

## Laboratório 7: Resolvendo problemas clássicos de concorrência e sincronização usando Semáforos e Monitores

### 1. Objetivos

- Desenvolver aplicações concorrentes que acessam recursos compartilhados.
- Compreender os mecanismos de semáforos e monitores para sincronização.

### 2. Materiais

- Distribuição Linux/Unix
- Ambiente de desenvolvimento para C/C++.
- Bibliotecas de programação: *pthread*, *semaphore* e outras.

### 3. Atividades

Considere os problemas clássicos de concorrência: leitores/escritores, jantar dos filósofos e barbeiro dorminhoco. Selecione dois problemas e implemente um usando semáforos e outro usando monitores (variáveis de condição e mutex em C). Os programas devem exibir como saída (tela ou arquivo) o que está acontecendo durante a interação das threads.

---

### Instruções para entrega via Moodle:

Colocar a solução de cada questão em uma pasta separada nomeada por `ex1`, `ex2` respectivamente. Cada pasta deve conter o *Makefile* para o exercício e *README* (se necessário). Inclua em todos os arquivos de código fonte um cabeçalho com a funcionalidade, autor(es) e data. Adicione comentários antes dos nomes das funções descrevendo a finalidade e os parâmetros de entrada e saída. Adicione comentários nos principais trechos de códigos do programa. Compactar em um único arquivo (**tar.gz**) e enviar via Moodle.

---