Introduction to COMP 206
Lab A

The labs, for this course, are designed to be completed on your own at home or in the 3rd floor Trottier labs. These labs are not graded. You do not hand in these labs. If you prefer to work on a lab with your TA tutorial group, then check the schedule for your TA's tutorial session. You will find this schedule in our MyCourses page under Content/Course Information/TA Information. Since the university has limited lab space, your TA might ask you to bring your laptop and work in a classroom instead of a lab.

This lab is about getting used to the School of Computer Science servers and the Linux command line.

Some labs will have a question zero. These questions will not be covered by the TA during the tutorial. It is extra content meant for you to do on your own.

At the end of this lab is a Mentoring section. The TA will use this during their tutorial section.

QUESTION ZERO: Optional problem

(A) Let us learn a little about the Linux Manifesto.

Google "Linux Documentation Project Manifesto". Summarize the manifesto in two to four sentences.

(B) Popularity of development environments

There are two common environments that run software: the server and the personal device. Developers need to know where their software will be deployed in order to properly construct their algorithm so that it is sensitive to the deployed environment. Servers have different requirements compared to personal computers. In this course we will develop software for both personal computers and servers.

Look at the following two links and determine the popularity of Windows vs Apple vs Linux for (A) used as a personal computer and (B) used as a server.

The two links are: https://www.w3schools.com/browsers/browsers_os.asp and https://en.wikipedia.org/wiki/Usage_share_of_operating_systems

QUESTION ONE: Getting and using your CS user account

All assignments will be graded on the mimi.cs.mcgill.ca server. This means that the assignments you submit must be able to run on this machine. It is important that you establish the following procedure when submitting assignments:

