

Computação Concorrente (DCC/UFRJ)

Aula 8: Exercícios usando locks e variáveis de condição

Prof. Silvana Rossetto

15 de outubro de 2019

- datas das provas
- ferramenta de detecção de condição de corrida

Variações do problema do Produtor/Consumidor

Lembrando, as condições básicas do problema são:

- 1 produtores não podem inserir novos elementos quando a área de dados está cheia;
- 2 consumidores não podem retirar elementos quando a área de dados está vazia;
- 3 os elementos devem ser retirados na mesma ordem em que foram inseridos;
- 4 os elementos inseridos não podem ser perdidos (sobreescritos por novos elementos).

Altere o código do produtor para que ele **preencha o Buffer inteiro a cada inserção**.

```
1: pthread_mutex_t mutex; pthread_cond_t cond;
2: int deve_esperar = 0, esperando = 0; int ativas = 0;
3: void *recurso (void *tid) {
4:     pthread_mutex_lock(&mutex);
5:     while(deve_esperar) {
6:         esperando++;
7:         pthread_cond_wait(&cond, &mutex);
8:         esperando--;
9:     }
10:    ativas++; deve_esperar=(ativas==3);
11:    if(esperando>0 && !deve_esperar)
12:        { pthread_cond_signal(&cond); }
13:    pthread_mutex_unlock(&mutex);
14:    /* usa o recurso */
15:    pthread_mutex_lock(&mutex);
16:    ativas--;
17:    if(ativas==0) { deve_esperar=0; }
18:    if(esperando>0 && !deve_esperar)
19:        { pthread_cond_signal(&cond); }
20:    pthread_mutex_unlock(&mutex);
}
```

Variações do problema dos leitores e escritores

Lembrando, as condições básicas do problema são:

- 1 Os leitores podem ler simultaneamente uma região de dados compartilhada
- 2 Apenas um escritor pode escrever a cada instante em uma região de dados compartilhada
- 3 Se um escritor está escrevendo, nenhum leitor pode ler a mesma região de dados compartilhada

Altere a lógica do problema para que o leitor somente leia se não houver escritor escrevendo ou **esperando para escrever**.