

Laboratório 3

Ambiente de desenvolvimento:

A atividade foi feita usando a ferramenta VsCode, no sistema operacional Windows, com as devidas instalações do gcc e MingW

Compilação e execução:

O programa foi compilado no Windows com:

gcc NomeDoPrograma.c -o NomeDoPrograma

e executado com:

NomeDoPrograma <N> <NTHREADS>

Onde:

***N** é a dimensão do vetor;*

***NTHREADS** é o número de Threads utilizadas;*

Medições e Avaliação de Desempenho:

A aplicação foi executada diversas vezes (pelo menos 5) para 2 e 4 Threads e também considerando os tamanhos dos vetores.

Os valores dos tempos de execução podem ser vistos mais adiante no relatório

Os valores na tabela dizem respeito ao ganho de desempenho estimado considerando a média dos tempos calculados

Tamanho do vetor	Número de Threads	
	2	4
10 ⁵	0.795331	0.4016634
10 ⁷	1.61424	1.97436
10 ⁹	1.90664	1.84037

10^5	2 threads
Tempo sequencial: 0.000469 Tempo concorrente: 0.000570 Ganho de desempenho: 0.821704	
Tempo sequencial: 0.000519 Tempo concorrente: 0.000600 Ganho de desempenho: 0.866077	
Tempo sequencial: 0.000407 Tempo concorrente: 0.000576 Ganho de desempenho: 0.706669	
Tempo sequencial: 0.000492 Tempo concorrente: 0.000591 Ganho de desempenho: 0.831641	
Tempo sequencial: 0.000432 Tempo concorrente: 0.000576 Ganho de desempenho: 0.750564	

10^5	4 threads
Tempo sequencial: 0.000497 Tempo concorrente: 0.001134 Ganho de desempenho: 0.437880	
Tempo sequencial: 0.000508 Tempo concorrente: 0.000753 Ganho de desempenho: 0.674944	
Tempo sequencial: 0.000482 Tempo concorrente: 0.002634 Ganho de desempenho: 0.183130	
Tempo sequencial: 0.000410 Tempo concorrente: 0.000820 Ganho de desempenho: 0.500671	
Tempo sequencial: 0.000524 Tempo concorrente: 0.002473 Ganho de desempenho: 0.211692	

10^7	2 threads
Tempo sequencial: 0.045962 Tempo concorrente: 0.029884 Ganho de desempenho: 1.538000	
Tempo sequencial: 0.046841 Tempo concorrente: 0.027304 Ganho de desempenho: 1.715528	
Tempo sequencial: 0.046211 Tempo concorrente: 0.030785 Ganho de desempenho: 1.501090	
Tempo sequencial: 0.051762 Tempo concorrente: 0.032368 Ganho de desempenho: 1.599135	
Tempo sequencial: 0.045910 Tempo concorrente: 0.026731 Ganho de desempenho: 1.717441	

10^7	4 threads
Tempo sequencial: 0.050032 Tempo concorrente: 0.025780 Ganho de desempenho: 1.940687	
Tempo sequencial: 0.061959 Tempo concorrente: 0.025760 Ganho de desempenho: 2.405207	
Tempo sequencial: 0.046690 Tempo concorrente: 0.024034 Ganho de desempenho: 1.942680	
Tempo sequencial: 0.045316 Tempo concorrente: 0.024972 Ganho de desempenho: 1.814644	
Tempo sequencial: 0.045592 Tempo concorrente: 0.025779 Ganho de desempenho: 1.768567	

10⁹	2 threads
Tempo sequencial: 10.020799 Tempo concorrente: 3.125608 Ganho de desempenho: 3.206032	
Tempo sequencial: 5.085926 Tempo concorrente: 3.936593 Ganho de desempenho: 1.291961	
Tempo sequencial: 5.836211 Tempo concorrente: 3.237445 Ganho de desempenho: 1.802721	
Tempo sequencial: 5.013096 Tempo concorrente: 2.977750 Ganho de desempenho: 1.683518	
Tempo sequencial: 6.357363 Tempo concorrente: 4.104238 Ganho de desempenho: 1.548975	

10⁹	4 threads
Tempo sequencial: 5.186259 Tempo concorrente: 2.360713 Ganho de desempenho: 2.196904	
Tempo sequencial: 5.152219 Tempo concorrente: 2.448692 Ganho de desempenho: 2.104070	
Tempo sequencial: 5.395550 Tempo concorrente: 2.967100 Ganho de desempenho: 1.818459	
Tempo sequencial: 5.090603 Tempo concorrente: 3.168868 Ganho de desempenho: 1.606442	
Tempo sequencial: 5.752325 Tempo concorrente: 3.897302 Ganho de desempenho: 1.475976	