

Configuração maquina: Intel i7-4500U com 2 núcleos e 4 threads e gráfico integrado , com 16gb de memoria RAM DDR3

Para desenvolvimento dessa atividade foi feita a geração de 6 matrizes aleatórias, duas de tamanho 500 x 500, duas de tamanho 1000 x 1000 e duas de tamanho 2000 x 2000 que denominamos cada arquivo binário que continha cada matriz com o nome 'matriz+tamanho+valor inteiro para identificar.bin' exemplificando 'matriz5001.bin', 'matriz5002.bin'. para os teste foi feita a multiplicação de matrizes usando dois programas, um que usa o método sequencial de processamento e outro que usa o método concorrente e ao executar os programas tínhamos como retorno novas matrizes em arquivos binários com identificação 3 para o arquivo gerado pelo código concorrente e 4 para o código sequencial e o tempo total de execução de cada processo como no exemplo abaixo para o processo sequencial

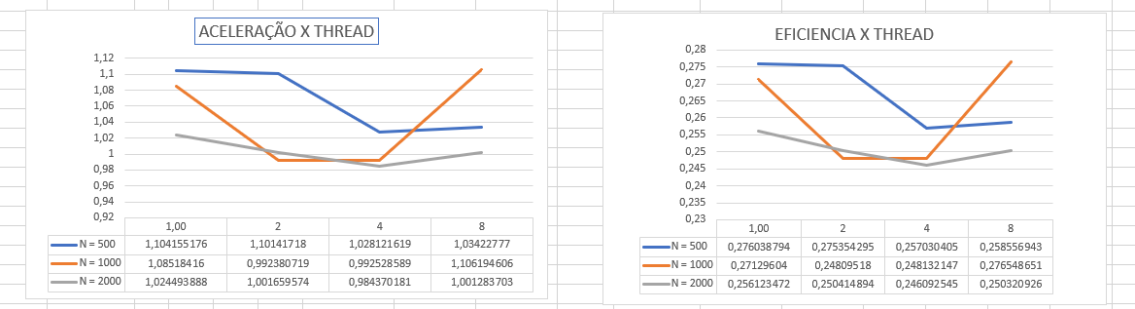
```
joaoa@DESKTOP-8VP102J:/mnt/c/Users/joao2/Desktop/Lab3$ gcc -o multiseri multmatseri.c timer.h -lm
joaoa@DESKTOP-8VP102J:/mnt/c/Users/joao2/Desktop/Lab3$ ./multiseri matriz5001.bin matriz5002.bin matriz5004.bin
Tempo total: 0.533152 segundos
joaoa@DESKTOP-8VP102J:/mnt/c/Users/joao2/Desktop/Lab3$ ./multiseri matriz10001.bin matriz10002.bin matriz10004.bin
Tempo total: 5.697393 segundos
joaoa@DESKTOP-8VP102J:/mnt/c/Users/joao2/Desktop/Lab3$ ./multiseri matriz20001.bin matriz20002.bin matriz20004.bin
Tempo total: 105.937784 segundos
```

Para o programa concorrente temo como comando de entrada para o arquivo de multiplicação de matrizes concorrente: <nome do arquivo de execução pôs compilação> <arquivo da primeira matriz de entrada> <arquivos da segunda matriz de entrada> <arquivo de saída das matrizes> <número de threads>. Para os testes do programa concorrente foi feita 3 execuções para cada tamanho de matriz e valor de thread como

mostrado na imagem abaixo cada linha representa os testes onde a coluna N representa o tamanho da matriz as colunas T1,T2,T3 representa os tempos em segundos(os tempos foram calculados com 6 casas decimais ,mas o Excel por padrão arredonda na demonstração visual da tabela) o tempo médio dos 3 testes , a coluna tempo_serial é o tempo dos testes de multiplico como na imagem anterior para o programa de multiplicação serial , a coluna Thread é o número de thread utilizadas por teste , a coluna Aceleração é feita pela formula tempo serial dividido pelo tempo médio , a coluna Eficiência é formada pela formula da aceleração dividido por p , sendo p o número de processadores , que no nosso caso é 4

N	T1	T2	T3	Tempo Medio	Aceleração	Eficiencia	Thread	tempo_serial
500	0,48	0,48	0,49	0,482859667	1,104155176	0,276038794	1,00	0,533152
500	0,49	0,49	0,48	0,48406	1,10141718	0,275354295	2	0,533152
500	0,48	0,59	0,49	0,518569	1,028121619	0,257030405	4	0,533152
500	0,49	0,57	0,49	0,515507333	1,03422777	0,258556943	8	0,533152
1000	4,89	5,28	5,58	5,250162333	1,08518416	0,27129604	1,00	5,70
1000	5,40	5,88	5,94	5,741136333	0,992380719	0,24809518	2	5,70
1000	5,45	5,82	5,95	5,740281	0,992528589	0,248132147	4	5,70
1000	5,02	5,37	5,07	5,150443667	1,106194606	0,276548651	8	5,70
2000	102,93	103,16	104,13	103,4049937	1,024493888	0,256123472	1,00	105,94
2000	104,48	107,80	105,01	105,7622637	1,001659574	0,250414894	2	105,94
2000	105,14	107,98	109,74	107,619863	0,984370181	0,246092545	4	105,94
2000	106,26	105,41	105,74	105,8019657	1,001283703	0,250320926	8	105,94

Com os valores coletados presentes na tabela formamos dois gráficos , um da relação aceleração em função das threads e outro da eficiência em função das threads como visto logo a baixo;



O que pode ser concluído ao observar levando em consideração as configurações de processador da máquina de teste que existe uma relação direta entre aceleração e eficiência quanto maior a aceleração maior ser a eficiência do processo.