

Trabalho 2

Implementar programa sequencial e paralelo para multiplicar matrizes. Use argumentos para executar o programa ou comandos do pré-processador para definir o número de threads e as dimensões da matriz pois serão parâmetros para a execução. Por exemplo:

```
>Exec multiplicaMatriz 1 128 >saidaExecucao
```

ou,

```
#define NUM_THREADS 1  
#define dimensão 128
```

Na multiplicação de matrizes usar diferentes números de threads (1, 2, 4, ...).

Conte o tempo de execução total do programa e conte também apenas o tempo da operação de multiplicação de matrizes.

Faça diversas tomadas de tempo com tamanhos de matrizes diferentes (128x128, 256x256, 512x512, ...) e elabore um gráfico de dimensão das matrizes x tempo de execução, para cada número de threads utilizado.

Gere números aleatórios no intervalo [0,10] para inicializar as matrizes.

Sugestão: inicialmente use uma matriz de dimensão menor (por exemplo, 4X4) para verificar que seu algoritmo funciona (use NUM_THREADS=2) e, apenas neste caso, utilize a impressão das matrizes para possibilitar a conferência.

Sobre o relatório:

A entrega do trabalho consiste em gerar um relatório que deve conter o enunciado do trabalho, o código gerado, o resultado da execução da matriz que comprova que o código está correto, a saída das execuções para tomada de tempo indicando o número de threads e as dimensões das matrizes utilizadas, os gráficos resultantes das tomadas de tempo e uma conclusão contendo a sua análise sobre os resultados obtidos (se possível, relacionar com o número de núcleos ou de threads que o seu computador dispõe).

Lembrete:

Ao tomar o tempo de execução do programa garanta que nenhum outro programa está executando na máquina (antivírus, browser, etc).