Task 1 [25 minutes]

Write a program in which multiple processes write data into a single named pipe, and another process reads the data. Ensure proper synchronization and data integrity.

Task 2 [30 minutes]

Write a program that builds a chat application between two unrelated processes using named pipe. Both the processes will read and write data through named the pipe. The communication should stop as soon as any of the processes has written “bye” in the pipe.

Hint:
You can use sleep to manage synchronization issues.

Task 3 [30 minutes]

Create a program that uses non-blocking I/O to read and write data to a named pipe, allowing processes to continue working rather than waiting for data to become available.

Hint:
Use O_NONBLOCK flag for non-blocking I/O to read and write.

Task 4 [35 minutes]

Build a logging system using named pipes, where multiple processes can write log entries into a named pipe, and a central process handles logging to files or other destinations.