New York University
Tandon School of Engineering
Department of Computer Science and
Engineering
Introduction to Operating Systems
Fall 2024
Assignment 4 (70 points)

Problem 1 (10 points) If you create a main() routine that calls fork() four times, i.e., if it includes the following code:

```
pid_t w=-1, x=-2, y=-3, z=-4;
w = fork();
if(w==0) x = fork();
if(x>0) y = fork();
if(y==0) z = 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 w, x, y and z for each process in the tree (i.e., whether 0, -1, -2, -3, -4, 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 exits. Each line/arrow in the process tree diagram shall represent a creation of a process, or alternatively a parent/child relationship.

Problem 2 (10 points) Write a program that creates the process tree shown below: (Assume a tree with a root process A, which forks B and C. B forks D and E. C forks F.) Clearly label each process with a unique identifier (A, B, C, D, E, F).

Problem 3 (10 points) Write a program whose main routine obtains a parameter n from the user (i.e., passed to your program when it was invoked from the shell, n¿5) and creates two child processes. The first child process shall compute the sum of integers from 1 to n, and the second child process shall compute the product of integers from 1 to n. The parent waits for both children to exit and then prints the difference between the sum and the product.

Do not use IPC in your solution to this problem (i.e., neither shared memory nor message passing).

Problem 4 (10 points) Consider a scenario with three processes, P1, P2, and P3. P1 creates P2, and P2 creates P3. Describe the parent-child relationships and potential race conditions if P1, P2, and P3 all try to access and modify a shared global variable simultaneously without any synchronization mechanisms.

Problem 5 (10 points) Write a C program that uses the 'execl()' system call to execute the 'ls' command with the '-l' option. The program should then print a message indicating that the 'ls -l' command has completed.

Problem 6 (10 points) Explain the difference between the 'fork()' and 'vfork()' system calls. Discuss the advantages and disadvantages of using 'vfork()'. Why is 'vfork()' often considered less safe than 'fork()'?

Problem 7 (10 points) Design a program that simulates a simple producer-consumer problem using two processes and a shared buffer of size 5. The producer process generates random integers and adds them to the buffer. The consumer process removes integers from the buffer and prints them. Implement appropriate synchronization mechanisms (e.g., semaphores) to prevent race conditions and ensure that the producer doesn't write to a full buffer and the consumer doesn't read from an empty buffer.

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.

- 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 form 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.