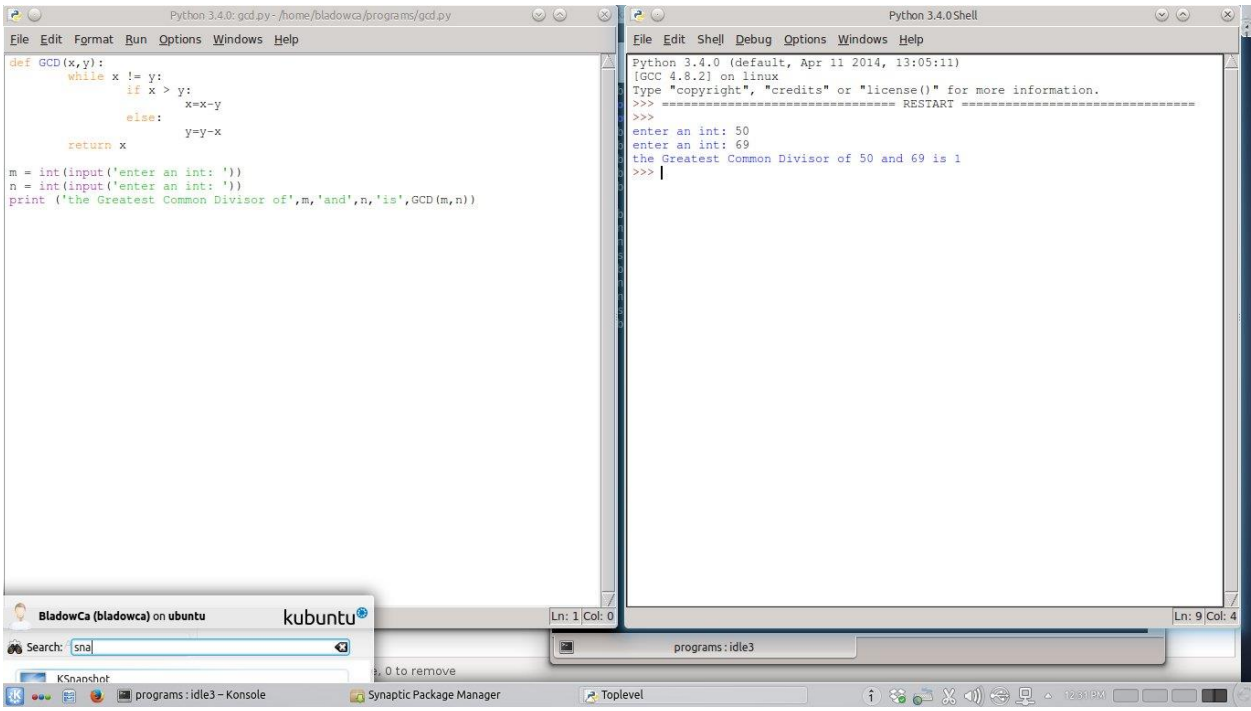
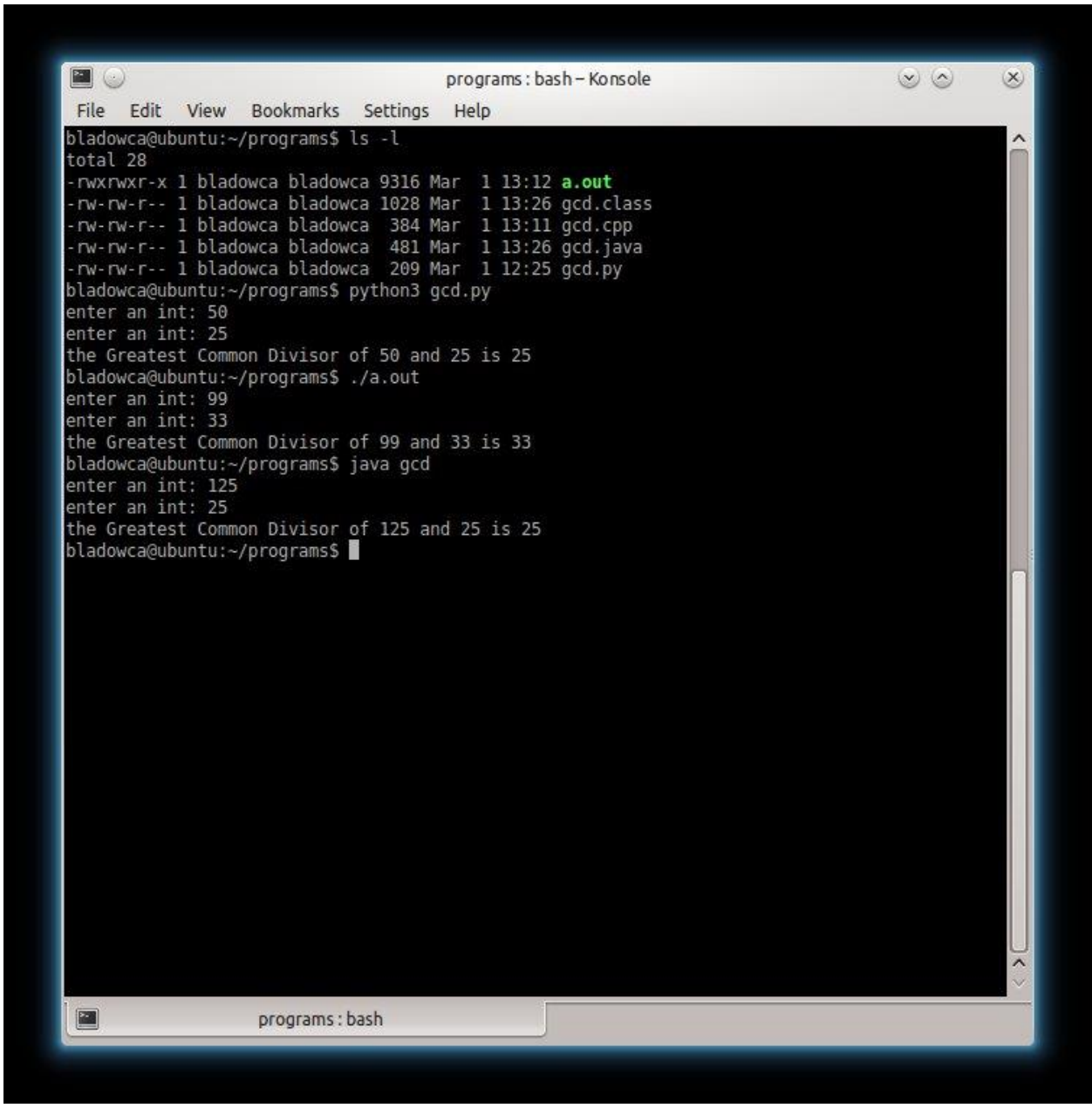
Casey Bladow
CSIS 360 - Spring 2015
Assignment #6 - 25 points
Due Tuesday, March 3

