

CS 3305A: Operating Systems
Department of Computer Science
Western University
Assignment 2
Fall 2020
Due Date: October 9, 2020

Purpose

The goals of this assignment are the following:

- Get experience with the *fork()*, *wait()* and *pipe()* system functions
- Learn more about how operating systems are structured
- Gain more experience with the C programming language from an OS perspective

Inter-Processes Communications (100 points)

Write a C program that will accept three strings from the user as **command line arguments** (for example, X, Y and Z). Your program will create a *parent* and *child* where the *parent* process will read X and the *child* process will read Y and Z. The *child* process will concatenate Y and Z to generate Y'. *child* process will send Y' to *parent* process by communicating through a *pipe* (i.e., shared memory). Then, the parent process will concatenate X and Y' and output the resulting string. The expected output from your program should look like the following for the arguments "CS" "3305" and " is fun!":

1. A pipe is created for communication between *parent* (PID 2255) and *child*
2. *parent* (PID 2255) created a *child* (PID 2256)
3. *parent* (PID 2255) receives X = "CS" from the user
4. *child* (PID 2256) receives Y = "3305" and Z = " is fun!" from the user
5. *child* (PID 2256) concatenates Y and Z to generate Y' = "3305 is fun!"
6. *child* (PID 2256) writes Y' into the *pipe*
7. *parent* (PID 2255) reads Y' from the *pipe* (Y' = "3305 is fun!")
8. *parent* (PID 2255) concatenates X and Y' to generate the string: "CS3305 is fun!"

Hints: fork(), wait(), getpid(), getppid(), pipe(), write(), read()

Mark Distribution

This section describes a tentative allocation of marks assigned for the desired features **(100 points)**

- a) Parent reads X from user: 10 points
- b) Child reads Y & Z from user: 10 points
- c) A pipe was created for communication between *parent* and *child*: 20 points
- d) Child writes Y' into the pipe: 25 points
- e) Parent reads Y' from the pipe: 25 points
- f) Output the correct string: 10 points

Computing Platform for Assignments

You are responsible for ensuring that your program compiles and runs without error on the computing platform mentioned on below. **Marks will be deducted** if your program fails to compile or your program runs into errors on the specified computing platform (see below).

- Students have virtual access to the MC 244 lab, which contains 30 Fedora 28 systems. Linux machines available to you are: **linux01.gaul.csd.uwo.ca** through **linux30.gaul.csd.uwo.ca**.
- It is your responsibility to ensure that your code compiles and runs on the above systems. You can SSH into MC 244 machines.
- If you are off campus, you have to SSH to **compute.gaul.csd.uwo.ca** first (this server is also known as **sylvia.gaul.csd.uwo.ca**, in honour of Dr. Sylvia Osborn), and then to one of the MC 244 systems (**linux01.gaul.csd.uwo.ca** through **linux30.gaul.csd.uwo.ca**).
- <https://wiki.sci.uwo.ca/sts/computer-science/gaul>

Provided Files

- One C file is provided: “assignment2.c”
- Your code should only be inside the provided C file “assignment2.c”
- Hints and the necessary statements that need to be outputted have been included in “assignment2.c” file
- If you have any questions regarding the code provided to you, contact the TAs and/or the Instructor

Assignment Submission

You must submit your Assignment through OWL. Be sure to test your code on one of MC 244 systems (see “Computing Platform for Assignments” section above). **Marks will be deducted** if your program fails to compile or your program runs into errors on the computing platform mentioned above. You need to submit one file “assignment1.c”.

Assignment 2 FAQ will be made available on OWL. Also, consult TAs, and the Instructor for any question you may have regarding this assignment.