

**ATIVIDADE 2 - CONCORRÊNCIA EM MEMÓRIA COMPARTILHADA
UTILIZANDO PTHREADS E OPENMP**

Alunos: João Victor de Mesquita Cândido dos Santos, RA:102028 , Raphael Ribeiro Faria, RA: 104120

Unidade Curricular: Programação Concorrente e Distribuída

Docente: Prof. Dr. Álvaro Luiz Fazenda

Problema 1 - N-Body

Pthreads:

Número de Threads	Tempo medido
1	17886 ms
2	8957 ms
4	4536 ms
8	4522 ms

OpenMP:

Número de Threads	Tempo medido
1	24966 ms
2	24261 ms
4	24005 ms
8	22742 ms

Problema 2 - Métricas

a,b e c) Para a resolução dos três tópicos do exercício se utilizou a Lei de *Amdahl* :

$$speedup = \frac{1}{[s + \frac{(1-s)}{p}]} \quad (1)$$

onde s é a fração serial de execução.

Já para o cálculo dos valores práticos de speedup das *threads* a seguinte relação:

$$speedup = \frac{\text{Tempo total com 1 Thread}}{\text{Tempo total com n Threads}} \quad (2)$$

Os tempos práticos obtidos foram:

- 1 Thread:
 - tempo total= 21578 milissegundos;
- 2 Threads:
 - tempo total= 21294 milissegundos;
- 4 Threads:
 - tempo total= 20756 milissegundos;

Para calcular o *speedup* teórico de acordo com a equação (1), foi verificado através de testes de execução para 1 *thread* que a porcentagem de código necessária para realizar operações em série é bem pequena e com valor de 0.0324% do código, tal valor corresponde ao termo s da equação e para o cálculo dos valores de *speedup* prático foi utilizada a equação (2) como já dito anteriormente. Munidos de tais equações foi possível o preenchimento da Tabela 1 abaixo.

Processadores	Speedup teórico	Speedup prático	Eficiência
1	1	1	99.9676%
2	1	1.01333709	99.9484%
4	3.996115775	1.039603006	99.9567%
8	7.981897057	-	-
16	15.92261609	-	-
32	31.68178812	-	-
64	62.71976417	-	-

Tabela 1 - *Speedup* do algoritmo com função *random* original.

d) É possível analisar através dos dados obtidos que a utilização de *Threads* aumenta o valor de *Speedup* mas não significativamente como era esperado ao se comparar com o valor de *speedup* teórico, como por exemplo o valor de *speedup* esperado para 4 processadores é quase 4 vezes a mais do que o valor de *speedup* prático. Isso de fato ocorre pois a lei de *Amdahl* não leva em conta algumas questões como por exemplo a sincronização, referência comum de tempo e tempo de mensagem entre os processos que estão sendo executados de forma paralela, tais questões influenciam diretamente no tempo de processamento.

e, f)

Os tempos práticos obtidos foram:

- 1 Thread:
 - tempo total= 61781 milissegundos;
- 2 Threads:
 - tempo total= 61762 milissegundos;
- 4 Threads:
 - tempo total= 60969 milissegundos;

Semelhante ao que foi feito antes da alteração da função *random* foram utilizadas as equações (1) e (2) que possibilitaram encontrar os valores de *speedup* teóricos e práticos, dado que o valor *s* na equação (1) foi medido de maneira prática o que resultou numa parcela 67.2731% do código. Munidos dos dados foi possível preencher a Tabela 2 abaixo.

Processadores	Speedup teórico	Speedup prático	Eficiência
1	1	1	32.7269%
2	1.196098566	1.000307633	32.6641%
4	1.324194186	1.013318244	31.6620%
8	1.401267481	-	-
16	1.4443365278	-	-
32	1.464218548	-	-
64	1.475264437	-	-

Tabela 2 - *Speedup* do algoritmo com função *random* modificada.

Dos dados obtidos nessa segunda parte foi possível perceber que a função *random* por não ser mais recursiva causou uma piora no tempo de execução do algoritmo, o tempo praticamente triplicou em cada uma das mudanças nos valor de processadores, isso de fato acarreta numa maior complexidade de tempo do código fazendo com que ela deixasse de ser constante para uma complexidade que depende do valor *n* recebido por parâmetro. A recursividade antes da alteração da função *random* era o motivo pelo qual não era possível paralelizar esse pedaço do código pois ela gerava uma situação de *data race* onde todas as *threads* buscavam o mesmo recurso ao mesmo tempo e isso de fato impossibilita a paralelização.

Descrição do equipamento utilizado:

1. Modelo do Processador: AMD Phenom(tm) II X4 B93 Processor, CPU - 1,6GHz, 4 núcleos físicos, possui *hyperthreading*;
2. Memória Principal: 3GB;
3. Sistema Operacional: Linux 64 bit. distro Linux Mint 18.2 "Sonya";
4. Compilador: GCC 5.4.0 20160609.