

Atividade 2 - Concorrência em memória compartilhada utilizando Pthreads, JavaThreads e OpenMP

Programação Concorrente e Distribuída
2^o Semestre de 2017

Entrega: Arquivo compactado contendo: Arquivo em PDF com relatório das medidas de tempo e especificações da arquitetura computacional utilizada, e o arquivo referente ao código-fonte desenvolvido.

Problema 1 - Log10

Deve-se escrever uma programa para calcular o \log_{10} de um vetor de números inteiros. Deve-se lembrar que a função intrínseca \log_{10} tem a seguinte sintaxe:

```
double = log10(double)
```

Crie três códigos-fonte distintos, em linguagem **C com Pthreads**, em **JavaThreads** e em linguagem **C com OpenMP**.

Meça o tempo de execução do trecho que calcula o novo vetor utilizando vetores de tamanho 20000000 (2×10^7), iniciados de forma aleatória, para 1, 2, 4 e 8 *threads*. Crie uma tabela com os tempos medidos variando o número de *threads* e a versão do programa.

Ao executar o programa os dados de entrada e saída devem ser:

Entrada

Um número inteiro longo que determina o tamanho do vetor.

Saída

Maior valor encontrado no novo vetor e o tempo de execução, em milissegundos, do trecho que calcula este novo vetor.

Exemplo de Entrada	Exemplo de Saída
20000000	7.30103 158

Problema 2 - N-Body

Na física, o problema do *N-Body* (https://en.wikipedia.org/wiki/N-body_problem) consiste em simular a interação gravitacional entre **N** partículas (corpos) em um sistema e prever como o sistema evoluiria em um período de tempo. Nesta aplicação, a posição inicial e a massa de cada partícula são geradas aleatoriamente. A aplicação calculará as forças gravitacionais, as posições e a velocidade de cada partícula em cada etapa do tempo da simulação.

O código-fonte da versão serial está disponibilizada no moodle, entretanto, para o caso em questão, apenas uma iteração no tempo está sendo calculada para fins de simplificação.

Não há interesse em descobrir diferentes algoritmos para a computação da simulação *N-Body*. O foco consiste em se concentrar na obtenção de uma versão concorrente/paralela do código fornecido. Portanto, não é permitido alterar o método de computação usado neste problema.

Deve-se criar um programa concorrente em **linguagem C e OpenMP**, e medir o tempo de execução, em milissegundos, relativo ao trecho que calcula as forças e atualiza as posições das partículas. Crie uma tabela com os tempos medidos variando o número de *threads* em 1, 2, 4 e 8.

Entrada

Os dados de entrada contém apenas o valor de **N** que corresponde à quantidade de partículas a serem consideradas ($0 < \mathbf{N} \leq 2^{15}$). Deve-se utilizar **N** (quantidade de partículas) igual à 25000 para as medidas de tempo.

Saída

Os dados de saída contém o valor da força máxima aplicada a uma determinada partícula, o avanço no tempo referente a simulação (variável `sim_t`) e o tempo de execução, em milissegundos, do trecho que executa a simulação.

Exemplo de Entrada	Exemplo de Saída
25000	1.33412e+18 0.001 22683