

Sean Skinner
CptS 370
Program 1 Ideas

First thing I need to do to replicate the commands using system calls is to fork my process multiple times. There needs to be 3 forks to create 3 child processes, child, gchild, ggchild. The ggchild needs to execute the `ps -A` command, then send its results to the gchild process to execute the `grep` command with `argv[1]` as a parameter. Finally the gchild command need to send its results to child to run the `wc -l` command.

In order to execute these commands I will use `execlp()` to execute these commands, then I will have to send the executed results to the other processes by using `dup2()` and pipes. However, because we are sending data between 3 processes we need two pipes, one to connect ggchild and gchild, and the other to connect gchild and child. Then we use `dup2()` where the first argument is the pipe file descriptor `pipe[0]` for read `pipe[1]` for write, and the second argument will be either `STDOUT_FILENO` for writes or `STDIN_FILENO` for reads. We need to use `dup2()` once in ggchild to send data, we need to use `dup2()` twice in gchild to receive data from ggchild and send data to child, and finally we need to use `dup2()` once in child to receive the data. In addition, each process needs to wait for others in order to correctly execute for example parent waits for child, child waits for gchild, gchild waits for ggchild.

Finally we have to be sure to close all file descriptors so arguments will not continue to look for more arguments.

Sample Outputs:

```
[sskinner@sig2 java]$ [sskinner@sig2 java]$ g++ -o processes processes.cpp
[sskinner@sig2 java]$ ./processes kworker
[sskinner@sig2 java]$ 29
ps -A | grep kworker | wc -l
29
[sskinner@sig2 java]$ ./processes sshd
[sskinner@sig2 java]$ 5
ps -A | grep sshd | wc -l
5
[sskinner@sig2 java]$ ./processes scsi
[sskinner@sig2 java]$ 7
ps -A | grep scsi | wc -l
7
[sskinner@sig2 java]$
```