



INF 1316

Relatório de Atividade - Lab 4

2210872 Felipe Antelo Machado de Oliveira
2112171 Pedro Henrique de Oliveira Valentim Gomes

Objetivo:

O Lab4, que possui 2 exercícios, visa recriar os mesmos exercícios do lab3, porém, dessa vez, utilizando conceitos de memória compartilhada para o armazenamento de vetores. Sendo assim, similaridades tanto em código, como em resultados são esperados desse lab. No entanto, buscamos observar as reais diferenças que resultam dessa mudança de uso de memória compartilhada, explorando novamente formas de paralelismo e de concorrência.

Estrutura do Programa :

Os programas gerados tem algumas coisas em comum, sendo elas a memória compartilhada criada com shmget, mapeada com shmat e removida com shmctl. Além disso, ambas possuem um Vetor a de 10.000 posições inicializado com 10 para cada. Para o código 1, temos que existe uma divisão do vetor em partes não sobrepostas entre os 10 processos. Já para o código 2, observamos que há uma situação que faz com que os 10 processos operem simultaneamente sobre todo o vetor.

Solução do programa:

No primeiro exercício, o vetor de 10.000 posições é dividido igualmente entre os 10 processos filhos. Cada processo trabalha exclusivamente em uma fatia de 1.000 elementos, multiplicando cada valor por 2 e acumulando a soma dessa porção. Ao final, essa soma parcial é armazenada em um vetor de somas na memória compartilhada. Após a execução de todos os filhos, o processo pai acessa essas somas parciais e calcula a soma total do vetor. Como não há sobreposição entre as áreas de trabalho dos processos, não ocorrem conflitos ou inconsistências, garantindo um paralelismo eficiente e correto.

No segundo exercício, os 10 processos filhos operam simultaneamente sobre todo o vetor, sem qualquer tipo de divisão ou controle de acesso. Cada processo percorre o vetor completo, multiplicando cada valor por 2 e somando 2 ao resultado. Como todos acessam e modificam a mesma área de memória ao mesmo tempo, sem sincronização, pode ser que os resultados finais variem entre execuções, ou que nem todas as posições do vetor tenham o mesmo valor no final. Esse comportamento serve como uma demonstração prática dos problemas de

concorrência sem controle em sistemas com memória compartilhada, como desejavam que fosse observado para esse nosso lab 4.

Observação e conclusões:

Como citado anteriormente, a grande diferença do lab3 para o lab4 é a existência de uma memória compartilhada, isso é, ocorre uma substituição de pipes por shmget. Existem, então, vantagens relacionadas com as mudanças do lab3 para o lab4, principalmente de desempenho, ao permitir que os processos filhos acessem diretamente o vetor original, sem a necessidade de cópias ou envio de dados por meio de pipes. Em cada exercício, saiu um resultado diferente por conta disso.

No exercício 1, isso tornou o paralelismo mais eficiente e direto, já que cada processo trabalhava em uma parte isolada do vetor sem interferência mútua.

Por outro lado, no exercício 2, o uso do mesmo vetor por todos os processos expôs um cenário real de concorrência, no qual a ausência de sincronização pode levar a resultados que o grupo considerou inconsistentes, mostrando na prática os riscos de condições de corrida.

Assim, o Lab 4 reforça a importância de dividir bem os dados e/ou utilizar mecanismos de controle de acesso quando vários processos compartilham a mesma memória.