



SIMULADOR DE PROPAGAÇÃO DE VÍRUS

UTILIZANDO OPENMP E PROGRAMAÇÃO PARALELA

ESTRATÉGIA DE PARALELIZAÇÃO

Paralelização externa: Calcular várias probabilidades ao mesmo tempo.

Paralelização interna: Calcular uma probabilidade, mas internamente paralelizando o percorrimto dos loops.

EXPERIMENTOS REALIZADOS

Os experimentos realizados se basearam nas seguintes configurações

Número de threads: 1, 2 e 4

Tamanho da população: 64, 128 e 192

Número de experimentos: 150

Probabilidade: 0 a 100%

RESULTADOS OBTIDOS

Solução 1:

População: 64 Experimentos: 150 Probabilidades 0 a 100%

Single thread: 45 segundos

2 THREADS

Tempo: 45 segundos

Speedup: 1.00

4 THREADS

Tempo: 45 segundos

Speedup: 1.00

RESULTADOS OBTIDOS

Solução 1:

População: 128 Experimentos: 150 Probabilidades 0 a 100%

Single thread: 338 segundos

2 THREADS

Tempo: 234 segundos

Speedup: 1.44

4 THREADS

Tempo: 227 segundos

Speedup: 1.49

RESULTADOS OBTIDOS

Solução 1:

População: 192 Experimentos: 150 Probabilidades 0 a 100%

Single thread: 1023 segundos

2 THREADS

Tempo: 675 segundos
Speedup: 1.51

4 THREADS

Tempo: 658 segundos
Speedup: 1.55

RESULTADOS OBTIDOS

Solução 2:

População: 64 Experimentos: 150 Probabilidades 0 a 100%

Single thread: 39 segundos

2 THREADS

Tempo: 33 segundos

Speedup: 1.18

4 THREADS

Tempo: 31 segundos

Speedup: 1.25

RESULTADOS OBTIDOS

Solução 2:

População: 128 Experimentos: 150 Probabilidades 0 a 100%

Single thread: 268 segundos

2 THREADS

Tempo: 166 segundos

Speedup: 1.61

4 THREADS

Tempo: 151 segundos

Speedup: 1.77

RESULTADOS OBTIDOS

Solução 2:

População: 192 Experimentos: 150 Probabilidades 0 a 100%

Single thread: 864 segundos

2 THREADS

Tempo: 514 segundos

Speedup: 1.68

4 THREADS

Tempo: 453 segundos

Speedup: 1.90

CONCLUSÃO SOBRE OS RESULTADOS

Paralelização externa:

Ao aumentar o tamanho da população, usar várias threads se tornou benéfico. Algo que foi percebido é que esse algoritmo consome bem mais memória que o com uma thread, pois é instanciada uma classe população para cada thread.

CONCLUSÃO SOBRE OS RESULTADOS

Paralelização interna:

Nos primeiros testes que foram feitos, usar várias threads não se mostrou benéfico. Mas, percebemos que, testar com 5000 repetições faz com que o overhead de inicialização das threads e as regiões críticas seja muito custoso, piorando o desempenho. Quando testamos com menos repetições, o speedup foi maior que 1.



SIMULADOR DE PROPAGAÇÃO DE VÍRUS

UTILIZANDO OPENMP E PROGRAMAÇÃO PARALELA

JOÃO GABRIEL SCHITLER, THALISSON FORTE