

Relatório - Lab 6

Bloqueio de Leitura

Na função `reader_lock`, as seguintes condições garantem a prioridade de escrita:

```
while (escritor_ativo || escritores_fila) {  
    pthread_cond_wait(&cond_leitor, &mutex);    //  
    leitores esperam se houver escritores ativos ou na  
    fila  
}
```

- Se um escritor estiver ativo (`escritor_ativo == 1`), novos leitores serão bloqueados.
- Se houver escritores na fila (`escritores_fila > 0`), novos leitores também serão bloqueados até que os escritores tenham terminado.

Assim que um escritor for solicitado (mesmo que nenhum escritor esteja ativo no momento), **nenhuma nova operação de leitura possa começar** até que o escritor tenha completado sua

operação. Leitores já ativos podem continuar suas operações, mas novos leitores serão impedidos de começar, respeitando a prioridade de escrita.

Desbloqueio de Leitura

Na função `reader_unlock`, temos o seguinte comportamento:

```
if (leitores == 0) {  
    pthread_cond_signal(&cond_escritor);    // acorda  
    um escritor se não houver mais leitores  
}
```

Assim que **não houver mais leitores ativos**, o escritor que está esperando será acordado para realizar sua operação. Uma vez que os leitores terminam, o próximo escritor da fila tem prioridade para executar.

Bloqueio de Escrita

Na função `writer_lock`, a prioridade de escrita é garantida por:

```
while (leitores > 0 || escritor_ativo) {  
    pthread_cond_wait(&cond_escritor, &mutex);    //  
    escritores esperam se houver leitores ou escritor  
    ativo  
}
```

- Um escritor só pode realizar sua operação se não houver leitores ativos (`leitores == 0`) e nenhum escritor ativo (`escritor_ativo == 0`).
- Escritores na fila aguardam até que todos os leitores tenham terminado, garantindo que a operação de escrita tenha exclusividade.

Quando há **leitores ativos**, os escritores esperam. Mas, quando um escritor entra na fila, **novas leituras são bloqueadas** pela função `reader_lock`, permitindo que o escritor eventualmente tenha a prioridade para executar.

Desbloqueio de Escrita

Na função `writer_unlock`, o escritor que termina sua operação notifica da seguinte forma:

```
if (escritores_fila > 0) {  
    pthread_cond_signal(&cond_escritor);    // acorda  
    outro escritor se houver escritores esperando  
} else {  
    pthread_cond_broadcast(&cond_leitor);    // acorda  
    todos os leitores se não houver escritores esperando  
}
```

- Se houver outros escritores na fila, o próximo escritor será acordado, respeitando a prioridade de escrita.
- Se não houver escritores na fila, todos os leitores são notificados de que podem começar suas operações.

Enquanto houver escritores esperando para acessar a lista, eles serão priorizados em relação aos leitores.

Comportamento Durante a Execução

- Quando uma thread tenta uma operação de escrita (inserção ou remoção), qualquer nova thread que tentar ler será bloqueada até que a operação de escrita seja concluída.
- Quando a thread escritora termina, ela sinaliza a próxima thread escritora (se houver) ou permite que as threads leitoras continuem (se não houver mais escritores esperando).
- Na presença de escritores, as leituras são suspensas até que as operações de escrita sejam atendidas.