

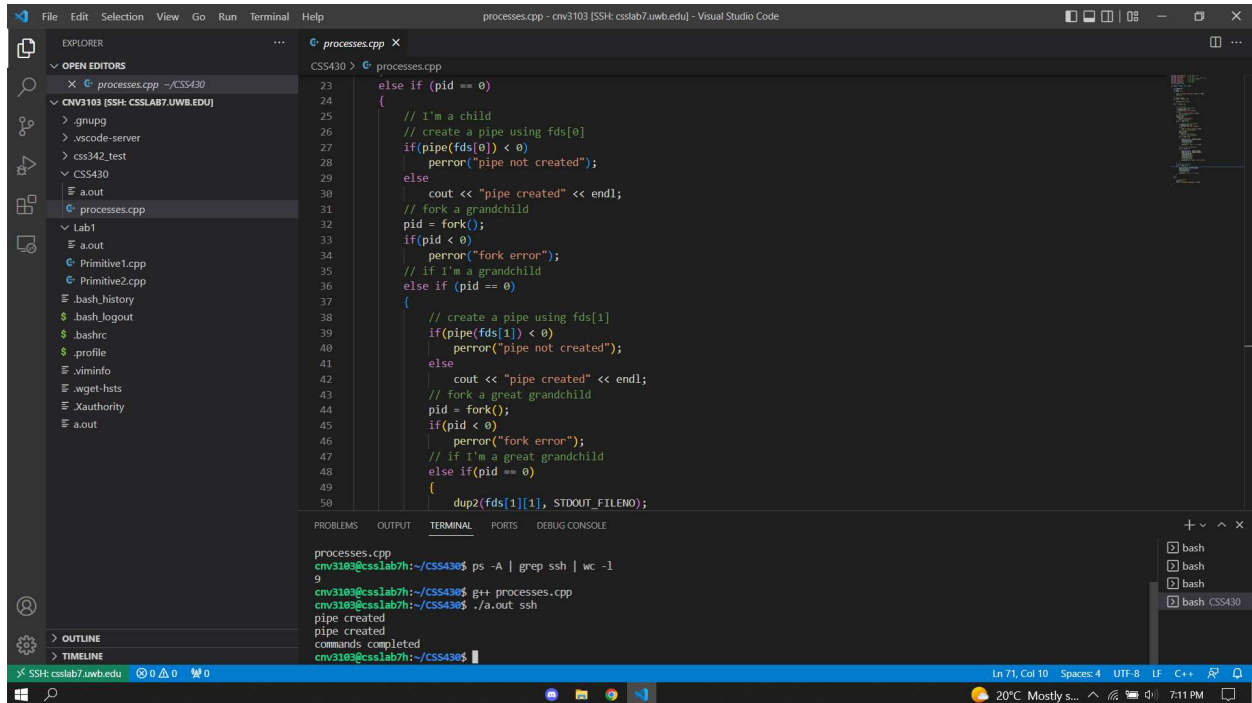
CSS430

Prof. Yang Peng

Proj1a

Nguyen Vi Cao

The following three images are the screenshot for the output of my assignment and the illustration of all processes and their pipe connections involved in processes.cpp. My username is cnv3103 and I was using Linux lab machine 7.



The screenshot shows the Visual Studio Code editor with the file `processes.cpp` open. The code is a C++ program that demonstrates process creation and pipe usage. It includes comments like `// I'm a child`, `// create a pipe using fds[0]`, `// fork a grandchild`, `// I'm a grandchild`, and `// create a pipe using fds[1]`. The code uses `fork()` to create child and grandchild processes and `pipe()` to create pipes. It also uses `dup2()` to redirect standard output to a pipe. The terminal output shows the execution of the program, including the command `ps -A | grep ssh | wc -l` which returns 9, and the command `g++ processes.cpp` which returns 0. The terminal also shows the output of the program, which includes "pipe created" and "commands completed".

Picture 1

- Note: for the same code, (I was only doing some minor changes to the comment from “pipe created” to “child pipe” none of the code was changed) but I was having two different output. Picture 1 was taken Saturday July 2<sup>nd</sup>, the output was 9 and the Picture 2 was taken on Sunday July 3<sup>rd</sup>, output was 5.

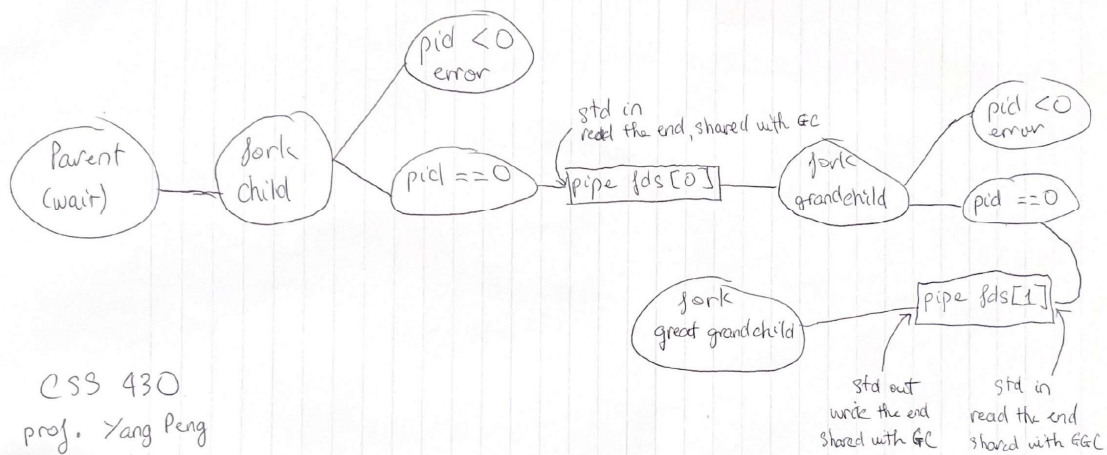
The image shows a Visual Studio Code editor window with a C++ file named `processes.cpp` open. The code implements a multi-process program using pipes and `fork()`. It creates a 'child pipe' and a 'grandchild pipe'.

```
25 // I'm a child
26 // create a pipe using fds[0]
27 if(pipe(fds[0]) < 0)
28     perror("pipe not created");
29 else
30     cout << "child pipe" << endl;
31 // fork a grandchild
32 pid = fork();
33 if(pid < 0)
34     perror("fork pipe");
35 // if I'm a grandchild
36 else if (pid == 0)
37 {
38     // create a pipe using fds[1]
39     if(pipe(fds[1]) < 0)
40         perror("pipe not created");
41     else
42         cout << "grandchild pipe" << endl;
43     // fork a great grandchild
44     pid = fork();
45     if(pid < 0)
46         perror("fork error");
47     // if I'm a great grandchild
48     else if (pid == 0)
49     {
50         dup2(fds[1][1], STDOUT_FILENO);
51         close(fds[1][0]);
52         close(fds[1][1]);
53     }
54 }
```

The terminal output shows the execution of the program:

```
cnv3103@cslab7.uwb.edu:~/CSS430$ ps -A | grep ssh | wc -l
5
cnv3103@cslab7.uwb.edu:~/CSS430$ g++ processes.cpp
cnv3103@cslab7.uwb.edu:~/CSS430$ ./a.out ssh
child pipe
grandchild pipe
commands completed
cnv3103@cslab7.uwb.edu:~/CSS430$
```

Picture 2



CSS 430  
prof. Yang Peng  
Assignment 1A.  
Nguyen Vi Cao