CSCD 240
Lab 3

All answers are to be produced by using ssh to log into a remote lab machine. You will use **ssh** to login to **cslinux.eastern.ewu.edu to complete this assignment. All commands requiring output capture should be executed using ssh** so all commands are executed on the remote machine and the output is captured on the local machine. In **all cases** you should capture the prompt, the command, and the output from the command.

1) Clearly explain the difference between **which, whereis, grep,** and **find**.

2) Issue the find command looking for the file named **ld** starting at the root directory.
   a. Assuming you are not logged in as root, you should get a list of errors as well as where the file was found. Capture the output and include it in your submission – you do not need to include all the permission errors just a few to get the idea but do include where the file was found.
   b. Repeat the command (again not as root) – illustrating a method of eliminating the error messages and printing only what was found.

3) In class we talked about the '–name' option for the **find** command.
   a. Explain how to use the size option.
   b. Issue and capture the results of the find command in your home directory that display all files that are greater than 1K. Do not search for more than 3 subfolders. Do not display error messages.

4) Use a text editor on the remote machine to create a file named frost.poem that contains the following text:

   The Road Not Taken by Robert Frost
   Two roads diverged in a yellow wood,
   And sorry I could not travel both
   and be one traveler, long I stood
   And looked down one as far as I could
   To where it bent in the undergrowth;

   Then took the other, as just as fair,
   And having perhaps the better claim
   Because it was grassy and wanted wear,
   Though as for that the passing there
   Had worn them really about the same,

   a. Use the **grep** command, capture both the command and the output, to finds all lines, including the line number, that end with a comma.
   b. Use the **grep** command, capture both the command and the output, to finds all lines, including the line number, containing the word **as**.
   c. Use the **grep** command, capture both the command and the output, to finds all lines, including the line number that starts with the word and (case DOES NOT matter).
   d. Use the **grep** command, capture both the command and the output, to finds all lines, including the line number that starts with the word and (case DOES matter).

5) Capture, creating a directory named **lab3**.
   a. Capture placing a copy of frost.poem in the directory **lab3**. There should be one copy of frost.poem in your home directory and one in **lab3**.
   b. Within your home directory, capture the **grep** command and its output that will ***recursively*** find all instances of the word I (case DOES matter) in all files that end with .poem.

6) Execute ***man source***.
   a. Did the manual page display?
   b. What command do you need to issue to get source to display?
   c. Capture the results of that command.
   d. Where could have you used this in relation to lab 1?

7) Using a text editor create a file named **myScript** that contains the following:

```
#!/bin/bash
string=”Hello World”
echo $string
```

   a. Execute the script with ./myScript and capture the output.
   b. Execute and capture the command that will change the permissions on myScript to be user executable without changing any other permissions.
   c. Execute the script with ./myScript and capture the output.

8) Using a text editor create a file named **secondScript** that contains the following:

```
#!/bin/ksh
string=”Hello World”
print $string
```

   a. Execute the script with ./secondScript and capture the output.
   b. Execute and capture the command that will change the permissions on secondScript to be user executable without changing any other permissions.
   c. Execute the script with ./secondScript and capture the output.
   d. What does the #! mean?
   e. In problem 7 what shell did the code execute in?
   f. In problem 8 what shell did the code execute in?

9) Using the **man** page for **env**
   a. Describe (in your own words not with captures from the man page) the output of **env** command with no arguments.
   b. Capture a command other than pwd that will show your current working directory.
   c. Describe what you would have to do to make this change permanent for all future sessions.

10) What is the difference between a shell variable and an environment variable in the bash shell?

11) Define what a process is and what a job is, clearly explain how jobs differ from processes.

12) Give the grep command that will start in your home directory and show the file names and line numbers containing the term "stdio" in all .c files in the home directory and all directories below the home.

13) Consider the following command ls – al | more.

   a. How many processes are created with that command?
   b. What exactly does "|" do in this command?

14) Using the man page for **ps**
   a. Issue and capture the **ps** command with the appropriate options to allow listing of all processes in the system.
   b. Using the output from part A, what was the first process started and by whom was it started?
   c. What was the first non-root process that was started?
   d. What was the last process started and by whom?

Terminate your **SSH** session with the exit command. For the next set of questions complete them on your local box or in the lab. NOTE: you **CAN'T** complete them via SSH.

15) In a single terminal window capture the command to start **firefox**, **gedit**, **gcalctool** and **gnomine** as background jobs:
   a. What are the job numbers of the above?
   b. What are the process ID numbers of the above?
   c. Capture the command and output to bring gedit to the foreground.
   d. Capture the command(s) to send gedit to the background.
   e. Capture the command(s) to kill gcalctoolone using its job number.
   f. Capture the command(s) to kill gnomine using its process number.
   g. Can CTRL C be used to kill any job? Why or why not?

16) Using a text editor create a file named hello.c that contains:

```
#include <stdio.h>

int main()
{
      printf("Hello\n");
      return 0;
}// end main
```

a. Capture the command that compiles hello.c and leaves the executable program in a file named hello.
b. Execute hello and capture its output.
c. Copy the contents of hello.c into a file named hello2 (no .c suffix)
d. Try to compile hello2 into a file named hello_2 and capture the output.
e. Explain the output from Part D.

17) Using a text editor create the following C programmed named almostEndless.c

```
#include <stdio.h>

int main()
{
      int x = 0;
      while(x < 20000000)
      {
            printf("..");
            fflush(stdout);
            sleep(3);

      }// end while

      return 0;
}// end main
```

a. Compile your program with gcc almostEndless.c
b. Start your program with ./a.out
c. With your program running, describe the commands you would use (without using ctrl-c) to terminate that program, from the same terminal window in which it was started
d. Execute and capture the commands, using process notation, to terminate a.out
e. Restart your program with ./a.out
f. Execute and capture the commands, using job notation, to terminate a.out

TO TURN IN:

- A PDF file - Name this text file your last name, first letter of your first name lab3.pdf. This file will contain all your answers. I want the question copied and then the answer to the question below it.
- A zip file that contains your PDF, all files created during this lab (C files, scripts, etc).

You zip will be named your last name first letter of your first name lab3.zip (example steinerslab3.zip)