Margaret Connor
P1 Report
Spring 2019

Process.cpp
In process.cpp I've created a nested series of fork calls to successfully perform the "ps -A | grep argv[1] | wc -l" command. The first fork in the parent creates the child, the second fork occurs in the child and creates the grandchild, and the last fork occurs in the grandchild and creates the grandchild. Each process executes one command or waits, using pipes and dup2 to send outputs to other processes.

Here is an example output of my code compared to the command.

```
[23:36:30] connorbg@uw1-320-04: ~/U/CSS430/p1 $ g++ processes.cpp
[23:37:23] connorbg@uw1-320-04: ~/U/CSS430/p1 $ ./a.out root
0
[23:37:29] connorbg@uw1-320-04: ~/U/CSS430/p1 $ g++ -o processes
processes.cpp
[23:37:58] connorbg@uw1-320-04: ~/U/CSS430/p1 $ ./processes kworker
44
[23:38:01] connorbg@uw1-320-04: ~/U/CSS430/p1 $ ps -A | grep kworker |
wc -l
44
[23:38:07] connorbg@uw1-320-04: ~/U/CSS430/p1 $ ./processes sshd
5
[23:38:14] connorbg@uw1-320-04: ~/U/CSS430/p1 $ ps -A | grep sshd | wc
-l
5
[23:38:18] connorbg@uw1-320-04: ~/U/CSS430/p1 $ ./processes scsi
12
[23:38:24] connorbg@uw1-320-04: ~/U/CSS430/p1 $ ps -A | grep scsi | wc
-l
12
```

Shell.java
Shell.java acts like a command interpreted in ThreadOS, the shell takes in a command and converts it to a string. The shell.java searches the string for where next '&' or ';' is found. If the delimiter '&' is found my program will run that command and move on to find the next delimiter, if ';' is found the program will run the command and wait until the command is complete. In the case that no delimiter is found and there still exist character in the string the program will execute the remaining characters as a command.

Given the string "PingPong abc 100 & PingPong xyz 100 ; PingPong 123 50" the program will stop at char 18 and select the substring 0 to 16 to run as a command, then it moves on focusing only on the remaining string which is now "PingPong xyz 100 ; PingPong 123 50."

I had an issue in which the ThreadOS would ask for commands twice and would always post the exception

Exception in thread "Thread-0" java.lang.IllegalThreadStateException
    at java.lang.Thread.start(Thread.java:708)
    at Scheduler.run(Scheduler.java:168)

I worked for hours trying to get this issue resolved, and after the turn in data spoke to a classmate about the issue. I ended up redownloading all the ThreadOS files and the issue went away. I made some changes to the code and resubmitted it.