

CURSO: Análise e Desenvolvimento de Sistemas

DISCIPLINA: Tópicos em Computação de Alto Desempenho

PROFESSOR: Msc. Luiz Mário Lustosa Pascoal

**TRABALHO N1 – COMPARAÇÃO COM ALGORITMOS DE ORDENAÇÃO
PARALELOS COM O OPENMP E JAVA**

- Mesmos grupos da Fábrica de Software.
- Ao implementar o problema descrito abaixo, compacte os arquivos desenvolvidos (código-fonte e relatório) no formato .zip e envie na área criada no Google Classroom da disciplina.

Objetivo do Trabalho: Implementar o algoritmo MergeSort paralelo usando as diretivas do OpenMP. Em seguida compare-os com as suas respectivas implementações em Java.

Para tal, cada grupo deve adaptar o algoritmo sequencial do MergeSort em anexo na tarefa e alterá-lo ao inserir as diretivas de paralelismo do OpenMP vistas em sala de aula.

O código deve ser testado para diversas instâncias de tamanho de vetores diferentes, onde o tamanho deve ser aumentado na ordem de $2^n * 1000$. Caso o valor de aumento seja 12, o código deve ser testado para os seguintes tamanhos:

- $2^0 = 1.000$;
- $2^1 = 2.000$;
- $2^2 = 4.000$;
- $2^3 = 8.000$;
- $2^4 = 16.000$;
- $2^5 = 32.000$;
- $2^6 = 64.000$;
- $2^7 = 128.000$;
- $2^8 = 256.000$;
- $2^9 = 512.000$;
- $2^{10} = 1.024.000$;
- $2^{11} = 2.048.000$;
- $2^{12} = 4.096.000$;

Compare o tempo de ordenação do algoritmo Sequencial com o algoritmo Paralelo na linguagem C e verifique se há melhoria de desempenho e tempo (principalmente para tamanhos grandes – sugestão de tamanho final: **25**); Por fim, também compare com os algoritmos sequencial e paralelo na linguagem JAVA enviados também em anexo;

Exemplo de saída para um aumento final de tamanho 16 com comparativo de tempo:

1000 elements =>	3 ms
2000 elements =>	4 ms
4000 elements =>	4 ms
8000 elements =>	30 ms
16000 elements =>	3 ms
32000 elements =>	6 ms
64000 elements =>	14 ms
128000 elements =>	24 ms
256000 elements =>	31 ms
512000 elements =>	74 ms
1024000 elements =>	208 ms
2048000 elements =>	217 ms
4096000 elements =>	355 ms
8192000 elements =>	771 ms
16384000 elements =>	1391 ms
32768000 elements =>	3058 ms

Ao final, cada grupo deverá realizar a escrita de um relatório descritivo dos resultados obtidos apresentando justificativas plausíveis que permitam a análise de desempenho obtidos em cada código implementado.