

Otimização do Programa da Aula 5

Desenvolvida por: Raul A. Gonzalez Augusto, RA.: 211023698

Objetivo: Utilizar a ferramenta Valgrind de levantamento de perfis de execução de programas, levantando hotspots para otimizar, e modificar o programa tentando melhorar o seu desempenho.

Computador Usado:

Avell High Performance A52 LIV

- Processador: Intel® Core™ i5-10300H CPU @ 2.50GHz
 - Cores: 4
 - Threads: 8
 - Frequência Máxima: 4.50 GHz
 - Cache: 8 MB Intel® Smart Cache
 - Bus Speed: 8 GT/s
- Placa de Vídeo: NVIDIA GeForce GTX 1650 Ti e Intel® UHD Graphics
- RAM: 16 GB DDR4-2666 Dual Channel
- Armazenamento: 500 GB nvme m.2 SSD 2GB/s E 1 Tera HDD 5400 rpm
- Sistema Operacional: Windows 10 pro 64bits

Programas

Compilado no gcc maquina virtual Linux.

Executando o programa original

A letra A apareceu 341710

Tempos:

1.921066

1.910217

1.915676

1.943044

1.903151

Média: 1.9186308

Primeira Otimização

A letra A apareceu 341710

Tempos:

1,975389

2,174811

3,331097

3,345616

3,353513

Média: 2,8360852

Perda de -47,81% de performance

A strlen é a função com maior impacto, 75.53% do tempo

Segunda otimizacao

A letra A apareceu 341710

Tempos:

1,014926

1,046984

1,028386

1,027261

1,032307

Média: 1,0299728

Ganho de 46,31% de performance

A função contaLetra é a mais demorada com 99.99%

Terceira otimização

A letra A apareceu 341710

Tempos:

1,065781

1,099734

1,069726

1,068924

1,104303

Média: 1,0816936

Ganho de 43,62% de performance

Ultima otimização

A letra A apareceu 341710

Tempos:

0,262404

0,261137

0,264669

0,270461

0,268155

Média: 0,2653652

Speedup de 7,23x

Conheço a instrução `mov rdi, %[tudo]`, que move o valor do endereço `tudo` para o `rdi` e `add rdi, 1`, que adiciona 1 no `rdi`.