



Laboratório 3 **Programação Concorrente**

Vitória Mendes Cortes Chaves, 120045487

Abril, 2024.

O contexto do laboratório consiste em criar um programa concorrente que realiza o produto de duas matrizes de dimensões diferentes, escolhidas via comando no terminal, que visa estimar o quanto de aceleração e eficiência conseguimos obter ao aumentar o número de threads que irão dividir as tarefas.

Na divisão de tarefas, houve a escolha de fornecer uma quantidade de linhas da matriz resultante para cada thread calcular concorrentemente, a fim de balancear a carga de tarefas, e possui a finalidade do programa ser executado de forma mais rápida.

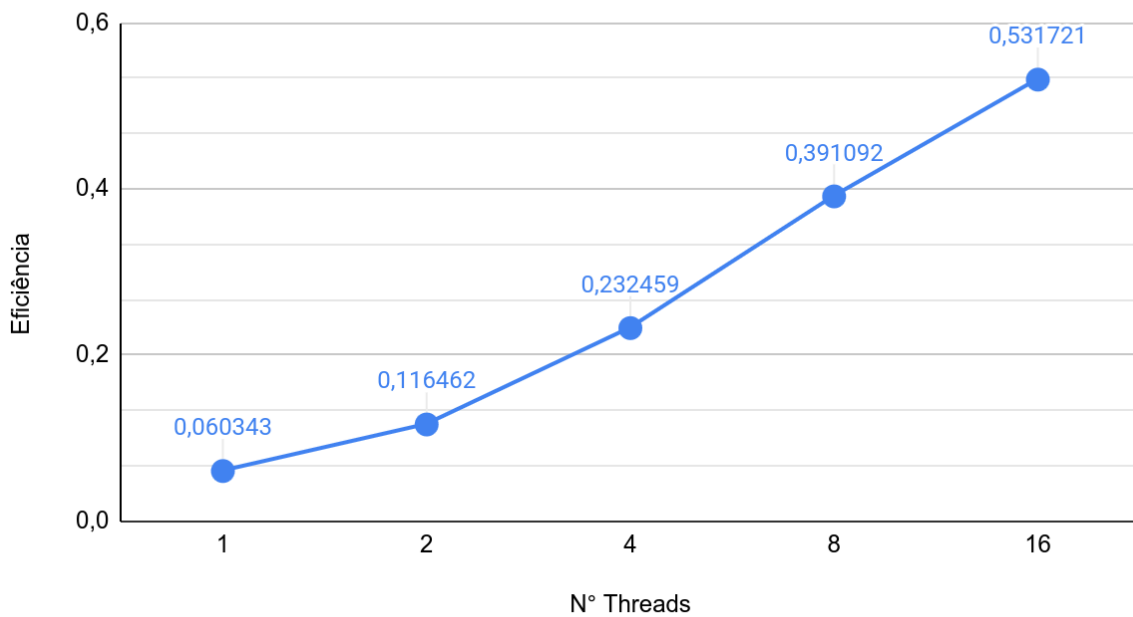
Para quantificar o tempo, escolhi considerar apenas as partes de processamento, onde para estimar eficiência e aceleração, utilizei o tempo médio de processamento das threads, e apenas do método que realiza o produto das matrizes no programa sequencial.

Após executar 3 vezes o processamento, extraindo o tempo médio, e com 1,2,4, 8 e 16 (bônus) threads, obtive os seguintes resultados:

