

# Relatório de Documentação do Trabalho de Hash de CI 057 - Algoritmos e Estrutura de Dados III

Paulo Mateus Luza Alvez GRR20203945 | Heric Camargo GRR20203959

05/04/2022

## Funções Públicas

**void initialize\_\_hash\_\_table (int hash[][M]);**

- Inicializa as valores da tabela, preenchendo os slots com o valor -1

**int hash\_\_search (int key, int t1[][M], int t2[][M]);**

- Faz uma busca do valor indicado em 'key' dentro da hash, caso encontre, retorna a posição o contrário, retorna -1

**void hash\_\_insert (int key, int t1[][M], int t2[][M]);**

- Faz a inserção do valor indicado em 'key' na hash. Segue as definições indicadas, não havendo re-hashing ou sobreposição de valores.

**void hash\_\_delete (int key, int t1[][M], int t2[][M]);**

- Faz a exclusão do valor indicado em 'key' na hash. Caso o valor excluído esteja na T1, marcamos o slot com o valor -2, que indica 'excluído'. Assim, evitando problemas na busca.

**void hash\_\_print (int t1[][M], int t2[][M]);**

- Imprime o resultado da hash. Funciona da seguinte maneira:
  1. Passa pela hash T1, e ao encontrar um valor diferente de -1 || -2 adiciona de maneira ordenada sua chave no array key\_array
  2. Da mesma maneira, seguindo a ordenação do key\_array, adiciona os valores da coluna e 'numero da hash' nos print\_arrays column\_array e hash\_array respectivamente
  3. Realiza o mesmo processo para a hash t2 Desta maneira, conseguimos obter as informações ordenadas de forma correta para o print da hash como pedido.

## Funções Internas

**int h1 (int key);**

- Função de acesso a hash t1

**int h2 (int key);**

- Função de acesso a hash t2

**void sorted\_inserction (int key\_array[M], int column\_array[M], int hash\_array[M], int key, int column, int hash, int pos);**

- Função de inserção ordenada nos arrays utilizados para o print