Assignment
OS Lab

September 2025

Problem statement

Implement a C program that demonstrates three **classic synchronization** scenarios using semaphores and threads:

- **Producer-Consumer:** Two producer threads generate multimedia files (audio/video) and place them in a shared buffer. Two consumer threads process files from the buffer. Use semaphores to synchronize access and to avoid buffer overflow/underflow.
- **Reader-Writer:** Three reader threads and one writer thread access a shared metadata database for the multimedia files. Readers may read simultaneously, but writers require exclusive access. No reader can read while a writer is writing.
- **Dining Philosopher (Resource Allocation):** Five processor threads each require access to two adjacent shared resources (e.g., Encoder, Decoder, Compressor, De-compressor, Filter) to process content. Ensure safe resource sharing without deadlock or starvation using semaphores.

Implement thread-safe code so that your program clearly demonstrates producer, consumer, reader, writer, and processor activities.