

Relatório: Funções Hash



Gabriel Murakami Alves

11071916

Mauá, 2021

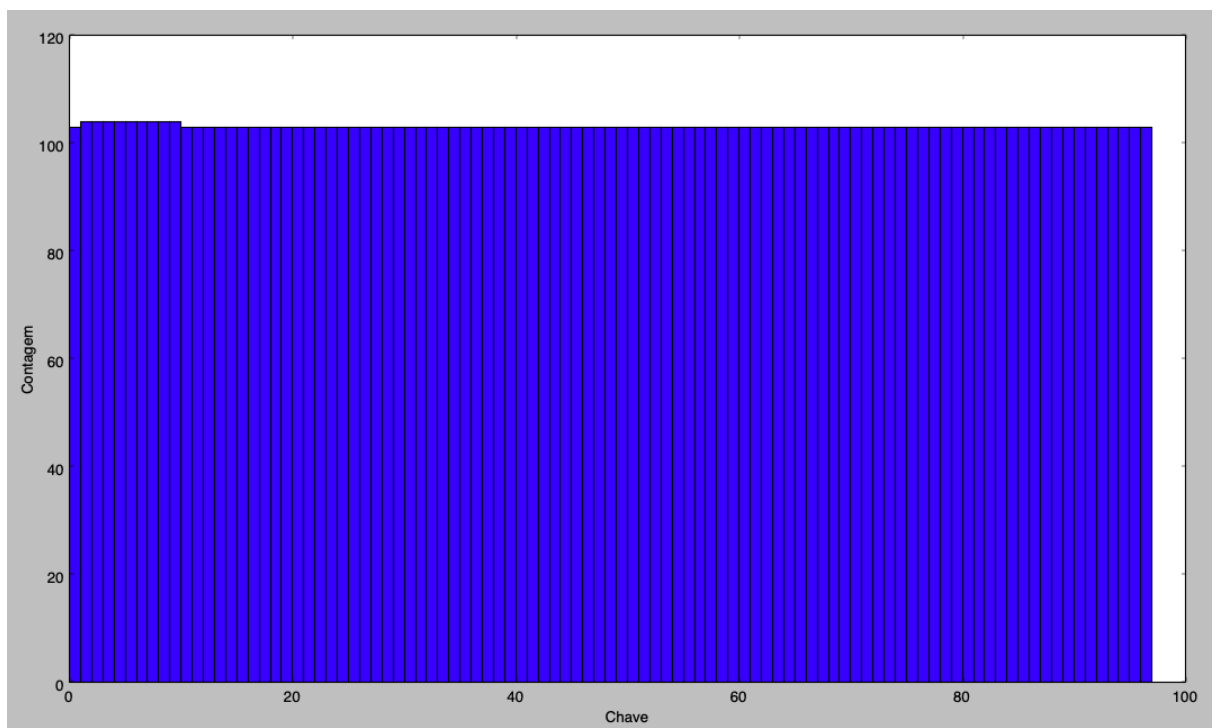
1. Estrutura e execução

Link para o repositório: https://github.com/gabriel-murakami/hash_function

O código foi implementado em Python e pode ser executado rodando diretamente o arquivo `main.py`. A biblioteca contendo as funções hash está contida no arquivo `hash_functions.py` e pode ser importado com `import hash_functions` no início do arquivo desejado. Depois da importação as funções na biblioteca podem ser acessadas com `hash_functions.div_method(key, m)` para o método da divisão e `hash_functions.multi_method(key, m, a)` para o método da multiplicação. Para os dois métodos o valor de m não pode ser menor ou igual a zero. Para o método de divisão, a constante A deve estar entre 0 e 1, não incluindo os extremos.

