

Módulo 3 - Laboratório 9

Exercício com semáforos

Computação Concorrente (ICP-117)
Prof. Silvana Rossetto

¹Instituto de Computação/UFRJ

Introdução

O objetivo deste Laboratório é praticar o uso de semáforos para implementar exclusão mútua e sincronização condicional. Vamos repetir o exercício do Lab 5, agora usando **semáforos**, ao invés de locks e variáveis de condição.

Atividade 1

Objetivo: Projetar e implementar um programa concorrente onde a ordem de execução das threads é controlada no programa.

Roteiro: Implemente um programa com 5 threads:

- A thread 1 imprime a frase “Volte sempre!”.
- A thread 2 imprime a frase “Fique a vontade.”
- A thread 3 imprime a frase “Sente-se por favor.”
- A thread 4 imprime a frase “Aceita um copo d’água?”
- A thread 5 imprime a frase “Seja bem-vindo!”

As threads devem ser criadas todas de uma vez na função main. A ordem de impressão das mensagens (execução das threads) deverá ser:

- A thread 5 deve sempre ser a primeira a imprimir sua mensagem.
- A thread 1 deve sempre ser a última a imprimir sua mensagem.
- A ordem em que as threads 2, 3 e 4 imprimem suas mensagens não importa, mas todas devem sempre imprimir suas mensagens depois da thread 5 e antes da thread 1.

Exemplos de saídas esperadas para o programa são:

```
(1)
Seja bem-vindo!
Aceita um copo d’água?
Sente-se por favor.
Fique a vontade.
Volte sempre!
```

```
(2)
Seja bem-vindo!
Fique a vontade.
Aceita um copo d’água?
Sente-se por favor.
Volte sempre!
```

```
(3)
Seja bem-vindo!
Fique a vontade.
```

Sente-se por favor.
Aceita um copo d'água?
Volte sempre!

Disponibilize o código implementado na **Atividade 1** 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 listar dúvidas ou dificuldades encontradas.