CS 3305A: Operating Systems
Department of Computer Science
Western University
Assignment 1
Fall 2021
Due Date: September 27, 2021

Purpose

The goals of this assignment are the following:

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

Parent and Child Processes (100 points)

Write a program in C that will perform the following tasks (must follow the task sequence below):

1. Your parent process will create a child process (i.e., child_1)
2. Parent process will wait for child_1 to complete before creating child_2
3. child_1 will call an external program "external_program.out" (hint: excel()) and pass its PID concatenated with string " for child_1". As a result of this external program call, child_1 will be replaced by external_program.out.
4. Parent process will create child_2 and then child_2 will create child_2.1
5. Inside child_2.1, a call to an external program "external_program.out" will be made (hint: excel()). child_2.1 must pass its PID to "external_program.out" concatenated with string " for child_2.1". As a result of this external program call, child_2.1 will be replaced by external_program.out.
6. Parent process will wait for child_1 and child_2 to be completed before it terminates.

The expected output from your program should look like the following:

```
parent (PID 333275) created child_1 (PID 333276)
parent (PID 333275) is waiting for child_1 (PID 333276) to complete before creating child_2
child_1 (PID 333276) is calling an external program external_program.out and leaving parent
From the external program: The PID was 333276 for child_1
parent (PID 333275) created child_2 (PID 333277)
child_2 (PID 333277) created child_2.1 (PID 333278)
child_2.1 (PID 333278) is calling an external program external_program.out and leaving child_2
From the external program: The PID was 333278 for child_2.1
child_1 and child_2 are completed and parent process is terminating…
```

Hints: fork(), wait(), getpid(), getppid(), execl(), strcat()

Mark Distribution

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

a) A parent process will create two child processes: 20 points
b) parent will wait for child_1 to complete before creating child_2: 20 points
c) child_1 will make a system call to an external program: 15 points
d) child_2 will create its own child child_2.1: 10 points
e) child_2.1 will make a system call to an external program: 15 points
f) parent process must not terminate until all child processes have completed: 20 points

Computing Platform for Assignments

You are responsible for ensuring that your program compiles and runs without error on the computing platform mentioned 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 (please see the Assignment 1 file transfer tutorial).
- 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**) (please see the Assignment 1 file transfer tutorial).
- https://wiki.sci.uwo.ca/sts/computer-science/gaul

Provided Files

- The source code for the external program "external_program.c" is provided.
- Do not make any changes to "external_program.c"
- When running the program, you must provide the path to "external_program.out" as an argument (see Assignment 1 tutorial Powerpoint)
- If you have any questions, please contact the designated TAs or the Instructor

Assignment Submission

You need to submit only one C file. The name of your submitted C file must be "assignment1.c". Marks will be deducted if your submitted C file name is different. 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.

Assignment 1 FAQ will be made available on OWL. Also, consult TAs, and the Instructor for any questions you may have regarding this assignment.