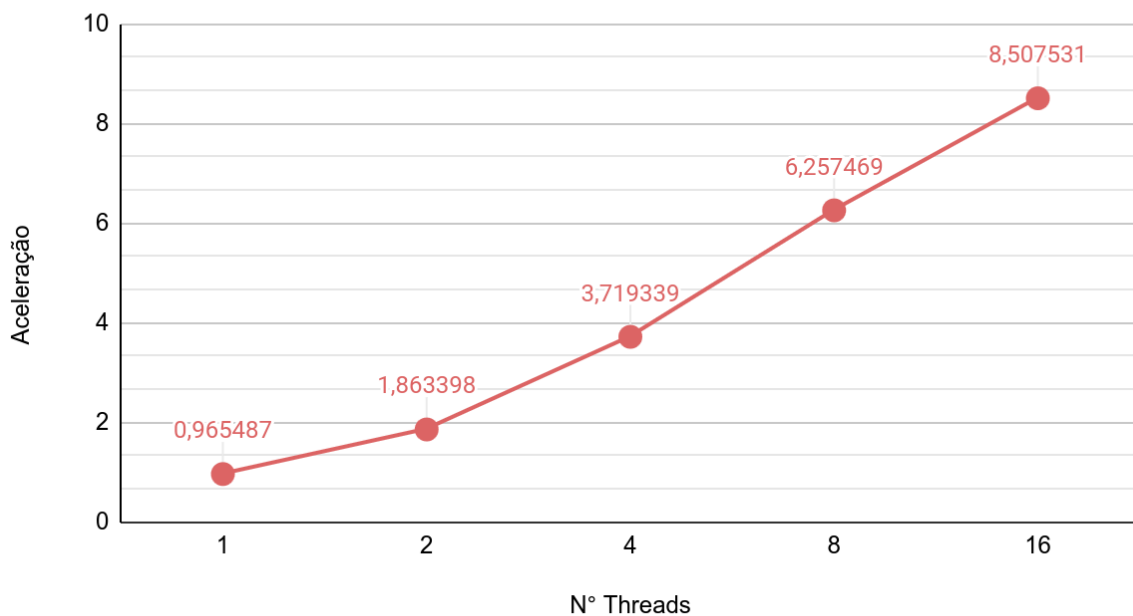
aceleração	eficiência	t_concorrente (s)	linhasMatriz	colunasMatriz	n° threads	t_sequencial (s)
0,965487	0,060343	0,514119	500	500	1	0,496375
1,863398	0,116462	0,264899	500	500	2	0,493613
3,719339	0,232459	0,133901	500	500	4	0,498025
6,257469	0,391092	0,079898	500	500	8	0,499957
8,507531	0,531721	0,058302	500	500	16	0,496005
0,967777	0,060486	4,114508	1000	1000	1	3,981926
1,895967	0,118498	2,120647	1000	1000	2	4,020676
3,693445	0,23084	1,074546	1000	1000	4	3,968777
7,277596	0,45485	0,54566	1000	1000	8	3,971092
8,857697	0,553606	0,451948	1000	1000	16	4,00322
0,963254	0,060203	34,570185	2000	2000	1	33,29988
1,869448	0,11684	17,771129	2000	2000	2	33,222194
3,712221	0,232014	8,993323	2000	2000	4	33,385203
7,339292	0,458706	4,568756	2000	2000	8	33,531435
9,672427	0,604527	3,459757	2000	2000	16	33,464244

Gerando então, os seguintes gráficos de eficiência e aceleração abaixo, que também podem ser encontrados neste [link](#)

