

Relatório Laboratório 1 – Parte 3:

Análise Comparativa de Desempenho

1. Objetivo

O seguinte relatório tem como objetivo realizar uma análise comparativa de desempenho entre a implementação manual do algoritmo crc feita pelo grupo e a implementação pronta presente na biblioteca indicada pelo professor.

As métricas comparativas analisadas foram o tempo de execução e o pico do uso de memória de ambas as implementações, executadas na máquina explicitada, os testes foram feitos para os tamanhos de 1500, 3000, 6000 e 16000 bytes.

2. Máquina Utilizada (Ambiente de Teste)

Processador: 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz

Núcleos / Threads: 4 / 8

RAM: 4 GB

SO: Microsoft Windows 11 Home Single Language

3. Metodologia

As implementações manual e da biblioteca foram executadas para os tamanhos de 1500, 3000, 6000 e 16000 bytes, utilizando bits aleatórios, para cada execução o tempo foi medido utilizando `time.perf_counter()`, como também foram medidos os picos de memória utilizando `tracemalloc`.

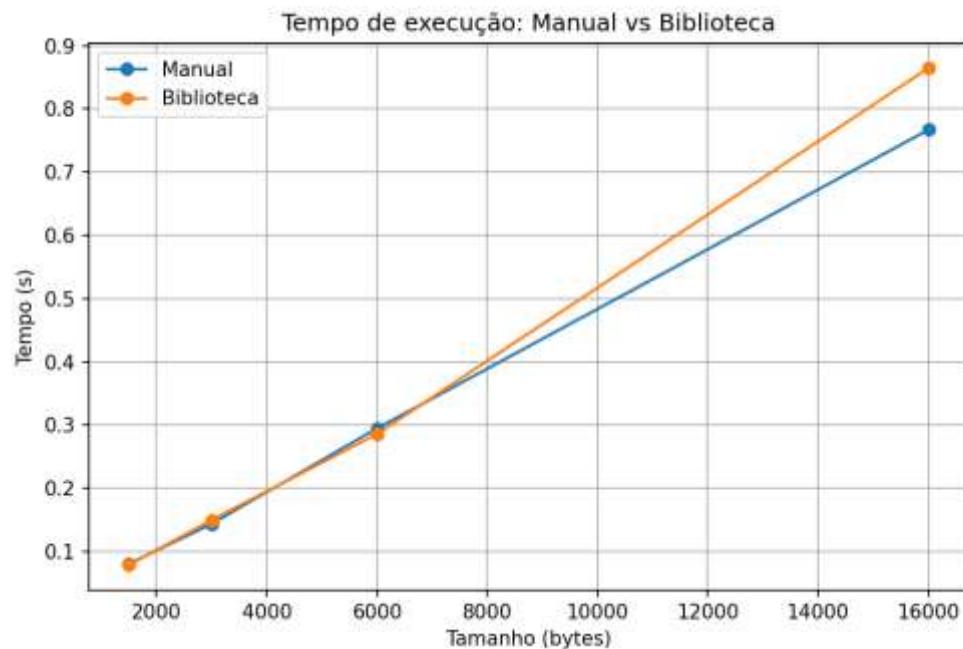
4. Resultados

4.1 Tabela Comparativa

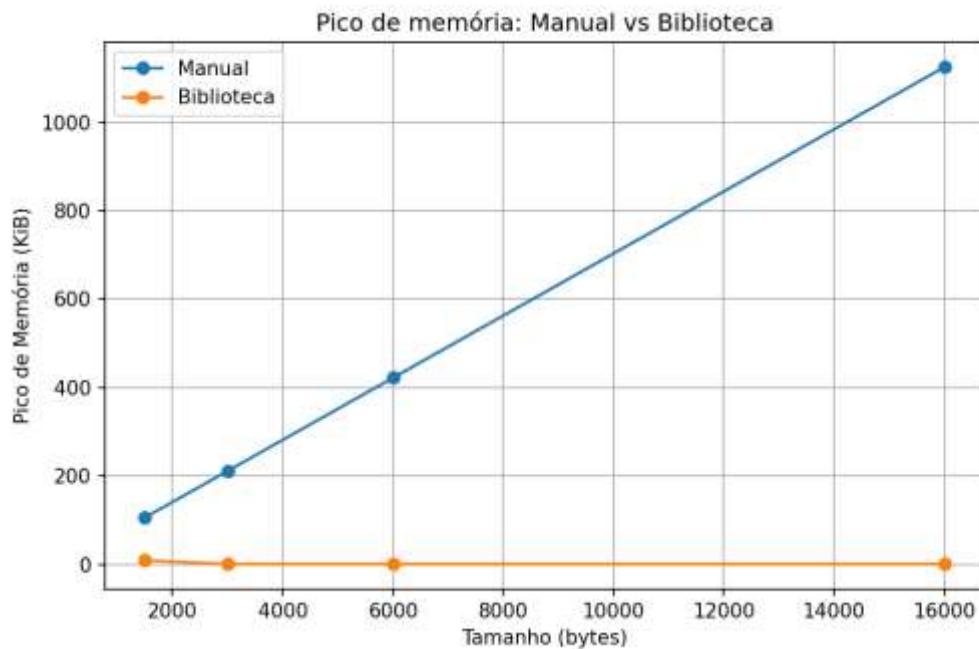
Tamanho (bytes)	Tempo Manual	Tempo Biblioteca	Pico Mem. Manual (KiB)	Pico Mem. Lib (KiB)
1500	0.080	0.078	105.758	8.756
3000	0.143	0.149	211.250	0.675
6000	0.294	0.285	422.188	0.675
16000	0.766	0.864	1125.313	0.675

4.2 Gráficos

4.2.1 Tempo x Tamanho



4.2.2 Pico Memoria x Tamanho



5. Análise e Discussão

5.1 Tempo

Os dados observados indicam que tanto a implementação manual realizada pelo grupo quanto a implementação da biblioteca possuem um tempo de execução da mesma ordem de grandeza, mostrando boa eficiência temporal da implementação manual, uma possível motivação é que tanto a biblioteca quanto a implementação do grupo utilizam python puro, tendo assim tempos no geral similares.

5.2 Memória

Os dados obtidos evidenciam que a implementação manual tem um pico de memória expressivamente maior que a implementação da biblioteca, tempo um aumento conforme o tamanho da mensagem, as implementações possuem ordens de grandeza diferentes, possivelmente está ineficiência da implementação manual vem da alocação temporária de muitas estruturas como listas e strings. Por outro lado, a biblioteca mantém um pico de memória baixo praticamente constante.

6. Conclusão

Observados os dados obtidos podemos concluir que a implementação manual cumpriu com o quesito de corretude, sendo também uma ótima escolha quando um dos requisitos é baixo tempo de execução para mensagens grandes, porém pecou na utilização de memória, tendo picos que crescem linearmente com o tamanho da mensagem, sendo relativamente ineficiente quando comparado com a implementação da biblioteca indicada pelo professor que possui um tempo de execução semelhante e por vezes menor, como também picos de memória praticamente constantes.