1. Let's install some programming language support for some programming languages used in the CSIS Department. To get you to try different methods of installing packages, I want you to use *Synaptic*for any installations in this step.
    - Create a directory called *programs* and you will write 3 computer programs and save them in this directory. You can make a directory from a terminal session with the command *mkdir programs*then change to the directory with the command *cd programs*. You are on your honor to use the *vim* editor to create each of these programs so that you continue to get practice with *vim*
    - Using *Synaptic*, check to see that *python3* is installed by searching for it. I believe it was installed by default when you installed Ubuntu. If it isn't, install it using *Synaptic*.
    - Using the *vim* editor, type in this **python program** and save it with the name *gcd.py*. Note the ~ characters displayed in the image are how *vim* displays non-existent lines (but I guess you already knew that from assignment 5). Run the program with the command *python3 gcd.py*
    - Using *Synaptic*, search for *idle3*. Mark it for installation then install it. This will install the Integrated DeveLopment Environment for python.
    - From the command line in the *programs* directory, type the command *idle3 gcd.py* then press *F5* to run the program. After entering the input and the program has run, ***take a screen shot of the idle3 shell and include it in your assignment 6 document.***



- Using *Synaptic* search for *build-essential*. Mark it for installation then install it. This will install the *g++* compiler for C++ programs.
- Using the *vim* editor, type in this **C++ program** and save it with the name *gcd.cpp* in your *programs* directory. Compile and link the program with the command *g++ gcd.cpp* then run it with the command *./a.out* (dot slash a.out).
- Using *Synaptic* search for *openjdk-7-jdk*. Mark it for installation then install it. This will install the *javac* compiler for Java programs and the *java* bytecode interpreter.
- Using the *vim* editor, type in this **Java program** and save it with the name *gcd.java* in your *programs* directory. (Note: this was my first Java program!) Compile the program with the command*javac gcd.java* then run it with the command *java gcd*
- Once all the programs are working, type the following commands from within a terminal window while in the *programs* directory.
  - *clear* (to clear the screen)
  - *ls -l* (to display the contents of the programs directory)
  - *python3 gcd.py* (to run the python program)
  - *./a.out* (to run the C++ program)
  - *java gcd* (to run the Java program)