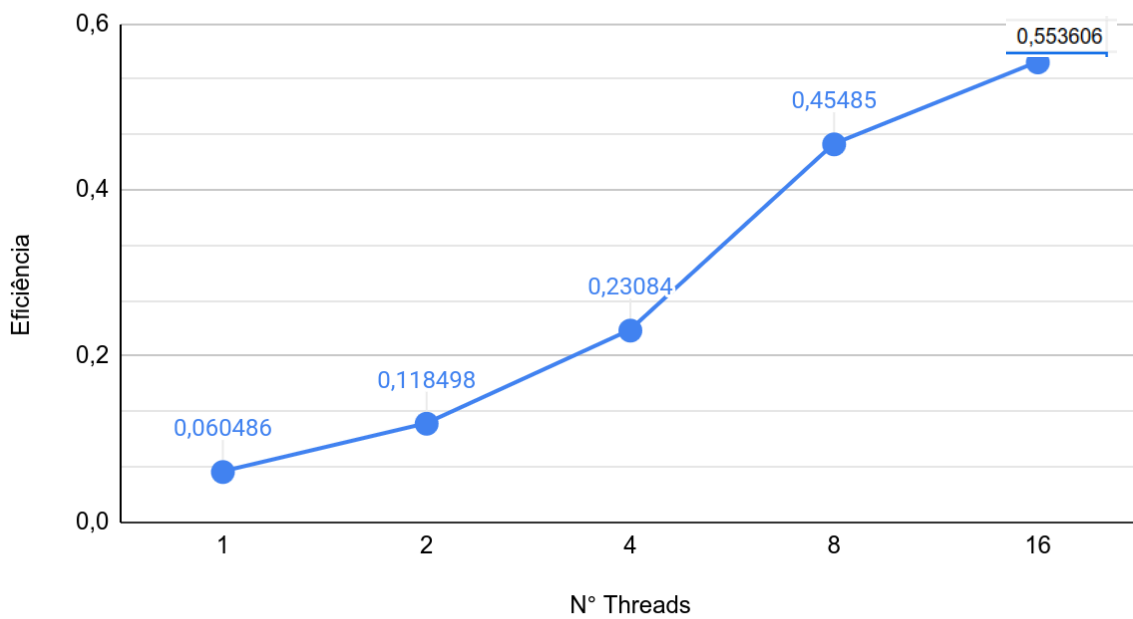
Eficiência - Matriz 500x500



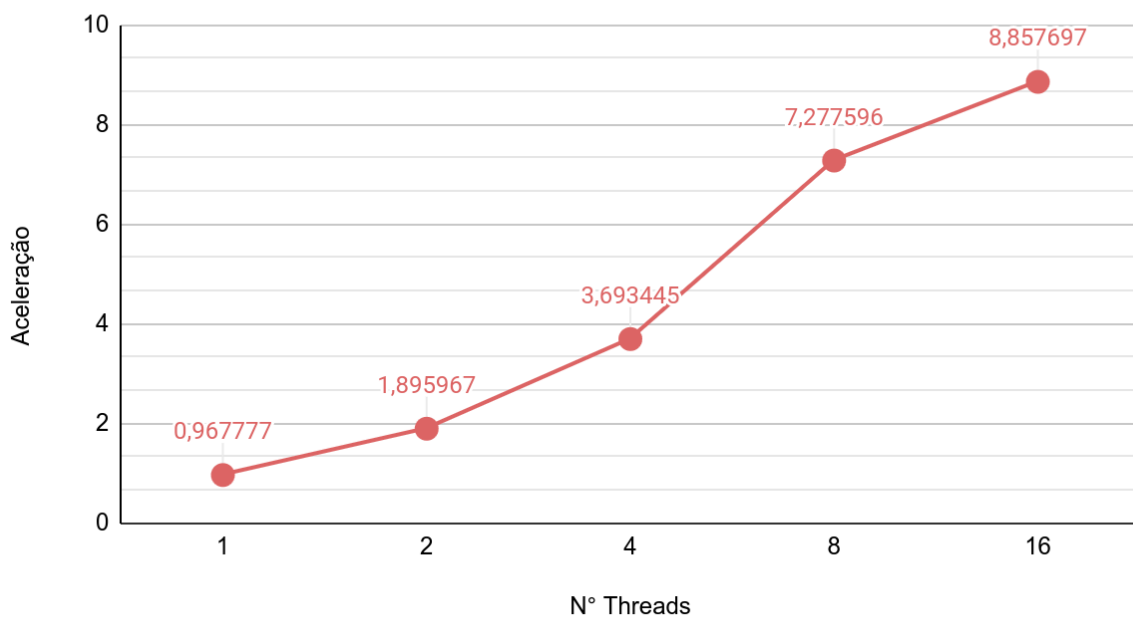
Aceleração - Matriz 500x500



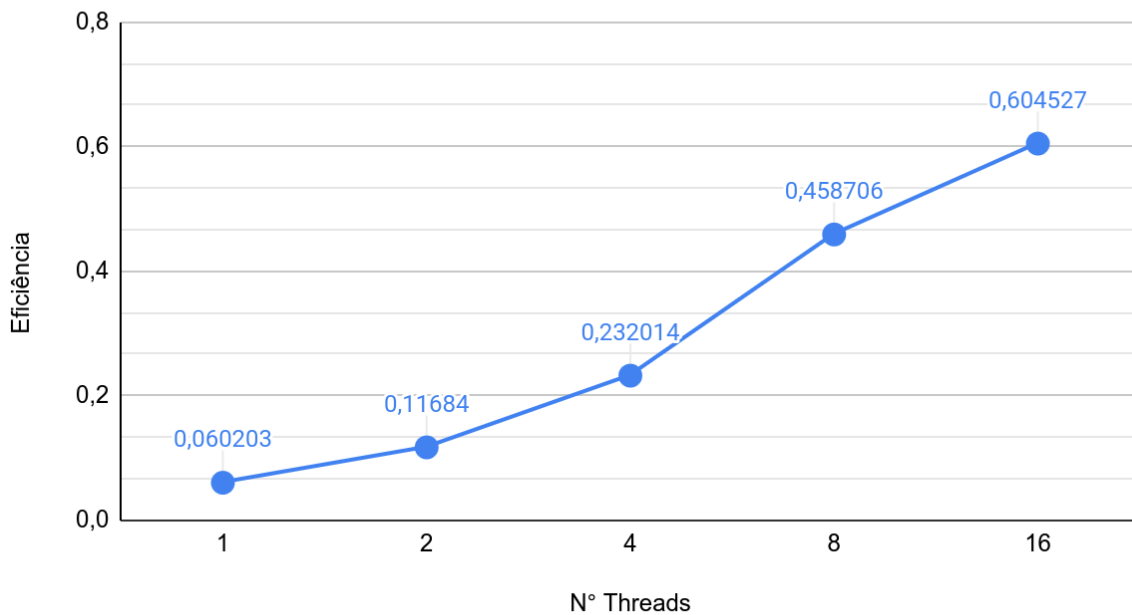
Eficiência - Matriz 1000x1000



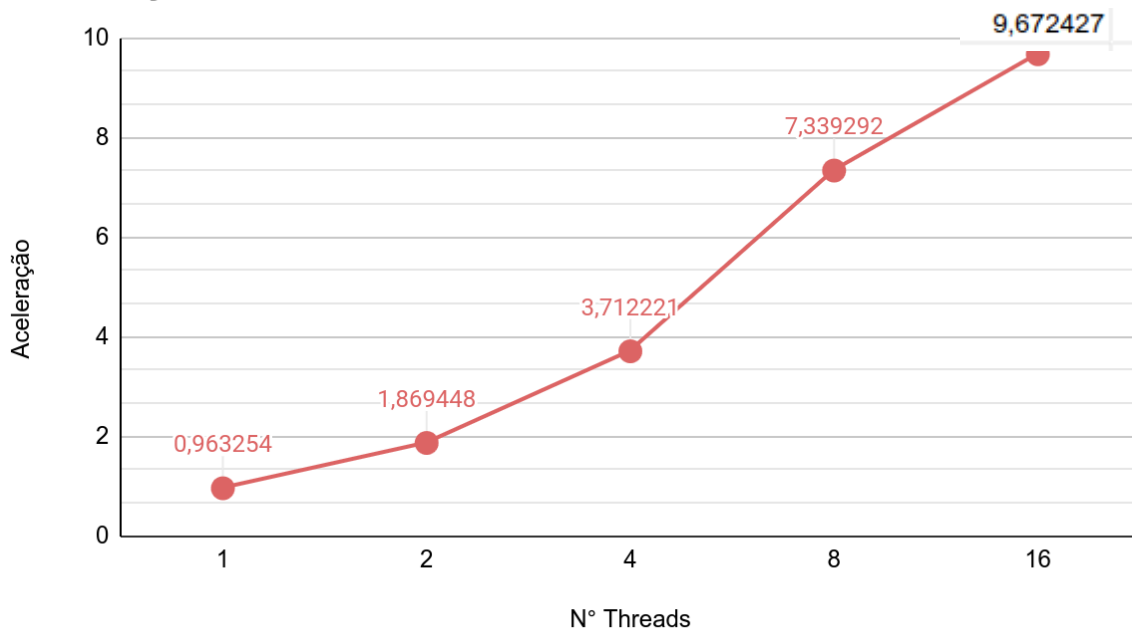
Aceleração - Matriz 1000x1000



Eficiência - Matriz 2000x2000



Aceleração - Matriz 2000x2000



Dadas as configurações de máquina:

- Processador: AMD Ryzen 7 5700G, com 8 núcleos e 16 threads lógicas
- Memória RAM: 16gb DDR4
- Placa de vídeo: AMD RX550

Pude observar que, se eu tentasse mais que 16 threads, não obteria melhoria em relação ao tempo, então 16 seria o limite para o melhor desempenho do programa concorrente, por ser o número de núcleos lógicos do processador utilizado.