

Módulo 1 - Laboratório 3

Implementação e avaliação de aplicações concorrentes (parte 2)

Computação Concorrente (MAB-117) 2021.1 REMOTO
Prof. Silvana Rossetto

¹Instituto de Computação/UFRJ

Introdução

O objetivo deste Laboratório é projetar e implementar uma versão concorrente para o problema de **encontrar o maior e o menor elemento em um vetor**; e avaliar o desempenho da aplicação em termos de tempo de execução. Usaremos a linguagem C e a biblioteca *Pthreads*.

Acompanhe a explanação da professora nas vídeo-aulas deste laboratório. Se tiver dúvidas, entre em contato por email.

Atividade 1

Objetivo: Implementar uma solução sequencial e uma solução concorrente para o problema de encontrar o menor e o maior valor em um vetor de números reais.

Roteiro:

1. Implemente a função de inicialização do vetor de entrada preenchendo seus campos com **valores aleatórios** do tipo **float**. O **número de elementos (N) do vetor deve ser informado pelo usuário na chamada do programa**. *Sugestão: defina a variável N do tipo **long long int** e use a função **atoll()** para converter o valor recebido do usuário (string) para **long long int**.*
2. Implemente uma **função sequencial** para encontrar o menor e o maior valor no vetor de entrada e meça o tempo de execução.
3. Implemente uma **função concorrente** para encontrar o menor e o maior valor no vetor de entrada e meça o tempo de execução. O **número de threads (NTHREADS) deve ser informado pelo usuário na chamada do programa**. *Use a função **pthread_exit()** para retornar o valor calculado pelas threads para a função **main()**.*
4. Verifique a corretude da solução concorrente comparando seus resultados com os resultados da versão sequencial para o mesmo vetor de entrada.
5. Calcule o ganho de desempenho (aceleração) obtido com a versão concorrente: $(T_{sequencial}/T_{concorrente})$. Considere os seguintes valores de N: 10^5 , 10^7 , 10^9 . Para a versão concorrente, experimente com 2 e 4 threads. *Inicialize o vetor uma vez, e repita as execuções das funções sequencial e concorrente várias vezes para cada configuração dos parâmetros de entrada. Calcule o tempo médio obtido nessas execuções para a versão sequencial e para a versão concorrente e então calcule a aceleração. Escreva os resultados de aceleração obtidos no README do código no GitHub ou GitLab.*

Entrega do laboratório: Disponibilize o código implementado nas duas atividades em um ambiente de acesso remoto (GitHub ou GitLab). Use o formulário de entrega desse laboratório para enviar o link do repositório do código implementado e responder às questões propostas.