1. Use the mimi.cs.mcgill.ca server for all assignments. Only this server will be accepted for assignments. The TAs access this server for grading. You can ssh into this machine or go to Trottier 3rd floor to work on this machine, to create your assignments. All computers on the third floor of Trottier can be used. You can also remotely connect using iOS **ssh** (at the command line of your PC) or using Windows **putty** (download the app for your PC).
   a. HOWEVER! You cannot log into this server using your first.last@mail.mcgill.ca account because that will redirect you to the **McGill** server and not to **mimi.cs.mcgill.ca** even though you logged in using a computer in Trottier or logged in from home by typing `ssh mimi.cs.mcgill.ca  !!`
   b. You must first create a computer science user account by doing the following:
      i. Make sure you are on campus (or using a VPN)
      ii. Open your browser and type: https://newuser.cs.mcgill.ca and follow the instructions.
      iii. If you wish to reset your password: https://newpassword.cs.mcgill.ca
2. Now try to login.
   a. If you are at home and using a Mac, go to the command line (launchpad then Terminal or search Terminal) and type `ssh mimi.cs.mcgill.ca`, you should see a prompt asking you for your CS user name and password. If all goes well, you will then see the welcome message and a prompt from the server. Or, you can type `ssh username@mimi.cs.mcgill.ca`, it will only prompt you for the password. (Some iOS users will need to do it this way because your iOS will assume your laptop's username, by default, when logging into the SOCS server ~ strange but true).
   b. If you are in the Trottier lab, you will see a GUI login screen. Enter your CS user name and password. If all goes well, you will see the desktop. Find the command line (terminal) icon and double click it. You will now see a similar interface as the Mac students using ssh: a welcome message and a prompt.
   c. If you are home with a Windows computer, download the **putty** app, since ssh is not installed on most Windows computers. Download the open source program named Putty. Just use Google to find it. Then use the GUI to input your username, password, and the URL mimi.cs.mcgill.ca. Then press the login or connect button. If all goes well, you should see the command line and welcome message.
3. Now, let us logout
   a. If you are at home using a Mac, then type `logout` and your connection with the server ends. Then click on the X to close the Terminal window.
   b. If you are in Trottier and in the command line window then type `exit` to close the window or press the X to close the window. Then you will need to logout of the desktop. On the right top of the screen (usually) is the logout menu option. Select it.
   c. If you are at home using Windows and Putty, then press X to close the Putty program.
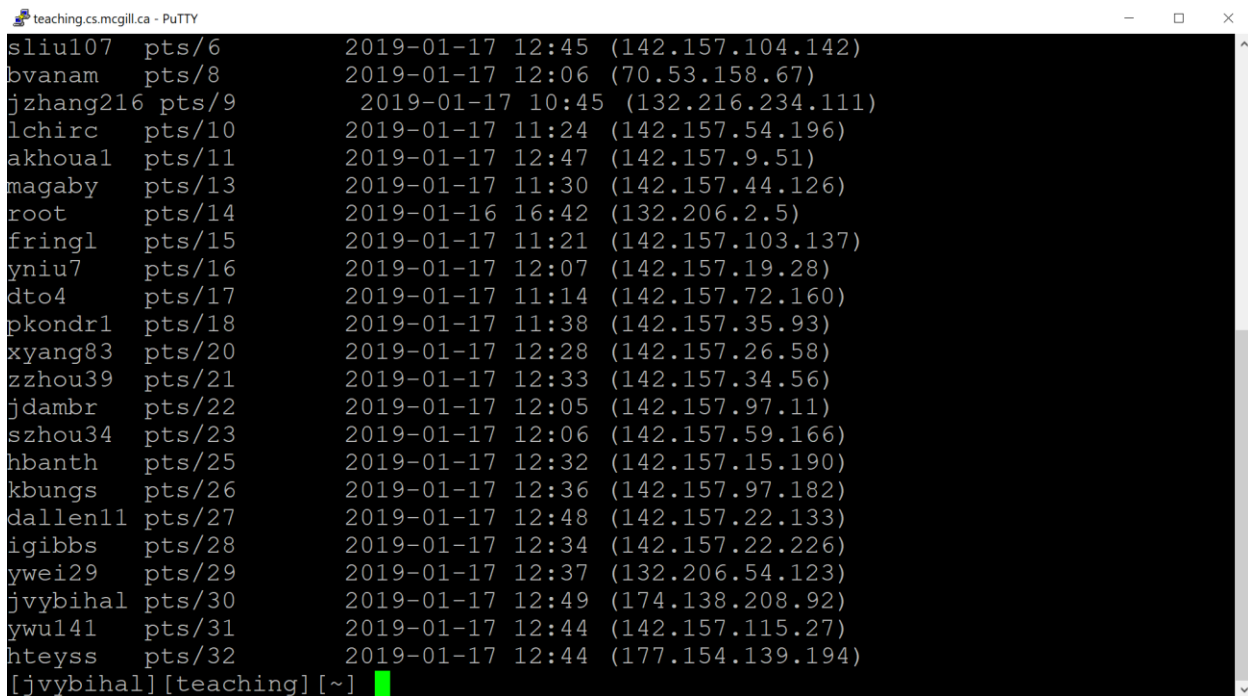
QUESTION TWO: Backing up Files

Try this experiment:

(A) Login to **mimi.cs.mcgill.ca**
(B) In your home directory `mkdir` a directory called **LabA**
(C) Then `cd` into that directory and `mkdir` a subdirectory called **backup**
(D) Using `cp` copy a random file from your home directory into LabA renaming it to **letter1.txt**. Select any file from your home directory to copy into the LabA subdirectory. Do not worry that you are changing the file's name within the LabA subdirectory. It will not affect the original file.
(E) Using `cp` copy another random file into LabA renaming it to **letter2.txt**.
(F) Using `touch` create a third file called **letter3.txt** in LabA.
(G) Look at your files by typing `ls` and `cat.`
(H) Using TAR archive your three text files into the directory backup. You will need to write something like this: `tar -cvf backup/myletters.tar *.txt.` This command takes all your text files (*.txt) and merges them into an archive file called myletters.tar within the subdirectory backup.
(I) Using `ls`, look at the directory backup to see if your myletters.tar file exists. If it does, then you did it correctly.
(J) Now, `cd` into the archive directory and extract the files. Check the class notes on how to do this. The files you archived should be extracted within the same directory (the **backup** directory). Once extracted they should appear within the directory.
(K) If something unexpected occurred while archiving / extracting, discuss this with your classmates or the TA.

