

Disciplina: DIM0612 — Programação Concorrente
Docente: Everton Ranielly de Sousa Cavalcante
Discente: Felipe Cortez de Sá

Relatório

1 Introdução

Este relatório mostra os resultado do trabalho da multiplicação de matriz programada sequencialmente e com threads comparando os resultado etc etc etc

2 Detalhes da implementação

O problema foi resolvido utilizando a linguagem de programação C++11 com a biblioteca `std::thread`. Para compilar, no Mac foi selecionado o clang e em Ubuntu g++, ambos utilizando a flag `-O2` para otimizações.

3 Estratégia

A matriz foi codificada como um vetor de tamanho $n * n$. Para a implementação com t threads, cada thread é responsável pelo cálculo de $\frac{n*n}{t}$ elementos da matriz C . Caso a divisão não seja exata, a última thread calcula o resto dos elementos, ou seja, $\frac{n*n}{t} + ((n*n) \bmod t)$.

4 Medição de desempenho

Foi utilizada a biblioteca `std::chrono` para medir o tempo de execução entre o momento anterior à chamada da primeira thread e o posterior ao último join.

5 Metodologia

Os testes foram feitos a partir de um script em Python

6 Resultados

gráficos do matplotlib tabela com mínimo, médio e máximo para cada dimensão da matriz

7 Discussão

A partir da realização desse trabalho foi possível concluir que o uso de threads pode aumentar o desempenho na resolução de problemas paralelos.