

**MAC0219 - INTRODUÇÃO À COMPUTAÇÃO PARALELA E  
DISTRIBUÍDA  
RELATÓRIO MINIEP5**

Nome: **Beatriz Viana Costa**

Número USP: **13673214**

**Questão 1:**

n	matrix_dgemm_0	matrix_dgemm_1
512	0.29683s	0.04241s
1024	2.19390s	0.40785s
2048	75.28621s	6.76807s
4096	674.94653s	57.80699s

TABELA 1. Tabela com o tempo de execução em segundos de cada implementação para matrizes de dimensão  $n \times n$ .

*Sim, houve melhora significativa no tempo de execução entre as duas implementações. Isso ocorreu pelo fato de na nova implementação não percorrer cada linha e cada coluna  $n$  vezes, e sim percorrer cada uma calculando a multiplicação dos elementos atuais das matrizes  $A$  e  $B$ , e a medida que os iteradores são incrementados, o valor de cada elemento na matriz  $C$  também é incrementado.*

**Solução:**

*Basicamente a mudança realizada no código foi a modificação da ordem dos laços, na versão original (*matrix\_dgemm\_0*) eram realizados primeiro os laços que iteravam as linhas e colunas de  $A$  e  $B$  respectivamente, e somente depois era iterado cada elemento das respectivas linhas e colunas. Na modificação, o último laço foi colocado como sendo o primeiro, e somente depois vieram os iterando das linhas e colunas, assim poderíamos realizar os cálculos necessários sem haver a necessidade de voltarmos às mesmas linhas e colunas.*

**Questão 2:**

*Seguem os resultados da execução dos mesmos testes em uma máquina diferente.*

n	matrix_dgemm_0	matrix_dgemm_1
512	0.23754s	0.07456s
1024	2.58637s	0.68540s
2048	55.28769s	5.87436s
4096	692.65789s	51.25938s

TABELA 2. Tabela com o tempo de execução em segundos de cada implementação para matrizes de dimensão  $n \times n$ .

*É possível notar que na versão *matrix\_dgemm\_0* no caso em que  $N = 2048$ , houve diferença no tempo de execução em uma máquina diferente. Contudo, nos demais casos observados a*

*diferença de tempo de execução foi pequena.*