After you've entered all the commands, ***take a screen shot of the terminal window and include it in your assignment 6 document***

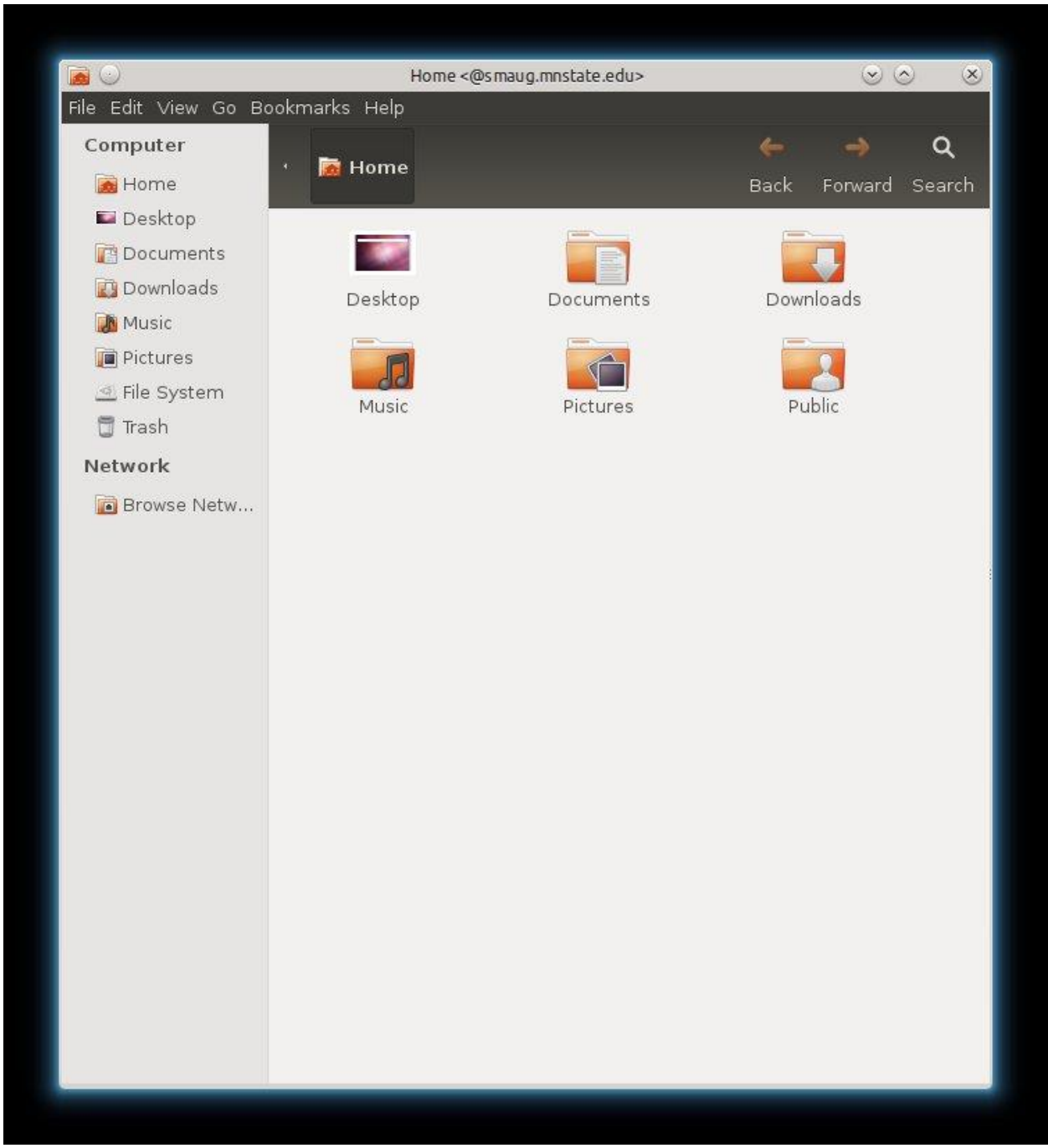


- I also was planning on having you install the latest version of *Eclipse* using **http://ubuntuhandbook.org/index.php/2014/06/install-latest-eclipse-ubuntu-14-04/** but I ran into a known issue**https://bugs.launchpad.net/ubuntu/+source/gtk2-engines-oxygen/+bug/1408614** (and I've tried changing themes... many times). I was able to successfully install *Eclipse* in one of my many Ubuntu installs and also a second time from a fresh Ubuntu install but at this time I'm unable to help you if you run into the issue I've had. Therefore, I'm not requiring this. If you want to try using the above article, it will take you only about 5 minutes. If you run into the same issue and you

can help me with a workaround on this problem, please let me know. I haven't tried, but I suspect installing an old version of *Eclipse* from the Ubuntu repositories would work just fine. You can always install the latest and greatest, but if it isn't in the Ubuntu repositories, there's no guarantee it's going to work. As such, I would encourage you to attend the ACM Workshop Tuesday, Feb 24th at 5:00PM in Bridges 161 if you would like to get *Eclipse* installed.

2. *kate* is a popular file editor used by many CSIS students when creating files on *smaug*. *kate* should have been installed when your installed the *KDE* desktop environment in assignment 3. Check to see that *kate* is installed. If not, install it. If you don't see the option to display a terminal at the bottom of *kate*, click on *Settings* > *Configure Kate...* > *Plugins*, then check the box for *Terminal tool view*. There is no verification required for this step as I'm assuming you would have motivation to have the *kate* editor.
3. This step is optional, so therefore requires no verification. If you want to connect to the *puff.mnstate.edu* web server outside of campus, you need to log into the MSUM VPN. After a couple of hours trying to do it, it finally dawned on me that if you're using a virtual machine, you connect to the VPN outside your virtual machine and there you have it. So if you notice a knob on my forehead, that's from me banging my head on the table. If you are not using a virtual machine (Ubuntu is on a separate partition or Ubuntu is the only thing on your hard drive), you can use **THESE** directions if you want. If you give it a try, please e-mail me and let me know if you were successful or not, or if you have a better set of directions.
4. The next part of this assignment deals with using OpenSSH as a client to access other computers. This is described in chapters 18 and 19.
    - Read chapter 18 pages 663-676 on OpenSSH clients while at the keyboard experimenting. Reading the remainder of the chapter is optional as that deals with setting up an OpenSSH server. The*JumpStart* section is especially important to use as a client. You can browse through the configuration discussions as everything should be set up and ready for you to use.
    - Read chapter 19 pages 687-698 on the details of *sftp* while at the keyboard experimenting. Reading the remainder of the chapter is optional as that deals with setting up an FTP server. Keep in mind that you should be using *sftp* and not *ftp* as the chapter points out (the commands are the same). Pages 695 (starting with ftp Specifics) through 698 are especially important for the commands.
5. Follow these steps for verification of learning the material in these chapters. Take care to not get things mixed up.

