New York University

Tandon School of Engineering

Department of Computer Science and
Engineering

Introduction to Operating Systems

Fall 2024

Assignment 4 (10 points)

Problem 1 (2 points)

If you create a main() routine that calls fork() twice, i.e., if it includes the
following code:

```
pid_t x=-11, y=-22;
x = fork();
if(x==0) y = fork();
```

Assuming all fork() calls succeed, draw a process tree similar to that of Fig.
3.8 (page 116) in your textbook, clearly indicating the values of x and y for each
process in the tree (i.e., whether 0, -11, -22, or larger than 0). Note that the
process tree should only have one node for each process and thus the number
of nodes should be equal to the number of processes. The process tree should
be a snapshot just after all forks completed but before any process exists. Each
line/arrow in the process tree diagram shall represent a creation of a process,
or alternatively a parent/child relationship.

(Insert process tree diagram here. The diagram should show three processes:
the parent with x ¿ 0 and y = -22, a child with x = 0 and y ¿ 0, and a grandchild
with x = 0 and y = 0.)

Problem 2 (4 points)

Write a program that creates the process tree shown below:

(Insert process tree diagram here. The diagram should show a parent process with two children, and each child having one child each. A total of 5 processes.)

Problem 3 (4 points)

Write a program whose main routine obtains two parameters, a and b, from the user (i.e., passed to your program when it was invoked from the shell, a ¿ 0, b ¿ 0) and creates a child process. The child process shall then compute and print the greatest common divisor (GCD) of a and b using Euclid's algorithm. The parent waits for the child to exit and then prints the least common multiple (LCM) of a and b, which is calculated as (a * b) / GCD(a, b). Do not use IPC in your solution to this problem (i.e., neither shared memory nor message passing).

Problem 4 (2 points)

Explain the difference between a zombie process and an orphan process. Give examples of how each might arise.

Problem 5 (2 points)

Describe a scenario where using signals might be preferable to using pipes for inter-process communication. Describe a scenario where pipes would be preferable to signals.

Problem 6 (4 points)

Write a C program that uses the 'execl' system call to execute the 'ls -l' command. The program should check the return value of 'execl' and print an appropriate error message if the execution fails. The program should also handle potential SIGINT signals gracefully, printing a message indicating that the program is exiting due to an interrupt before terminating.

Problem 7 (2 points)

Suppose you have a parent process that creates multiple child processes. Each child process performs a computationally intensive task and then exits. If the parent process simply uses 'wait()' to wait for each child, what is a potential problem, and how could this problem be addressed?

What to hand in (using Brightspace)

Please submit the following files individually:

1. Source file(s) with appropriate comments. The naming should be similar to "lab#_$.c" (# is replaced with the assignment number and $ with the question number within the assignment, e.g., `lab4_b.c`, for lab 4, question b OR `lab5_1a` for lab 5, question 1a).

2. A single pdf file (for images + report/answers to short-answer questions), named "lab#.pdf" (# is replaced by the assignment number), containing:

   - Screenshot(s) of your terminal window showing the current directory, the command used to compile your program, the command used to run your program and the output of your program.

3. Your Makefile, if any. This is applicable only to kernel modules.

RULES

- You shall use kernel version 4.x.x or above. You shall not use kernel version 3.x.x.

- You may consult with other students about GENERAL concepts or methods but copying code (or code fragments) or algorithms is NOT ALLOWED and is considered cheating (whether copied from other students, the internet or any other source).

- If you are having trouble, please ask your teaching assistant for help.

- You must submit your assignment prior to the deadline.