

# Análise de Desempenho na Implementação de Regressão Linear em C

Pablo Augusto Matos da Silva  
Matrícula 2022015139

14 de março de 2025

## Resumo

Esta atividade apresenta um experimento comparando o desempenho de diferentes abordagens para implementar uma regressão linear simples na linguagem C. Foram analisadas a chamada de funções, o uso de laços dentro das funções e a execução direta dos laços na função principal (main). Os resultados demonstram o impacto dessas estratégias na eficiência do código.

## 1 Introdução

A regressão linear simples é um método estatístico fundamental amplamente utilizado em diversas áreas. Sua implementação em linguagens de baixo nível, como C, pode ser realizada de diferentes formas, afetando o desempenho da execução. Este experimento busca comparar três abordagens distintas para avaliar a influência da estrutura do código na eficiência computacional.

## 2 Metodologia

O experimento consistiu na implementação da regressão linear simples utilizando três abordagens distintas:

- **Uso de funções com laços de repetição:** Cálculo da regressão linear modularizado em funções específicas, cada uma com um laço de repetição.

- **Uso de funções com um unico laço de repetição na main:** Implementação de um unico laço de repetição na função main chamando as demais funções para realização do cálculo.
- **Sem uso de funções e com um unico laço de repetição na main:** Execução dos cálculos diretamente dentro da função main e um unico laço de repetição e sem modularização.

Para medir a eficiência de cada abordagem, utilizamos a biblioteca `time.h` para registrar o tempo de execução. Cada implementação foi testada múltiplas vezes para obter uma média representativa dos tempos.

### 3 Resultados e Discussão

Os tempos médios de execução das três abordagens foram os seguintes:

Abordagem	Número de Elementos	Tempo Médio (s)
Uso de funções com laços de repetição	128	0.000002896
	256	0.000005573
	512	0.000011275
	1024	0.000022692
	2048	0.000045463
	4096	0.00009109
	8192	0.000182563
	16384	0.000365395
	32768	0.000730354
Uso de funções com um unico laço de repetição na main	128	0.000000722
	256	0.000001111
	512	0.000002273
	1024	0.000004504
	2048	0.000008988
	4096	0.000017854
	8192	0.000033877
	16384	0.000064543
	32768	0.000131592
Sem uso de funções e com um unico laço de repetição na main	128	0.000000457
	256	0.000000687
	512	0.000001225
	1024	0.000002293
	2048	0.000004444
	4096	0.000008939
	8192	0.000017715
	16384	0.000036121
	32768	0.000073608

Tabela 1: Tempos médios de execução para cada abordagem considerando diferentes números de elementos.