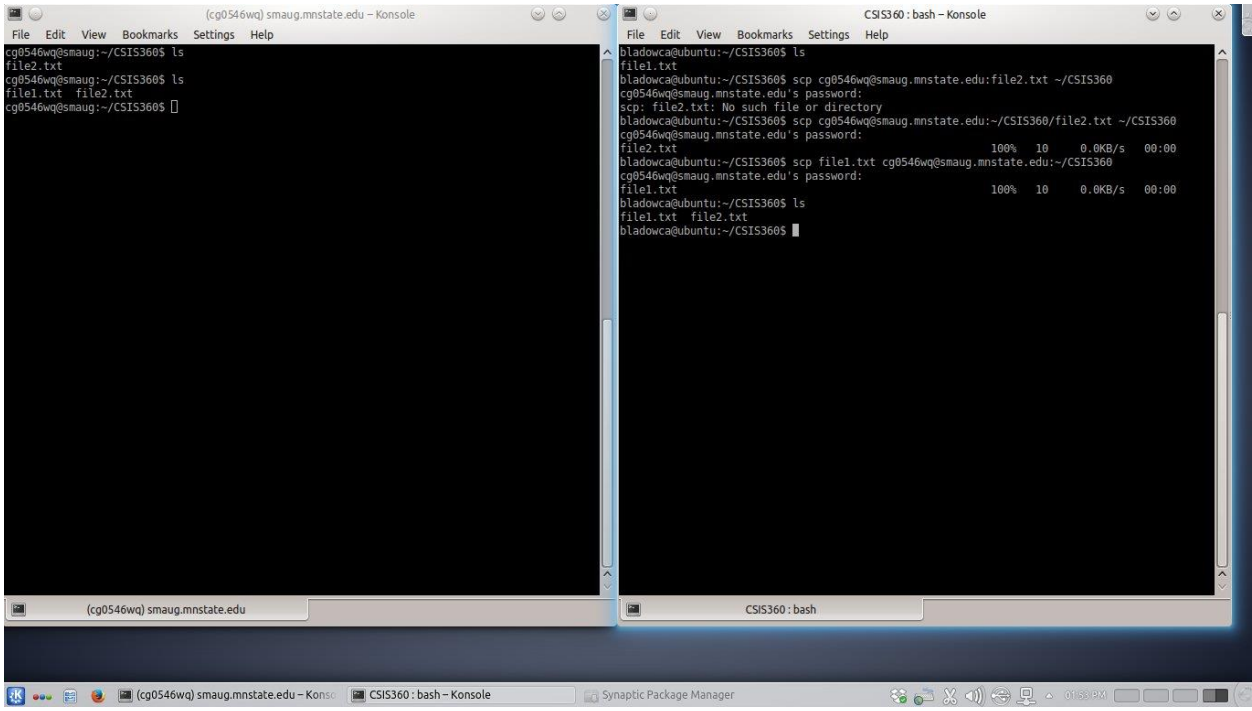
- Open a terminal window on your machine and make a directory CSIS360 (or something close to that if you happen to already have that directory) then change to it. Using *vim* create the file*file1*.*txt* that contains anything you want (the contents of the file is irrelevant to this assignment).
- Open a terminal window on your machine and log in to your *smaug* account using *ssh*. The command
  ssh -X yourStarID@smaug.mnstate.edu
  will do this.
- From your smaug account, type the command *xeyes*. After you close it, type the command *nautilus* and ***do a screen grab of the nautilus window and include it in your assignment 6 document***.



Then go ahead and close the nautilus file manager.

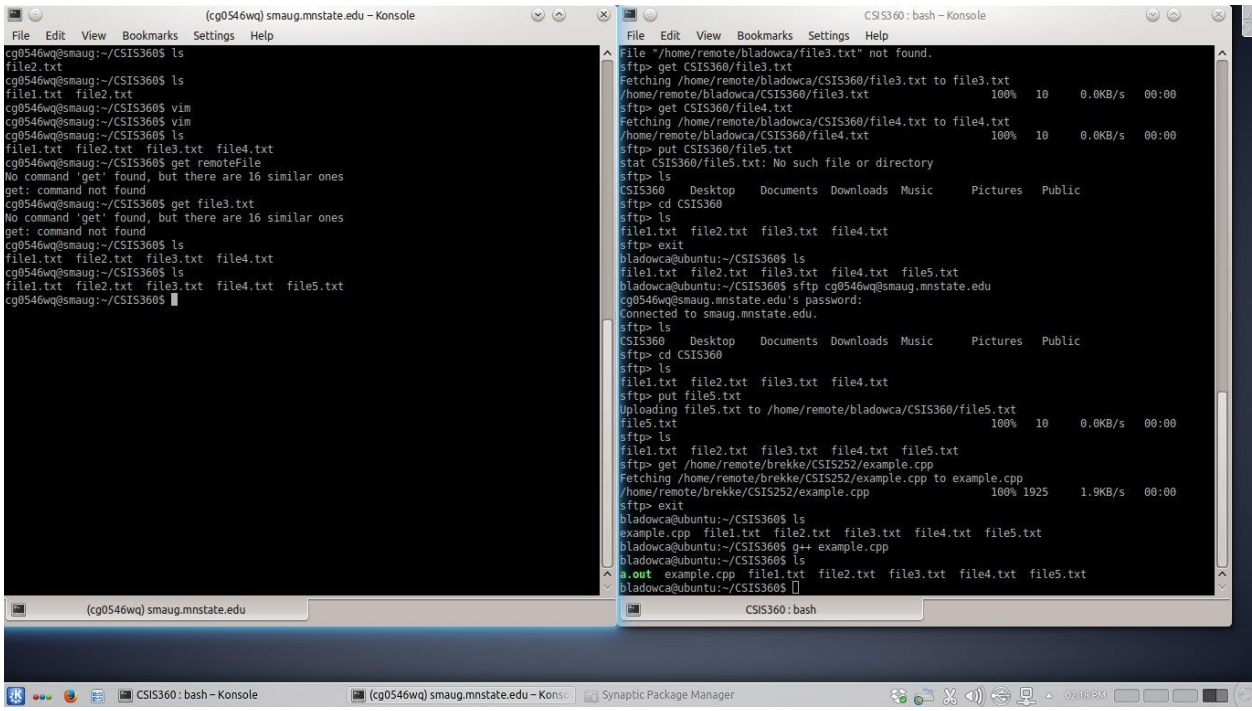
- From your smaug account, create a directory CSIS360 (or something close to that if you happen to already have that directory) then change to it. Using *vim* create the file *file2.txt* that contains anything you want.
- From the terminal window on your Ubuntu box, issue the *scp* commands that will copy the file *file1.txt* on your local machine to your CSIS360 directory on smaug and the file *file2.txt* from your CSIS360 directory on

