

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ

RELATÓRIO DE AVALIAÇÃO ESTRUTURADA DE
TABELAS HASH

Giovani Yamaguchi Tortato
Matheus Garozi

Objetivo

Comparar a eficiência de duas implementações de tabelas hash que utilizam o método de Encadeamento Separado para tratamento de colisões.]A comparação é feita com base no número de colisões, tempo de execução e distribuição de chaves (clusterização).

Metodologia de Implementação

Estrutura do Código: A arquitetura do sistema é baseada em uma classe fundamental e abstrata, denominada `TabelaHashBase`, responsável por encapsular o núcleo da estrutura de dados, incluindo a matriz de armazenamento, o controle de colisões e os métodos de inserção e busca.

Tratamento de Colisões: O mecanismo empregado para resolver conflitos de índices foi o Encadeamento Separado. Ao invés de procurar linearmente um novo espaço no vetor principal, cada posição da tabela armazena o início de uma lista encadeada, onde todos os elementos que mapeiam para o mesmo índice são armazenados.

Capacidade da Tabela: A estrutura foi definida com uma capacidade constante e não redimensionável de 32 posições. Essa limitação proposital foi estabelecida para forçar uma alta taxa de ocupação, para permitir uma análise mais rigorosa do desempenho das funções hash sob condições de estresse.

Funções Hash:

Tabela 1 (Hash por Soma de ASCII): Utiliza um método direto onde o índice é gerado pela soma dos valores ASCII de todos os caracteres da chave, seguida pela operação de módulo da capacidade.

Tabela 2 (Hash por Dobramento): Implementa um algoritmo que divide a chave em segmentos, converte esses segmentos em valores numéricos e soma-os. O índice final é determinado pelo módulo dessa soma.

Comparação de desempenho:

Métrica	Função Hash 1 (Soma ASCII)	Função Hash 2 (Dobramento)
Número de Colisões	4969	4969
Tempo Total de Inserção	2.6034 ms	3.2092 ms
Tempo Total de Busca	10.0695 ms	12.3323ms

Análise de distribuição:

Métrica	Função Hash 1 (Soma ASCII)	Função Hash 2 (Dobramento)
Média de Chaves por Posição	156.28	156.28
Desvio Padrão	10.07	13.51
Pior Posição (Máximo)	182 chaves	185 chaves

Resultados de Desempenho:

Função Hash 1: Soma ASCII

```
RELATÓRIO DA TABELA HASH: Função Hash 1 (Soma ASCII)
=====
1. Número de Colisões: 4969
2. Tempo Total de Inserção: 2,6034 ms
3. Tempo Total de Busca (1000 nomes): 10,0695 ms
4. Distribuição das Chaves por Posição (Clusterização):
+-----+-----+
| Posição | Chaves Inseridas |
+-----+-----+
| 0        | 160                |
| 1        | 171                |
| 2        | 161                |
| 3        | 146                |
| 4        | 150                |
| 5        | 149                |
| 6        | 167                |
| 7        | 159                |
| 8        | 164                |
| 9        | 146                |
| 10       | 155                |
| 11       | 149                |
| 12       | 176                |
| 13       | 150                |
| 14       | 182                |
| 15       | 148                |
| 16       | 139                |
| 17       | 146                |
| 18       | 162                |
| 19       | 160                |
| 20       | 148                |
| 21       | 152                |
| 22       | 158                |
| 23       | 159                |
| 24       | 147                |
| 25       | 149                |
| 26       | 157                |
| 27       | 147                |
| 28       | 160                |
| 29       | 159                |
| 30       | 177                |
| 31       | 148                |
+-----+-----+
```

Função Hash 2: Dobramento

```
=====
RELATÓRIO DA TABELA HASH: Função Hash 2 (Dobramento)
=====
1. Número de Colisões: 4969
2. Tempo Total de Inserção: 3,2092 ms
3. Tempo Total de Busca (1000 nomes): 12,3323 ms
4. Distribuição das Chaves por Posição (Clusterização):
+-----+-----+
| Posição | Chaves Inseridas |
+-----+-----+
| 0        | 165               |
| 1        | 136               |
| 2        | 146               |
| 3        | 151               |
| 4        | 176               |
| 5        | 162               |
| 6        | 151               |
| 7        | 142               |
| 8        | 169               |
| 9        | 171               |
| 10       | 155               |
| 11       | 146               |
| 12       | 172               |
| 13       | 150               |
| 14       | 168               |
| 15       | 160               |
| 16       | 143               |
| 17       | 131               |
| 18       | 185               |
| 19       | 160               |
| 20       | 164               |
| 21       | 153               |
| 22       | 156               |
| 23       | 154               |
| 24       | 144               |
| 25       | 176               |
| 26       | 171               |
| 27       | 168               |
| 28       | 138               |
| 29       | 133               |
| 30       | 160               |
| 31       | 145               |
+-----+-----+
```

Conclusão

A Função Hash 1 (Soma ASCII) teve uma uniformidade ligeiramente superior, com uma variação menor entre a posição mínima e máxima.

A Função Hash 1 (Soma ASCII) foi consistentemente mais rápida, tanto na inserção (2,6034 ms) quanto na busca (10,0695 ms). A diferença nos tempos (aproximadamente 0.6 ms mais rápida na inserção e 2.2 ms mais rápida na busca) isso acontece porque o cálculo da soma ASCII é menos complexo.