QUESTION THREE: Interacting with other users

There are many ways to interact with users on a server. Let us look at one interesting way:

(A) Login to mimi.cs.mcgill.ca
(B) Find out the users currently logged in by typing the command: `who`
(C) Use the `grep` command to find someone in the list of active users. You will need to type
     something like this: `who | grep 'abc'`
     Where 'abc' is the string or substring of the username you want to find.
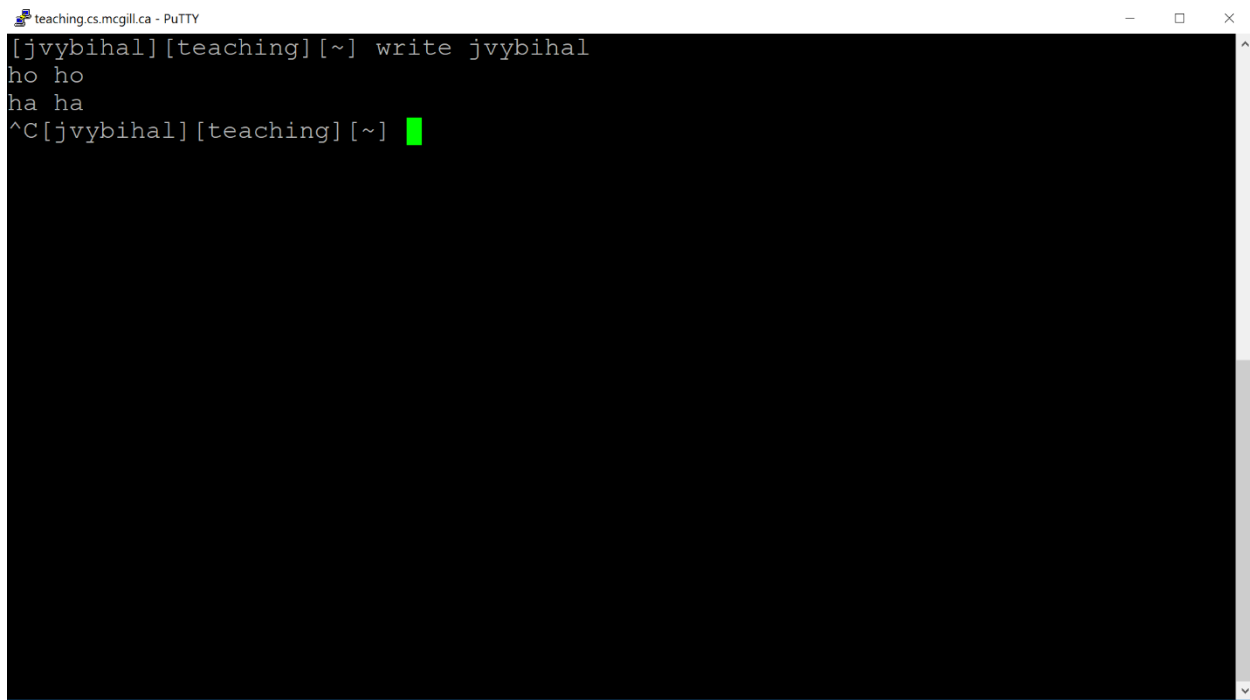(D) Notice that the command `who` displays something like this:

```
teaching.cs.mcgill.ca - PuTTY                                          —    □    ×
sliu107   pts/6          2019-01-17 12:45 (142.157.104.142)
bvanam    pts/8          2019-01-17 12:06 (70.53.158.67)
jzhang216 pts/9           2019-01-17 10:45 (132.216.234.111)
lchirc    pts/10         2019-01-17 11:24 (142.157.54.196)
akhoua1   pts/11         2019-01-17 12:47 (142.157.9.51)
magaby    pts/13         2019-01-17 11:30 (142.157.44.126)
root      pts/14         2019-01-16 16:42 (132.206.2.5)
fringl    pts/15         2019-01-17 11:21 (142.157.103.137)
yniu7     pts/16         2019-01-17 12:07 (142.157.19.28)
dto4      pts/17         2019-01-17 11:14 (142.157.72.160)
pkondr1   pts/18         2019-01-17 11:38 (142.157.35.93)
xyang83   pts/20         2019-01-17 12:28 (142.157.26.58)
zzhou39   pts/21         2019-01-17 12:33 (142.157.34.56)
jdambr    pts/22         2019-01-17 12:05 (142.157.97.11)
szhou34   pts/23         2019-01-17 12:06 (142.157.59.166)
hbanth    pts/25         2019-01-17 12:32 (142.157.15.190)
kbungs    pts/26         2019-01-17 12:36 (142.157.97.182)
dallen11  pts/27         2019-01-17 12:48 (142.157.22.133)
igibbs    pts/28         2019-01-17 12:34 (142.157.22.226)
ywei29    pts/29         2019-01-17 12:37 (132.206.54.123)
jvybihal  pts/30         2019-01-17 12:49 (174.138.208.92)
ywu141    pts/31         2019-01-17 12:44 (142.157.115.27)
hteyss    pts/32         2019-01-17 12:44 (177.154.139.194)
[jvybihal][teaching][~]
```

