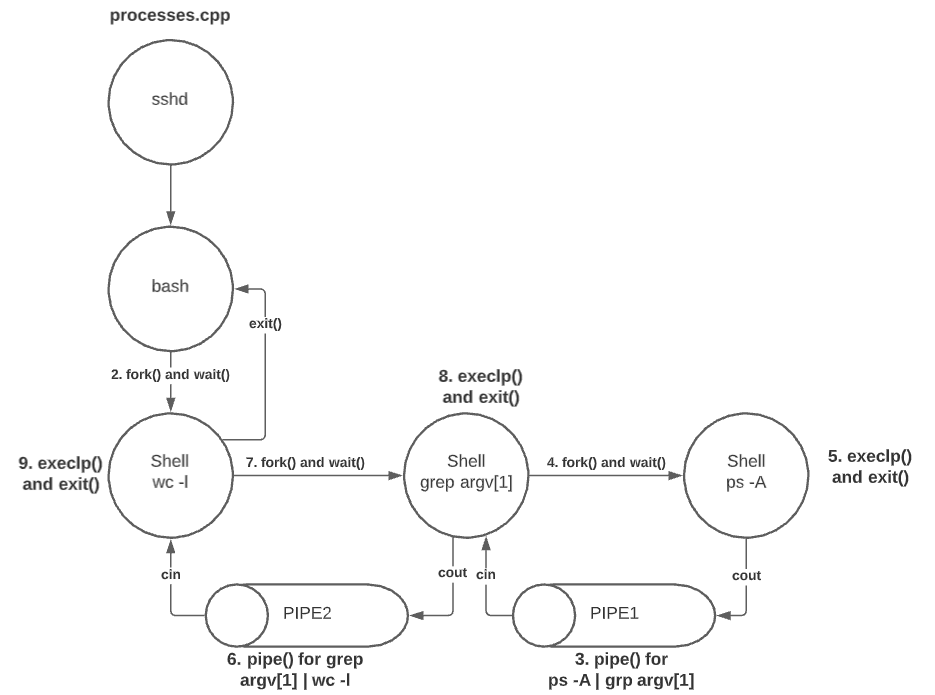
Jessica Nguyen

CSS 430, Project1C

processes.cpp

1. **DESIGN**
2. Flowchart: <https://drive.google.com/file/d/1DG3bdQRAtSNBO_unopjczU2v_rHNB5XH/view?usp=sharing>



1. Components:

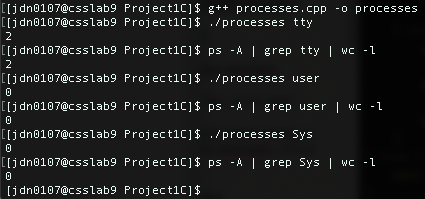
* 1 parent -- 3 nested child processes, 2 pipes
* **Parent** (bash) > **child** (wc -l) > **grandchild** (grep) > **great** **grandchild** (ps -A)
* In order to connect ps -A and grep argv[1], I made one pipe with file directory FD1
* In order to connect grep argv[1] and wc -l, made another pipe with file directory FD2

**Pseudocode**:

1. fork() parent into child
2. Make pipe() out of FD1
3. fork() child into grandchild
4. Make pipe() out of FD2
5. fork() grandchild into great grandchild
6. Execute **execlp() “ps -A”** in great grandchild, return
7. Grandchild finished waiting, execute **execlp() “grep \_\_\_”**, return
8. Child finished waiting, execute **execlp() “wc -l”**, return
9. Parent finished waiting, return
10. **HOW TO TEST PROCESSES.CPP**

* I used “cout <<” statements to keep track of where I am in the program, and I noticed that after calling dup2() to redirect the outputs of File Directories, the cout statements don’t show up on my terminal anymore. It helped me prove that there was a pipe that stored the output instead of my stdout
* I also put in test scenarios like empty string and extra parameters to see if the program would stop running
* I then tried to comment out all the close() methods to see which ones I need in order to make the pipe flow to the correct output. The program stalls if I don’t close one of them

1. **SCREENSHOT OUTPUT**

****

**Additional tests:**

**1 ./process** kworker

**2 ./process** bash

**3 ./process** sshd

**4 ./process** (empty arguments)

**5 ./process a b c** (extra arguments)

