ECE 282 Lab 7

Lab Due: 11:00pm April 2nd , 2019

READ CHAPTER 6

Austin Sypolt

# Command practice (include in report)

What do the following commands do?

1. kill  
   Terminates a process. Sends a specific signal to specified process or process group. If there is no signal specified the ‘TERM’ signal is sent, this will kill processes which do not catch this signal.
2. kill -9  
   All processes with a process identification number (PID) larger than 9 will be signaled.
3. pkill  
   Look up or signal processed based on attributes of the intended process.
4. top  
   Displays Linux tasks
5. top –u ‘root’ –d 0.5 [replace ‘root’ with your username]  
   top –u asypolt –d 0.5

-u Monitor only processes with an effective UID or user name matching the given one.

-d Specifies the delay between screen updates (ss.tt), so our 0.5 has half a second delay.

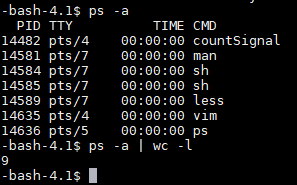
This command looks at only the processes with ‘asypolt’ as their UID, refreshing the screen with a delay of 0.5 seconds.

1. who | tr ‘a’ ‘A’  
   tr allows for the translation or deletion of characters. In this example the who page is displayed with all ‘a’ replaced by ‘A’.

My UID asypolt changes to Asypolt for example.

is ‘tr’ a user program or a utility tool? Why?  
tr is a utility tool because it’s designed to optimize the system through its functionality

Write down the commands that can do the following tasks:

1. Count **all** the processes that are currently running on your system  
   **Hint: ps -a**  
   

There are 8 processes running, 1 less than the output due to the first line in the ps –a command showing the category titles.

1. How use ‘kill’ to send an SIGINT to a process?  
   **Hint: man 7 signal. Find the Value for SIGINT.**kill –s 2

Same as an interrupt from the keyboard (Ctrl + C).

1. **Bonus:** List ONLY the hidden files in your current directory  
   **Hint: ls and grep.**  
   ls –a | grep ‘^\.’

# Why you keep interrupting me?

6.8 Are you sure? Modify the sigdemol. c program so that it asks if the user really meant to

kill the program. A sample run should look like the following:

hello

hello

Interrupted! OK to quit ( y /n) ? n

hello

hello

Interrupted! OK to quit (y/n)? y

$

What happens if the user presses Ctrl-C when the program is waiting for an answer to the

OK to quit (yin)? question? Write the code and see what happens.

# Why you keep interrupting m?

6.7 Modify the sigdemol. c program so it counts the number of times the user presses Ctrl-C.

The revised program will print the message OOCH!, then OOCH! ! , where the number of exclamation

points equals the number of times the handler has been called.

In addition to printing an increasing number of exclamation points, the program should

accept an integer as a command-line argument. After the user presses Ctrl-C that many

times, the program should exit.

# Play again and again (Bonus %10)

1. Compare the 5 versions of play\_again from the course companion website.
2. Modify play\_again3.c so that it does not show or respond to any keyboard input except ‘y’ or ‘n’.  
   Notice that play\_again\* programs are designed to work along with atm.sh bash script.

Submit the report and all the codes to Canvas.