Lab 1: Unix Commands

Due. Tuesday, June 4th, 2019

In this lab, you will work with a partner to practice some Unix commands.

What To Do

Part 1: Unix commands

You will explore a directory called **Lab1a** using some Unix commands. If you do not remember which Unix command to use, you can look at the [slides on Canvas](). You can also use the man pages, trial and error (but be careful), and your favorite search engine to complete the lab.

Step 1. Find a partner. You will complete this lab with a partner to practice pair-programming and to ge through the exercises more quickly.

Step 2. For this lab, you should be using some flavor of Linux or MacOS. If you do not have Linux or Mac system you can use the CU's online coding environment, [here]().

Step 3. Pick your favorite text editor. Unix systems feature a lot of different text editors, such as pico, emacs, and vim. Choose any of them to complete the exercises.

Create a file called "lab1a.txt". Write your and your partner's name at the top of the file. You will record each of the commands you used during this activity in this file.
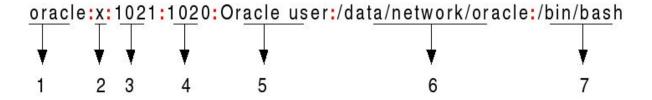
Step 5. Complete the following exercise and record all of the commands:

1. Make a directory named **cs3308**.
2. Rename the directory **cs3308** to **CSCI3308**. Enter the directory.
3. Download the **Lab1a** folder [here]() and place it in this **CSCI3308** directory. You can also download the compressed version [here]() and unzip it.
4. List the contents of **Lab1a** directory. Are there any hidden files? What command did you use to display them?
5. List the files in the **pictures** directory, sorted according to their size, listing one file per line. Which file is the biggest one and what command did you use to find this?
6. Create a new directory called **bigData** and copy both the largest file from the **pictures** directory and the largest file from the **texts** directory into it. How many bytes is this directory and what commands did you use?

7. Use a Unix command to create a zipped file called "bigData.zip" that contains files currently in **bigData**. How many bytes is this file?
8. Use a Unix command to create a tar file called "bigData.tar" that contains the files currently in **bigData**. How many bytes is this file?
9. Delete the **bigData** directory and the file "bigData.zip". What commands did you use?
10. Extract the contents in the tar file "bigData.tar". What options did you use? What does each do?
11. What is the difference between "little1.html" and "little2.html"?
12. Use the find command to find the file called "target.txt". What is the path from the **Lab1a** directory? What's its full path on your machine?
13. Finally, use the commands head and tail to create a new short story made up of the first five lines of one of the books in the **texts** directory and the last five lines of a different book in the **texts** directory. Write down the commands you used.

Part 2: (More) advanced Unix commands

The /etc/passwd file contains one entry per line for each user (or user account) of the system. All fields are separated by a colon (:) symbol. There are seven fields as follows:

```
oracle:x:1021:1020:Oracle user:/data/network/oracle:/bin/bash
       1   2   3        4             5                    6            7
```

1. **Username:** It is used when user logs in. It should be between 1 and 32 characters in length.
2. **Password:** An x character indicates that encrypted password is stored in /etc/shadow file. Please note that you need to use the passwd command to computes the hash of a password typed at the CLI or to store/update the hash of the password in /etc/shadow file.
3. **User ID (UID):** Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.
4. **Group ID (GID):** The primary group ID (stored in /etc/group file)
5. **User ID Info:** The comment field. It allow you to add extra information about the users such as user's full name, phone number etc. This field use by finger command.
6. **Home directory:** The absolute path to the directory the user will be in when they log in. If this directory does not exists then users directory becomes /
7. **Command/shell:** The absolute path of a command or shell (/bin/bash). Typically, this is a shell. Please note that it does not have to be a shell.

Source:

You will use a sample /etc/passwd to practice some more advanced Unix commands.

Step 1. Make a copy of the file /etc/passwd in the **Lab1a** directory with the name "passwdDemo.txt".

Step 2. Using the "passwordDemo.txt" file as input, figure out the commands to complete the exercise below. Record your commands in your "lab1a.txt" file. (HINT: Some useful commands you may need: `grep, sort, chmod`)

1. Sort the file based on the username.
2. Sort the file based on UserID. Since this is a number, be sure to specify that it is a number (e.g., use a command option) so it sorts the numbers correctly.
3. Show all the lines in the file with the name daemon in it.
4. Get the number of lines in the file using a Unix command.
5. Print only those lines in the file with a unique GID.
6. Use your command for the previous question and now redirect the output to a file named "tmp". Use the > operator.
7. Write the Unix command you used in the previous question into a script file named "getUniqueGID.sh". What is the command to run this script file? Did you encounter a "permission denied" error? Change the permissions of the file to allow users to execute the file. What is the command used to change file access permissions? Did the script run this time?

What To Turn In

You will submit your lab1a.txt file to Canvas.