smaug to your local machine's CSIS360 directory. Type the command *ls* on both terminal windows and ***do a screen grab of the two terminal windows and include it in your assignment 6 document***.



The terminal window of your Ubuntu box should show the *scp* commands that did the file copies.

- From your smaug account in the CSIS360 directory, create two more files named *file3.txt* and *file4.txt* using the *vim* editor. These files can contain whatever you wish. On your Ubuntu box, use the *vim* editor to create a file named *file5.txt* in your CSIS360 directory. The file can contain whatever you want.
- From a terminal window on your Ubuntu box, use *sftp* to download the files *file3.txt* and *file4.txt* to your Ubuntu box CSIS360 directory and upload the file *file5.txt* to your CSIS360 directory on smaug. Next download the file */home/remote/brekke/CSIS252/example.cpp* to your CSIS360 directory on your Ubuntu box. Quit *sftp* and compile and link *example.cpp* to produce the executable*a.out*. Type the command *ls* on both terminal windows and ***do a screen grab of the two terminal windows and include it in your assignment 6 document***.



The terminal window of your Ubuntu box should show the *sftp* commands. Your smaug terminal window should have 5 files and your Ubuntu box terminal window should have 7 files.

6. ***Answer the following questions in your assignment 6 document*** No guessing like I've seen in previous assignments when such questions are asked!
    - What's the purpose of using the -X option when using *ssh*?

      SSH X forwarding – allows you to forward programs that use a GUI through your remote SSH connection to a CS machine.

    - What's the difference between *scp* and *sftp*?

      scp can only be used for transferring files, and it is non-interactive. sftp is more elaborate, and allows interactive commands to do things like creating directories, deleting directories and files, etc.

    - What single command issued from a terminal window on your Ubuntu box will list the files in your CSIS360 directory on smaug?

      ssh –X cg0546wq@smaug.mnstate.edu ls ~/CSIS360

    - What is the *mput* command using *sftp*?

      put multiple files on the remote machine

- What *sftp* command will allow you to see what remote directory you are in?

  pwd

- What *sftp* will allow you to see what local directory you are in?

  lpwd

- What command allows you to change your local directory using *sftp*?

  lcd

- How can you execute a shell command when you are using *sftp*? What does executing a shell command mean?

  ! <command>

  It means you are running a shell script. Shell scripts are commands that then execute one by one.

- What's the difference between binary and ASCII transfer mode in *sftp*?

