

# Trabalho Prático

## Algoritmos e Estruturas de Dados

### Universidade Federal de São João Del Rei

Christoffer de Paula Oliveira

19 de Junho de 2016

## 1 Introdução

Um termo muito usado na computação é o de função de hash. Função de hash seria qualquer algoritmo que mapeie dados grandes e de tamanho variável para pequenos dados de tamanho fixo. Por esse motivo, as funções Hash são conhecidas por resumirem o dado, sendo funções fáceis e rápidas de se calcular. A principal aplicação dessas funções é a comparação de dados grandes ou secretos. Se tratando de uma função de hash não podemos desconsiderar certas propriedades, como por exemplo a função deve ser unidirecional, ou seja, não é possível recuperar os dados a partir do resumo gerado pela função, outra propriedade é a de colisões que seria quando dois dados originais geram o mesmo resumo, a menor quantidade de colisões definem as melhores funções de hash, outra propriedade é a de que se um mesmo dado passar pela função de hash mais de uma vez deve gerar o mesmo resumo.

## 2 Objetivo

Objetivo do trabalho é implementar um código em linguagem C, onde deverá conter uma função de hash, para mapear os dados em um array. Já para a exibição dos, dados deverá ser feita uma ordenação de maneira decrescente em relação a frequência, quantas vezes o dado se repetiu, e ordem lexicográfica em caso do mesmo número de frequência.

## 3 Implementação

O algoritmo no primeiro instante deixa o dado na forma padrão, no caso deste trabalho tirando as maiúsculas e sinais de pontuação, para depois disso se o dado estiver de acordo com o pedido, no nosso caso estar em minúsculo, sem sinal de pontuação e ter mais de 1 caracter, usar a hash para mapear essa dado

e alocar no array. Por último é feita a ordenação dos dados como pedido e exibição.

### **3.1 Função de Hash**

A função de hash utilizada foi a chamada Método de Congruência Linear, que consiste em calcular o resto da divisão do valor inteiro que representa o dado, pelo tamanho da tabela de hash (tamanho do array a ser utilizado).

### **3.2 Tratamento dos Dados**

Como o dado é uma string é feito um processo para calcular o valor dessa string a partir da tabela ASCII considerando a posição das letras, que consiste em pegar o resultado da operação de um valor fixo que é multiplicado, pelo resultado que foi iniciado com o valor 7, pela posição e pelo valor na tabela ASCII de cada letra, o resultado dessa operação enquanto não terminar as letras é o novo resultado que será multiplicado.

### **3.3 Endereçamento Aberto**

Para o tratamento das colisões gerados pela função de hash temos o algoritmo conhecido como endereçamento aberto, nesse algoritmo há uma estratégia muito utilizada que é a sondagem linear, que consiste em tentar espalhar os dados que geraram colisões de forma linear no array, ou seja se dois dados geraram o mesmo resumo é feita a verificação do próximo índice do array, se estiver vazio o dado é alocado se não vá na posição seguinte até que seja encontrada uma posição vazia ou percorrido todo o array. Essa posição é calculada com o resto da divisão do tamanho do array pela posição gerada na função de hash mais a quantidade de vezes que tentou alocar o dado e não conseguiu.

### **3.4 Exibição dos Dados**

Para exibir os dados foi usado um vetor auxiliar que armazena somente as palavras que tiverem frequência maior que 2, então usando o algoritmo de ordenação selectionsort, foi feita a ordenação de acordo com a frequência de cada dado em ordem decrescente, porém usando o mesmo algoritmo de ordenação é feita a ordenação das palavras(dados) em ordem lexicográfica se estiverem com mesmo número de frequência que outras palavras já inseridas, no fim é exibido cada elemento desse array auxiliar.

## 4 Ambiente

*S.O: Ubuntu 15.01, 64 Bits*

*Processador: Intel Core i3-2330M @ 2.2GHz*

*Memória RAM: 3,0 GB*

*Compilador: gcc*

*IDE: Geany*

## 5 Conclusão

Neste trabalho foi apresentado a função de hash e suas propriedades, o uso do endereçamento aberto para tratamento de colisões. Para a função de hash foi utilizado o método de congruência linear e para endereçamento aberto utilizado a estratégia de sondagem linear, fica claro que com o uso de hash não é necessário saber qual a chave utilizada para armazenar o dado, o que se torna bastante viável para informações sigilosas, para acesso dos dados a hash traz uma rapidez considerável na busca sendo apenas  $O(1)$ , contudo vale resaltar que a função de hash tem que ser rápida e eficaz, para não demantar muito processo e permanecer com sua complexidade.

## 6 Referências

<http://www.techtudo.com.br/artigos/noticia/2012/07/o-que-e-hash.html>

[https://pt.wikibooks.org/wiki/Algoritmos\\_e\\_Estruturas\\_de\\_Dados/Tabela\\_de\\_Hash\\_Endere.C3.A7amento\\_Aberto](https://pt.wikibooks.org/wiki/Algoritmos_e_Estruturas_de_Dados/Tabela_de_Hash_Endere.C3.A7amento_Aberto)

<https://programacaodescomplicada.wordpress.com/>