

# EP 2 - Seção Crítica e Posix threads

- Gabriel Hoffman - 10783250
- Guilherme Balog - 11270649

## Solução

Inicialmente tivemos dificuldades com a solução proposta pelos pseudocodigos, visto que estes estavam retornando uma solução que ficava alternando entre uma thread e outra sendo executada de forma completa e não com uma thread iniciando assim que outra libera a seção crítica. Porém após alguns testes e modificações conseguimos alcançar um resultado satisfatório que acreditamos estar correto.

Como saída tivemos o seguinte resultado:

```
[Inicio do programa]

[Fim do programa]

Thread 0 entrou na seção crítica
Count: 1
Thread 0 saiu na seção crítica
Thread 0 entrou na seção não crítica
Thread 0 saiu na seção não crítica

Thread 1 entrou na seção crítica
Count: 2
Thread 1 saiu na seção crítica
Thread 1 entrou na seção não crítica

Thread 0 entrou na seção crítica
Thread 1 saiu na seção não crítica
Count: 3
Thread 0 saiu na seção crítica
Thread 0 entrou na seção não crítica

Thread 1 entrou na seção crítica
Thread 0 saiu na seção não crítica
Count: 4
Thread 1 saiu na seção crítica
Thread 1 entrou na seção não crítica

Thread 0 entrou na seção crítica
Thread 1 saiu na seção não crítica
```

```
Count: 5
Thread 0 saiu na seção crítica
Thread 0 entrou na seção não crítica

Thread 1 entrou na seção crítica
Thread 0 saiu na seção não crítica
Count: 6
Thread 1 saiu na seção crítica
Thread 1 entrou na seção não crítica

Thread 0 entrou na seção crítica
Thread 1 saiu na seção não crítica
Count: 7
Thread 0 saiu na seção crítica
Thread 0 entrou na seção não crítica

Thread 1 entrou na seção crítica
Thread 0 saiu na seção não crítica
Count: 8
Thread 1 saiu na seção crítica
Thread 1 entrou na seção não crítica

Thread 0 entrou na seção crítica
Thread 1 saiu na seção não crítica
Count: 9
Thread 0 saiu na seção crítica
Thread 0 entrou na seção não crítica

Thread 1 entrou na seção crítica
Thread 0 saiu na seção não crítica
Count: 10
Thread 1 saiu na seção crítica
Thread 1 entrou na seção não crítica
```

Alguns pontos interessantes podem ser observados:

- O programa exibe "Fim do programa" logo no início pois a função `main()` terminou de executar, mas enquanto isso as *threads* continuam executando
- Após a criação da *thread* 0 e da sua primeira execução, as duas *threads* começam a alternar, assim que uma acaba a seção crítica, a outra começa a executar.