) [5 marks]: Write a script that adds, subtracts, multiplies, and divides two numbers of your choice. Print each equation and answer to the screen. Have your script answer the question “is the answer to the multiplication question is greater than 10?” and print this to the screen. Declare the numbers as variables, do not rewrite them multiple times.

(3c) [5 marks]: Use a text editor to create two unique files. Write a script that inputs both of these files and outputs them in a single output file. Send Jonathan both input files labeled: `[yourname]_3c.input1` or `[yourname]_3c.input2`.