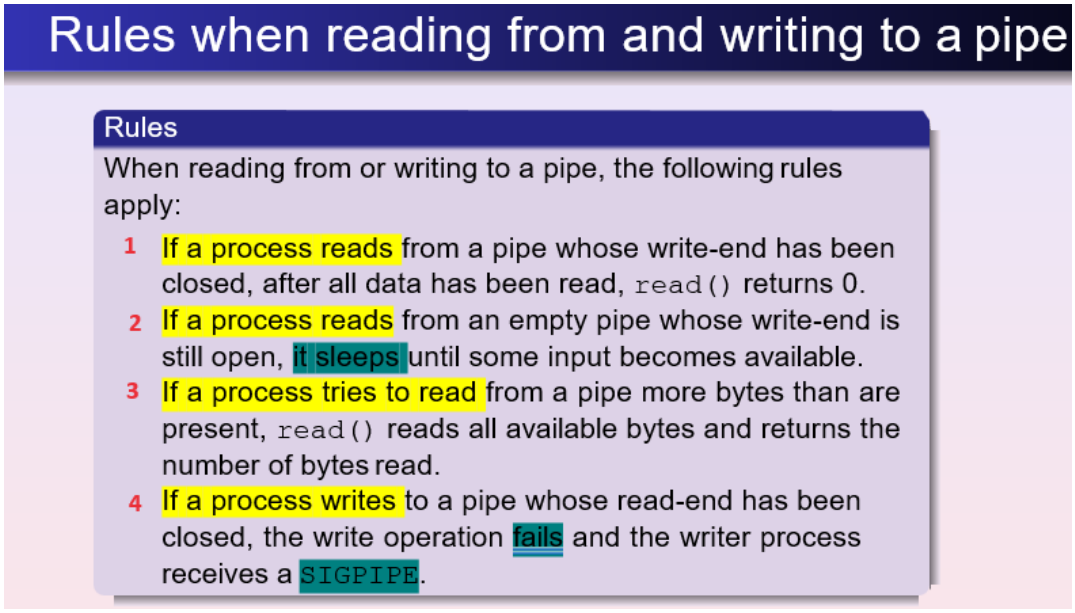
COMP 2560 Winter 2024 — Lab 8

In class, we discussed the following 4 rules when using a pipe.



In this lab, we will experience each of them.

Rule #1

For Rule #1, please see the attached code, **lab8r1.c**, read the code and run the code to see how Rule #1 is realized in the code.

Please note that in the code **lab8r1.c**, I did not create a child process such that the child process writes and the parent process reads. You are encouraged to do so, that is,

1. The parent process creates a child process. The child process intends to write to the pipe so the child process closes its read-end of the pipe. The parent process intends to read from the pipe so the parent process closes its write-end of the pipe.
2. The child process writes the message msg1 to the pipe and then closes its write-end of the pipe.
3. The parent process reads the msg1 message and displays it. The parent process then tries to read again from the pipe and observes the return value of the second read should be zero.

Please code the above 3 steps to demonstrate Rule #1 between a parent process and a child process.

Rule #2

I have demonstrated Rule #2 in class using a code I posted, please indicate what code I used in class to demonstrate this and what changes I have suggested to make to that code so that the behavior of Rule #2 is fully revealed.

No coding is required, but write a short paragraph to answer the questions above.

Rule #3

To demonstrate Rule#3, you can make small changes to **lab8r1.c** so that your code requests to read 32 bytes from the pipe, but only 16 bytes (msg1) so far are available in the pipe to be read. A sample output of such a program could look like the screenshot below.

```
danwu@delta:~/comp2560w2024/labcode$ ./lab8r3
 I have requested to read 32 bytes, but only  16  bytes in the pipe so far.
 This is the 16 bytes  I read: hello, world #1
danwu@delta:~/comp2560w2024/labcode$
```

Please make changes to lab8r1.c to demonstrate Rule #3 to produce a similar output as the screenshot above.

Rule #4

You will have a question related to Rule #4 in the upcoming assignment.

Submit code and its related script files for Rules #1 and #3. Submit the short paragraph required for Rule #2.

All by the end of 11: 59 PM, Mar. 24.