

New York University
Tandon School of Engineering
Department of Computer Science and Engineering
Introduction to Operating Systems
Fall 2024
Assignment 4 (10 points)

Your Name

September 12, 2025

Problem 1 (3 points)

(a) (1 point) Explain the difference between a process and a thread. Provide examples of when you might choose to use threads over processes, and vice versa.

(b) (2 points) Describe the concept of context switching in an operating system. What are the key components of the process state that need to be saved and restored during a context switch? How does context switching impact system performance?

Problem 2 (4 points)

(a) (2 points) A `main()` routine calls `fork()` twice as follows:

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

Assuming all `fork()` calls succeed, draw the process tree. Clearly label each process with the values of `x` and `y`. (A visual representation would be included here in a PDF submission, showing a tree with branches and nodes labeled with the values of `x` and `y` for each process. This cannot be accurately represented in LaTeX without a graphics package and specific image inclusion).

(b) (2 points) Write a C program (`lab4_problem2.c`) that creates the process tree described in part (a). The program should print the process ID (PID) and the values of `x` and `y` for each process. Include error handling for failed `fork()` calls.

Problem 3 (3 points)

Write a C program (`lab4_problem3.c`) that simulates a simple producer-consumer problem using two threads. The producer thread generates random numbers between 1 and 100 and adds them to a shared buffer (represented as a circular buffer with a fixed size, e.g., 5). The consumer thread removes numbers from the buffer and prints them to the console. Use appropriate synchronization mechanisms (e.g., mutexes, condition variables) to prevent race conditions and ensure proper data sharing between the threads. The program should run for a specified number of iterations (e.g., 20), after which both threads should terminate gracefully. Include error handling and clear comments in your code.

What to hand in (using Brightspace)

1. Source file(s) with appropriate comments (e.g., `lab4_problem2.c`, `lab4_problem3.c`). 2. A single PDF file (`lab4.pdf`) containing:

- Screenshot(s) of your terminal window showing compilation, execution, and output for Problem 2 and Problem 3.
- The process tree diagram for Problem 2 (a).

3. Makefile (if applicable).

RULES

- Use kernel version 4.x.x or above.
- Do not use kernel version 3.x.x.
- Collaboration on general concepts is allowed; code copying is strictly prohibited.
- Ask your TA for help if needed.
- Submit your assignment before the deadline.