A fun activity it to communicate with other students who are currently logged in. One way to do that is
to use the `write` command. You can `man write` to find more information. The basic syntax is the
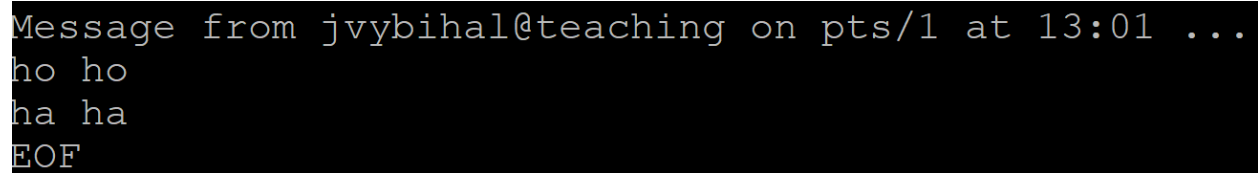following:

`write USERNAME`

Where USERNAME is the user name that is displayed in the first column of the `who` output. Then you
can type sentences to that user, press enter, and type more. All that will be displayed on your friend's
screen. To end writing to your friend, press `control-c`. Be careful, this also works for random users,
not only your friends.

For example, I logged in twice and spoke to myself:

```
teaching.cs.mcgill.ca - PuTTY                                            —    □    ×
[jvybihal][teaching][~] write jvybihal
ho ho
ha ha
^C[jvybihal][teaching][~] █
```

On the other account I saw:

```
Message from jvybihal@teaching on pts/1 at 13:01 ...
ho ho
ha ha
EOF
```

To block people from doing this, use the command: `mesg n`

You can let people write to you again with this command: `mesg y`

The meaning of the command is obvious: Messaging NO, Messaging YES. Have fun annoying people.

You have completed the lab.

MENTORING SECTION

The TA will review the following things during their tutorial:

- Review
  - Begin this tutorial by making sure everyone has an account and can log into mimi.cs.mcgill.ca
  - Give a demonstration of how to transfer files from mimi to a laptop and back again
  - Review how to use grep from the command line
- Lab
  - Go step by step through the command line portion of the lab with the students