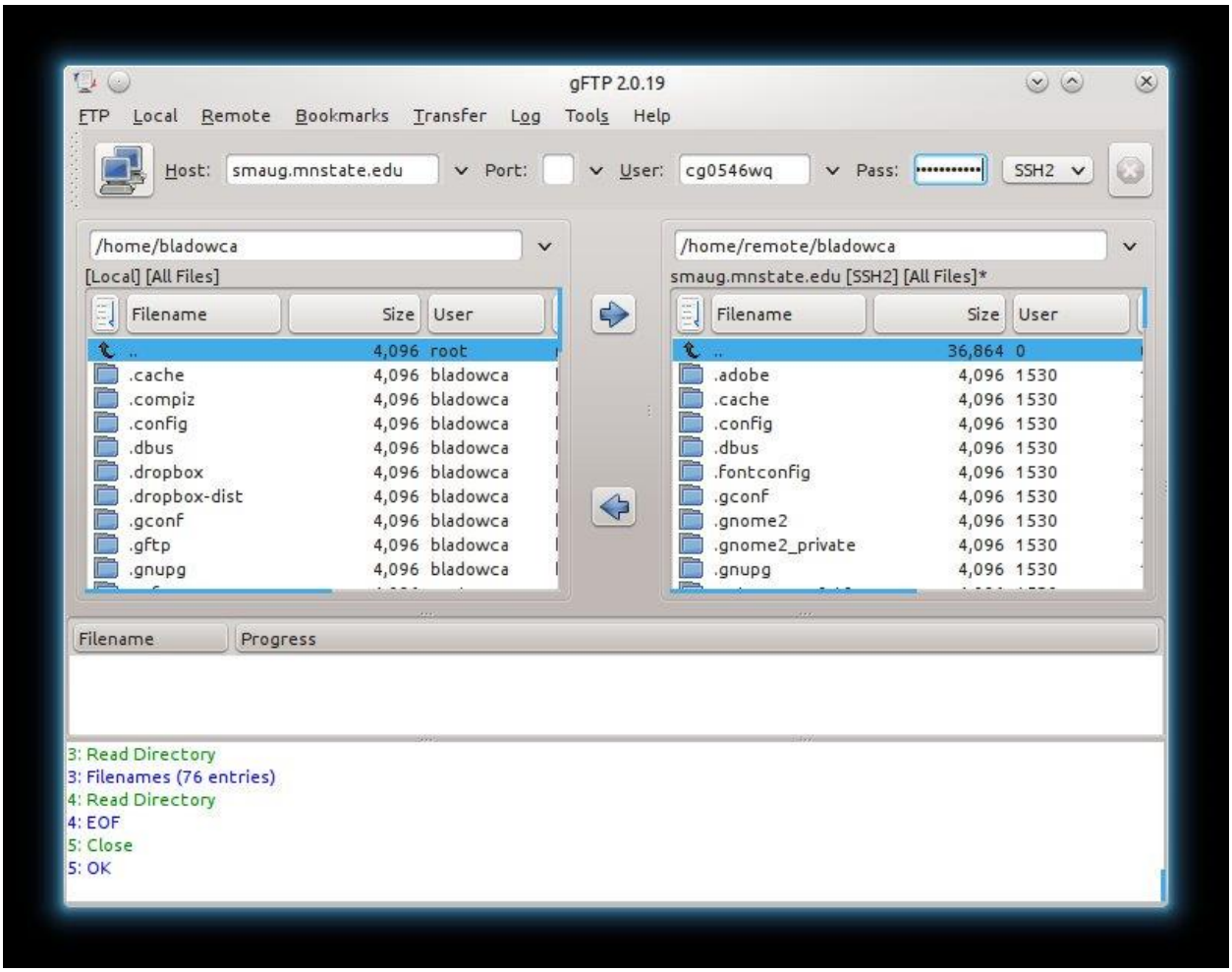
  ASCII will ‘convert’ text to the appropriate EOL/EOF (etc.) for the receiving OS.

  Binary gives you an exact copy, regardless of the OS.

- What's the *sftp* command that will download your entire CSIS360 directory (including the directory structure) from smaug? Note: I couldn't find it in the chapter so the answer is to use the -r option (recursive) as in *mget -r CSIS360*. Use this as your answer (fair game in the first test!).

  mget –r CSIS360

7. Install *gftp*. Use whatever method you wish for installation. Log into your smaug account using SSH2 as the protocol. After you have logged in, ***take a screen shot of the gftp window and include it in your assignment 6 document.***



Experiment with *gftp* if you wish as it's pretty straight forward.

8. ***What did you think of this assignment? How long did it take you? Do you have any suggestions for the next time I teach this class?***

   It was neat. Definitely shows vim's ability to code in multiple languages. Think I spent 3-4 hours on it, had lunch and a few other things going on. Took a little break as well. I wouldn't change anything with this one. Some of the instructions had to be read with extra care so you knew which Terminal window you were working in, but it was fine.

9. ***What did you learn from this assignment?***

   Transferring files around via the command line. I've always used GUI methods to do this so having an understanding of how to do it without a GUI available never hurts.