

Lab 4

João Assayag 118113834

****Atividade 1****

Os valores impressos foram diferentes a cada execução? Por quê?

Sim, as threads não seguem uma ordem fixa, e isso muda a cada execução. Como elas compartilham uma variável global, uma thread pode ler o valor antes que outra termine de alterá-lo, o que causa erros. Com isso, as threads acabam escrevendo o mesmo valor na memória, perdendo incrementos.

É possível que a variável `soma` termine com valor acima de 200.000 quando executamos com 2 threads? Por quê?

Não, como o código não tem incremento, não é possível passar do valor máximo do loop de 10.000 por thread.

****Atividade 2****

Execute o programa várias vezes. Os valores impressos foram sempre os esperados? Por quê?

Sim, agora as threads sempre retornam os mesmos valores, pois elas não leem a variável enquanto outra thread está alterando-a. Isso garante que nenhuma thread sobrescreva a memória com valores antigos, evitando a perda de incrementos e a sobrescrita.

****Atividade 3****

Quais valores devem ser impressos na saída do programa? (todos os múltiplos de 10 dentro da faixa de valores assumidos pela variável `soma`, alguns múltiplos de 10, os primeiros 10.000 múltiplos de 10, os últimos 10.000 múltiplos de 10?)

A thread de impressão não segue uma ordem sequencial, então ela não imprime todos os múltiplos de 10. Ela executa quando consegue, dependendo de obter tempo de processamento e de rodar após outras threads terem rodado. Quando isso ocorre e coincidentemente pega um número divisível por 10, ela imprime.

Execute o programa várias vezes e observe os resultados. Os valores impressos foram os esperados? Por quê?

Não, os valores impressos não foram os esperados. O programa acabou imprimindo números que não são múltiplos de 10. Isso aconteceu devido a um erro de atomicidade, pois a verificação se um número é múltiplo de 10 pode estar desatualizada quando a thread vai imprimir o número, já que ele pode ter sido alterado por outra thread nesse intervalo.

Altere o programa usando locks para resolver o problema de violação de atomicidade. Agora, quais valores espera-se que sejam impressos na saída do programa?

Sim, agora, como há um lock quando a variável é lida, até ela ser impressa, não pode ser alterada, garantindo a corretude do programa mostrando valores múltiplos de 10.

Execute o programa várias vezes e observe os resultados. O problema foi resolvido? Justifique.

Sim, pois não apresentou nenhuma incoerência em todos os testes