Lab 4 - Shell Scripting

Testing

A combination of the -rss, -comm, -command, and -group flags were executed with the following two lines of code in a terminal for each test to compare the system output to the product of our shell program:

ps -eo pid,user > t1system.txt
./ps.sh > t1lab.txt

The order of the output on our command input is always be pid, user, group, rss, and comm or command. The flags only specify if the output is present or not.

The first test involved using no flag arguments, exhibiting only the PID and UID of each process. This test was successful since the output of both t1system.txt and t1lab.txt matched apart from obvious spacing differences. The second test only specified the -rss flag, as can be seen with the populated resident set size column for each process in t2system.txt and t2lab.txt. The third test used the long -command flag, correctly printing each process command in both t3system.txt and t3lab.txt. The fourth test attempted to execute the lines with -comm and -command flags, but printed an error message to the file indicating that both flags cannot be used at once. This is an expected response to our input since only one of -comm or -command can be specified. The next two tests use the -group with -rss and the -group with -comm. The final test uses all three flags (-group, -rss, and –comm) as seen in t7system.txt and t7lab.txt.

There were no notable differences in output between the system and final shell program in each test case. However, there is a small disparity between the two outputs (system and lab) under the command name for the last listed process in each test. The final entry in the lab output files specify the extension (ps.*sh*), whereas the system output only includes the name without the extension (ps). This discrepancy exists as the system version invokes its own version of the ps command and thus does not want to invoke our ps.sh version.