CS 250 Lab 1 – Introduction to Linux, the Shell, and Emacs

1. Login to Windows.
2. Click on the Start menu, then find and start the X2Go client.
3. We are going to use the Windows system to connect to another computer system: the computer science department's server named "ice". A session is a collection of settings that allow you to connect to another computer in a specific way. Once the X2Go client starts, perform the following steps to create a new session that will allow you to quickly connect to ice. You only have to create a session once on each computer that you use. After that, you can use the previously created session anytime you wish to connect.
	1. From the menu bar, select Session > New Session
	2. Under the Session tab, set the following text fields. Pay attention to case.
		1. Session name: Ice
		2. Host: ice.truman.edu
		3. Login: <your TSU.edu login name>
		4. Session type: XFCE
	3. Click the OK button.
4. The new session you just created appears on the left of the X2Go window. Click on the session to start the login process.
5. Your username should already be filled in using the information you provided in step 3). Type your password in the textbox and click OK. If you get a dialog box telling you that the server is unknown, and asking if you want to trust it, click "Yes". In a few moments, the Linux desktop should appear in a new window. Click on the maximize button of the "ice" window.
6. The Linux system may ask you some initial configuration questions. It will only ask these once. If you are asked to choose a password for a "new keyring", click "cancel". If you are asked to configure panels, choose the default.
7. You can interact with the Linux GUI from the Applications menu at the top left of the screen. You should feel free to look explore the Linux GUI later. In this class, we will not be using the GUI to interact with Linux except in a few circumstances. Instead, we will be learning the Linux command line.
8. Access the command line by finding the icon representing the terminal at the bottom of the screen. Click on the icon to start a terminal session. When the terminal starts, it should look something like this:



1. When you are finished with your Linux session, **do not click to close box on the X2Go window. It is very important that you log off of the Linux system properly when you are done**. If you do not log off of the Linux system before you leave, you can corrupt your Linux settings, which may make it impossible for you to login again without having the systems administrator reset your settings. Simply closing the X2Go window is equivalent to turning off the power off on your desktop system instead of shutting it down.

To log off of Linux, go to the Applications menu and choose Log Out. When the "Log Out" menu appears, click the appropriate panel.

1. Once the Linux window disappears, quit X2Go from the File menu. You have now properly logged off of ice.
2. We will now start to learn the Linux command line. Open X2Go again, login to ice, and open a terminal if one does not appear automatically.
3. Today we will be learning some of the basic commands that are required to successfully work in the Linux command line environment. The program that you interact with in the terminal window is called a "shell". The particular shell we will be using is called "bash". (Bash is am acronym for "Bourne Again Shell".) As we learn new commands, you will be asked to answer some questions about the commands. Write your answers in the space provided.
4. As in most other operating systems, the Unix file system is organized into a tree structure. The very top of the tree is called the root. The root is denoted by the symbol '/'. Folders in Linux are called "directories" (dirs, for short).

We can name a specific directory or file anywhere in the computer by listing the set of folders encountered as we traverse the file system from the root to the dir or file. This listing is called the directory or file's "pathname". In a pathname, each folder level in the file system tree is separated from the one above it by the '/' symbol.

Each user on a Linux system has a directory that belongs just to them, called their home directory. When you first log in, your home directory is also your "working directory". You can change your working directory at any time, but your home directory never changes.

You can find your current working directory using the "pwd" (print working directory) command. Type the command pwd in the terminal window, and then hit the <enter> key. What is the pathname of your current working directory?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. The "ls" command is used to list the files in the working directory. Ls will show regular files in white, and subdirectories in purple. Execute the ls command. What subdirectories exist in your home directory?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. You can make your own subdirectories using the "mkdir" command. Type "mkdir CS250". Use the ls command to verify that you created a subdirectory named "CS250".
2. The "cd" command is used to change your working directory. Type "cd CS250" to change your working directory to CS250. Use the pwd command to verify that CS250 is your current working directory.
3. When navigating through the file system, the shell provides a couple of shortcuts to help you move around. The shortcut ".." always refers to the directory *above* the current working directory. The shortcut "." always refers to the current working directory. Type "cd ..", and then verify that you are back in your home directory.
4. Change your working directory back to the CS250 subdirectory.
5. To start an application from the shell, you type the application's name. Start the web browser "firefox". Once Firefox opens, make the terminal window the foreground window by clicking on it.
6. Notice that when you run an application, the shell waits for the application to finish before it allows you to run another command. Quit Firefox. Notice that only after the application has ended does the shell print another prompt, indicating that it is ready for you to type the next command.

Often, we want an application to start and "run in the background", allowing us to continue to interact with the shell while the application runs. You can tell the shell to launch an application and then run the application in the background by following the application name with an ampersand when you start it. Type "firefox &" in the shell. Firefox will start up, and the shell should immediately print a prompt indicating that you can type the next command.

1. The editor that we will be using this semester is called "emacs". We are using emacs because it seamlessly integrates many of the Linux developer tools like the compiler (gcc), the build manager (make), and the debugger (gdb) that we will be using this semester. There are other editors installed on most Linux systems, and you should feel free to experiment these outside of class. However, you need to develop some skill with emacs, because it is the de-facto programmer's editor on Linux.
2. Start emacs in the background by typing "emacs &". You should see a window that looks like this:



1. A few emacs commands are available from its mene system, but most advanced commands in emacs are issued by typing special key combinations, using either the "meta" key, or a <ctrl> key sequence. In Linux, the <esc> key is the <meta> key. Click on the blue text labeled "Emacs Tutorial" and work through the tutorial. If you aren't finished by the time that class ends, make a note of your stopping place, so you can continue on your own before the next class meeting. To exit emacs, choose quit from the File menu, or use the key sequence <ctrl>-x <ctrl>-c.
2. You will probably want to access the Linux server from your own machine. If you have a Windows or Macintosh machine, you can download a copy of X2Go for your system from wiki.x2go.org.