

Módulo 2 - Laboratório 8

Implementação de uma aplicação que usa o padrão leitores/escritores usando o conceito de monitores

Computação Concorrente (ICP-117)
Prof. Silvana Rossetto

¹IC/UFRJ

Introdução

O objetivo deste Laboratório é continuar implementando problemas clássicos de concorrência usando o conceito de **monitores** com o suporte da linguagem Java. Nesta semana, vamos trabalhar com o problema dos **leitores/escritores**.

Atividade 1

Objetivo: Projetar e implementar uma aplicação concorrente em Java usando o padrão básico **leitores/escritores**.

Descrição: A aplicação terá como elemento central uma **variável inteira inicializada com valor 0** (representando a nossa base de dados) que será lida e alterada pelas threads. Três tipos de threads deverão ser implementadas:

- **Leitora:** lê a variável e a imprime na tela indicando se é um valor par ou ímpar.
- **Escritora:** modifica a variável escrevendo o valor do seu identificador de thread.
- **Leitora/Escritora:** primeiro lê a variável e a imprime na tela; faz um processamento bobo qualquer; e depois modifica a variável incrementando seu valor de 1.

IMPORTANTE: Implemente de forma separada a classe/monitor com o **padrão básico de leitores e escritores** (pode usar integralmente a implementação disponibilizada pela professora). Implemente também de forma separada as classes para cada tipo de thread. E por último, implemente a classe principal. Todas as threads devem acessar a mesma variável inteira (variável compartilhada), usando para isso os métodos do **padrão básico de leitores e escritores** (*EntraLeitor*, *SaiLeitor*, *EntraEscritor*, *SaiEscritor*). Os acessos de leitura e escrita devem estar entre essas chamadas.

Roteiro:

1. Considere um número L de threads leitoras ($L \geq 1$), um número E de threads escritoras ($E \geq 1$) e um número LE de threads leitoras e escritoras ($LE \geq 1$).
2. Acrescente no seu código a impressão de **informações que permitam acompanhar a execução da aplicação para verificar se as condições lógicas do problema são satisfeitas**.
3. Execute a aplicação **várias vezes** e avalie os resultados obtidos.
4. Altere o número de threads de cada tipo e reexecute a aplicação.

Disponibilize o código implementado na (as três versões) em um ambiente de acesso remoto (GitHub ou GitLab). Use o formulário de entrega desse laboratório para enviar o link do repositório do código implementado e responder às questões propostas.