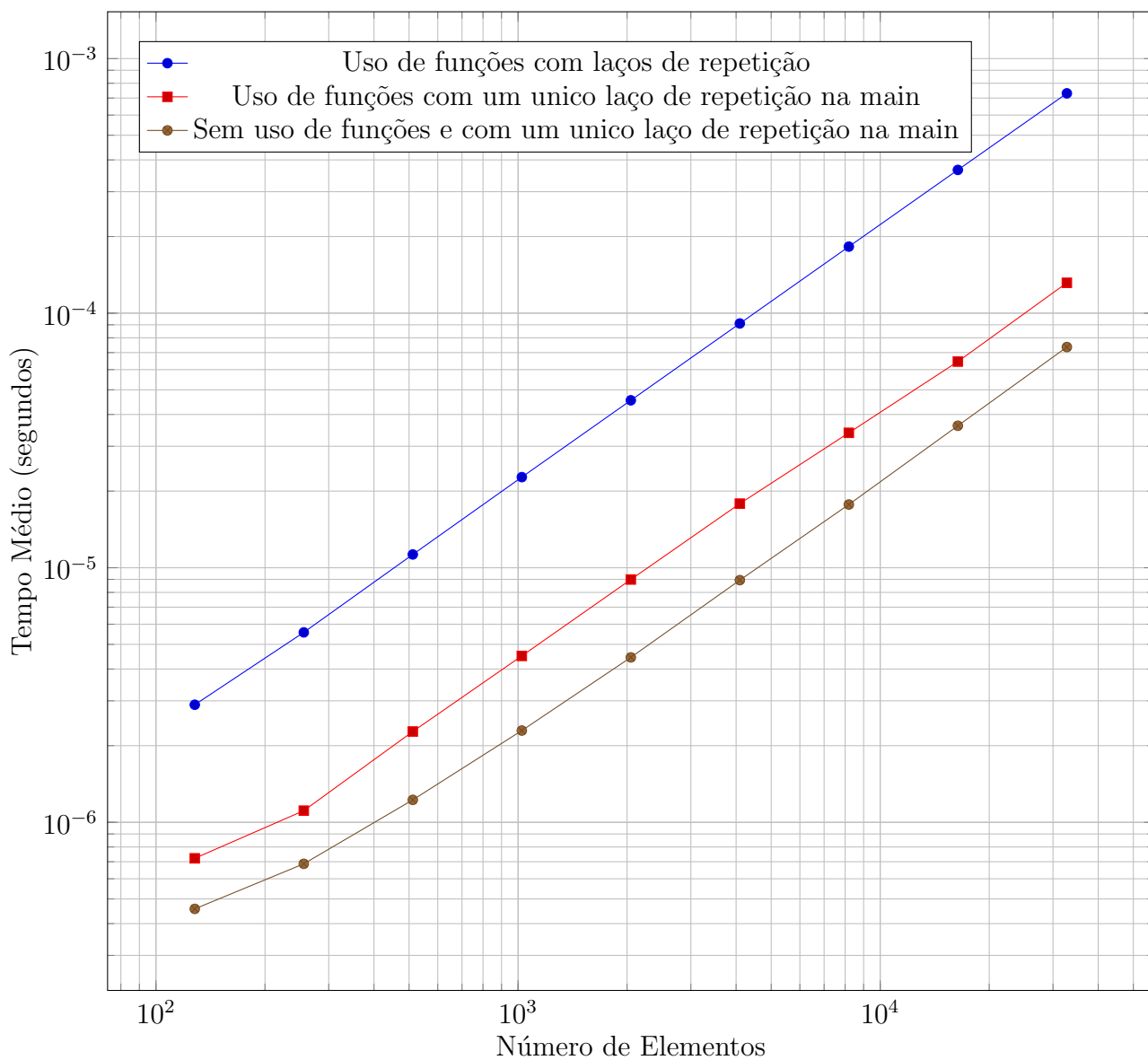


Figura 1: Comparação de Tempo Médio entre Diferentes Abordagens

Os resultados demonstram ganhos de até aproximadamente 89,92 % no tempo de execução ao utilizar um único laço de repetição e evitar a modularização por meio de funções para os cálculos. No entanto, é importante ressaltar que a modularização contribui significativamente para a legibilidade do código. Cabe ao desenvolvedor equilibrar a otimização do desempenho com a manutenção e clareza da implementação.

## 4 Detalhamento dos Testes

### 4.1 Hardware

Para realização dos experimentos foi utilizado um computador com as seguintes configurações:

- **Processador:** Intel® Core™ i7-12700H Alder Lake - 2.3GHz (Turbo Max 4.7GHz) - 24MB cache
- **Memória:** 16GB [2x 8GB - Dual Channel] Memória DDR4 (3200 MHz)

### 4.2 Software

- **Sistema Operacional:** Pop!\_OS 22.04 LTS 64 bits
- **Linguagem:** C17 (201710)
- **Compilador:** gcc (Ubuntu 11.4.0-1ubuntu1 22.04) 11.4.0

Para mais detalhes sobre o código, ele está disponível no repositório online do GitHub [1].

## 5 Conclusão

Este experimento permitiu avaliar o impacto da estrutura do código na eficiência de uma implementação em C. Observou-se que uma análise cuidadosa da complexidade assintótica dos algoritmos pode resultar em ganhos significativos de desempenho. Estudos futuros podem explorar outras otimizações, como o uso de paralelismo e diferentes estratégias de alocação de memória, ampliando as possibilidades de melhoria na eficiência computacional.

## Referências

- [1] Pablo Augusto Matos da Silva. *Código-fonte do experimento de regressão linear em C*. GitHub, 2025. Disponível em: <https://github.com/pabloaugmatrix/programacaoparalela/tree/main/atividade>.