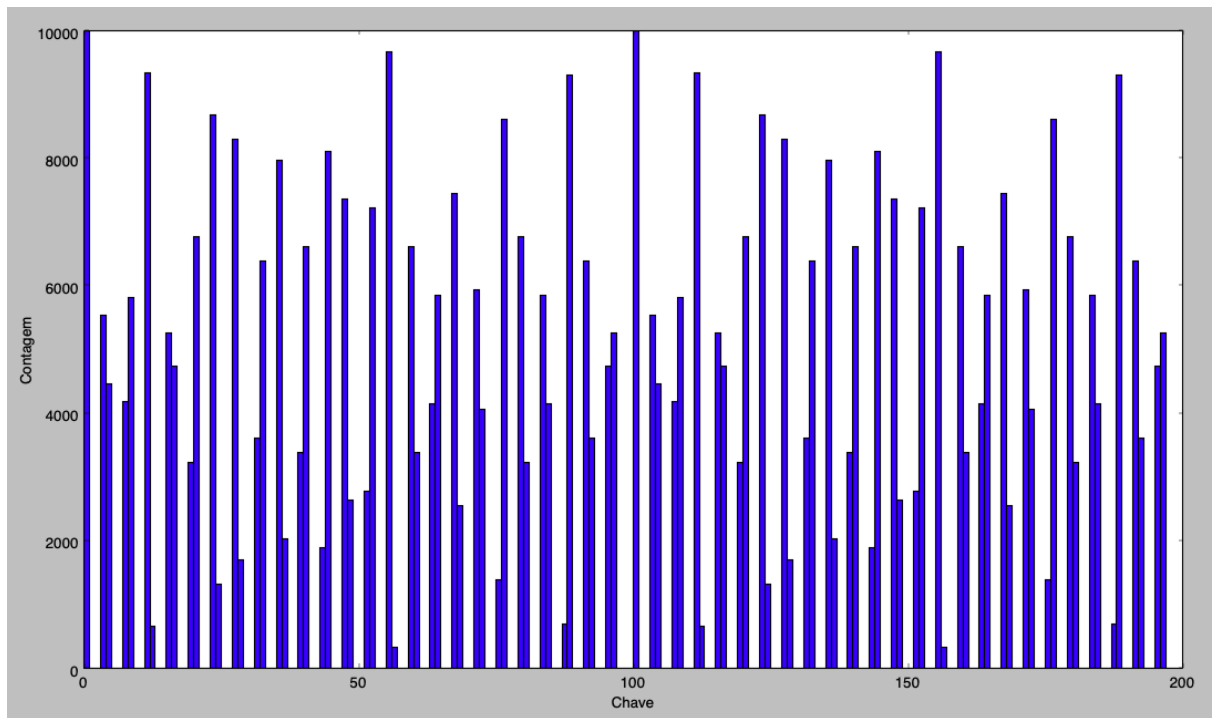
2. Método da Divisão

- Saída: [3, 15, 27, 39, 51, 63, 75, 87, 99]. Para esse espalhamento é possível perceber que todos os valores são múltiplos de 3 e o próximo número é sempre o anterior mais 12
- Saída: [3, 14, 25, 36, 47, 58, 69, 80, 91]. Para esse espalhamento é possível perceber que iniciando no 3, cada valor é o seu anterior mais 11
- Espalhamento para $m=97$ e $\text{chave}=\{1,2,3,\dots,10000\}$

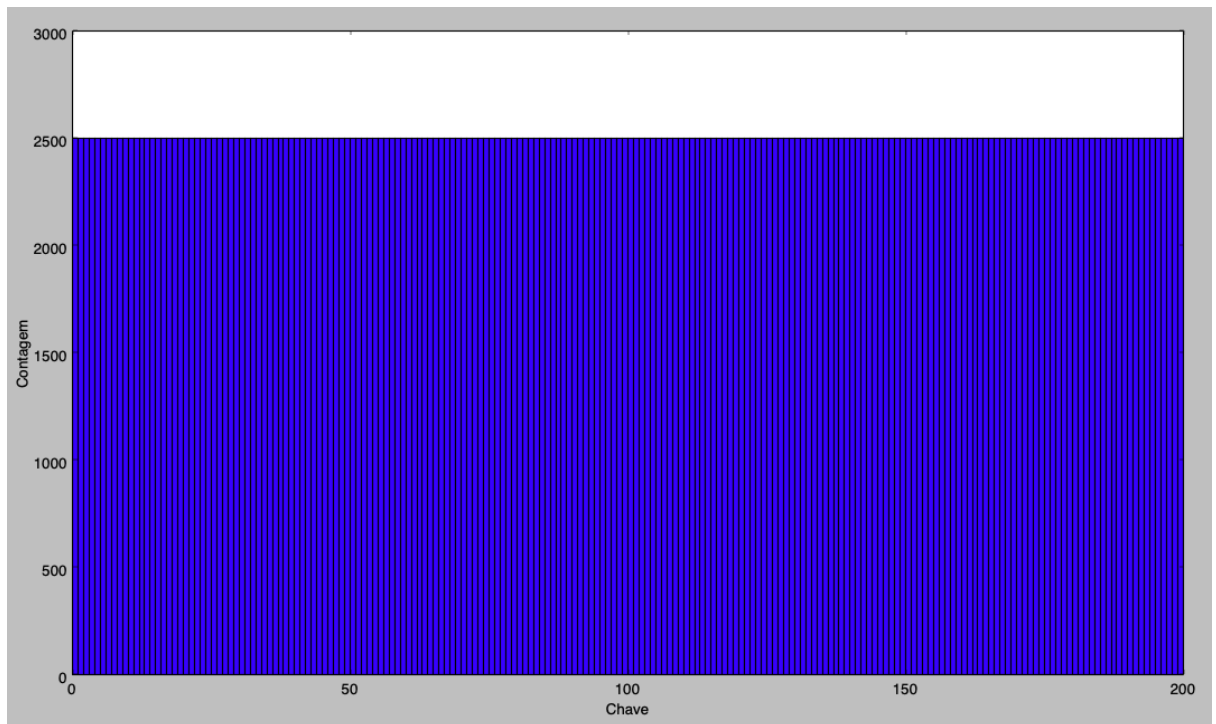


3. Método da Multiplicação

a. Espalhamento para $m=200$ e $A=0.62$



b. Espalhamento para $m=200$ e $A=0.61803398875$



É notável a diferença de resultado com a simples mudança da constante A . No primeiro caso temos um espalhamento aleatório, no segundo caso temos um espalhamento uniforme perfeito, com a distribuição exata de 2500 itens para cada chave