

# INF01124 - Classificação e Pesquisa de Dados - Exercício 4

## Professor João Comba

### 1 Implementação de Tabela Hash

O objetivo deste laboratório é implementar uma tabela Hash. O tamanho da tabela  $M$  deve ser informado quando da construção da tabela. A tabela hash deve armazenar registros compostos de uma chave identificadora e dados satélites associados. Para a implementação da Tabela Hash, realize as seguintes tarefas:

1. escolha uma função de hash para mapear a chave de cada registro para um inteiro entre 0 e  $M - 1$ ;
2. implemente o método de resolução de conflitos de endereçamento fechado com listas encadeadas;
3. implemente a operação de inserção de um registro na tabela hash; e
4. implemente a operação de busca de um registro por uma chave na tabela hash.

### 2 Arquivo de Testes

A implementação da tabela Hash deve ser testada com um arquivo .CSV contendo descrições de 18944 nomes de jogadores de futebol. O arquivo é disponibilizado junto ao enunciado deste trabalho com o nome de *players-fifa.csv*. A primeira linha do arquivo contém a descrição em cada coluna do registro a ser armazenado. A chave é o identificador de cada jogador (sofifa\_id), e os dados satélites consistem no nome completo do jogador e um string contendo uma lista de posições que o jogador atua. Cada linha do arquivo contém os dados de um jogador.

Um exemplo deste arquivo é mostrado abaixo:

```
sofifa_id,name,player_positions
158023,Lionel Andres Messi Cuccittini,"RW, ST, CF"
20801,Cristiano Ronaldo dos Santos Aveiro,"ST, LW"
200389,Jan Oblak,GK
188545,Robert Lewandowski,ST
190871,Neymar da Silva Santos Junior,"LW, CAM"
192985,Kevin De Bruyne,"CAM, CM"
231747,Kylian Mbappe Lottin,"ST, LW, RW"
192448,Marc-Andre ter Stegen,GK
203376,Virgil van Dijk,CB
212831,Alisson Ramses Becker,GK
...
```

A lista de posições nas quais o jogador atua usa as seguintes abreviaturas:

```
GK goalkeeper
CB center-back
RB right back
LB left back
CDM central defensive midfielder
CM central midfielder
```

RM right midfielder  
LM left midfielder  
CAM central attacking midfielder  
LW left wing  
RW right wing  
CF center forward  
ST striker

### 3 Experimentos

Para testar a sua implementação da tabela hash vamos coletar estatísticas relacionadas à construção da tabela e consultas por registros armazenados na tabela. Como o tamanho da tabela é muito importante na performance, vamos realizar experimentos com os seguintes valores de tamanho da tabela: 997, 1999, 3989 e 7993. Todos estes números são números primos imediatamente anteriores aos números 1000, 2000, 4000 e 8000.

Para cada tamanho de tabela, realize um experimento que consiste em inserir na tabela os dados de jogadores do arquivo *players-fifa.csv*, e após realize consultas por identificadores (*sofifa\_id*) de jogadores descritas no arquivo *consultas.csv*. Para avaliar a performance para cada tamanho de tabela, gere um arquivo de nome *experimentoM.txt* (onde M possui os valores 997, 1999, 3989 e 7993) com estatísticas sobre a construção da tabela e consultas realizadas:

- **estatísticas da tabela hash:** tempo de construção da tabela, taxa de ocupação da tabela (número de entradas usadas divididas pelo tamanho da tabela), tamanho máximo de lista, e a média do tamanho das listas (para entradas não vazias); e
- **estatísticas das consultas:** tempo de realização de todas as consultas seguido de resultados das consultas. Para cada chave encontrada, retorne o nome do jogador se ele for encontrado, ou o string *NAO ENCONTRADO*. Ao final da linha, adicione o número de valores consultados durante a consulta. Ao final, retorne o número máximo e médio de testes executados em todas as consultas.

O arquivo de saída deve ter o seguinte formato:

```
PARTE1: ESTATISTICAS DA TABELA HASH
TEMPO DE CONSTRUCAO DA TABELA #TEMPO EM MILISEGUNDOS
TAXA DE OCUPACAO #NUMERO_OCUPADAS/TAMANHO
TAMANHO MAXIMO DE LISTA #MAX1
TAMANHO MEDIO DE LISTA #MED1

PARTE 2: ESTATISTICAS DAS CONSULTAS
TEMPO PARA REALIZACAO DE TODAS CONSULTAS #TEMPO EM MILISEGUNDOS
158023 Lionel Andres Messi Cuccittini #CONSULTAS
20801 Cristiano Ronaldo dos Santos Aveiro #CONSULTAS
...
99999 NAO ENCONTRADO #CONSULTAS
MAXIMO NUMERO DE TESTES POR NOME ENCONTRADO #MAX2
MEDIA NUMERO DE TESTES POR NOME ENCONTRADO #MED2
```

### 4 Opcional: Desafio Bônus - Problema Hash Tables

Resolva o problema BEE 1256 - Hash Tables

Este é um problema de maratona de programação. O problema será considerado completo se a solução for aceita no site da BeeCrowd. O comprovante da aceitação e o código deve ser entregue junto com a solução do laboratório. A entrega correta valerá um adicional de 25% pontos.

## 5 Entrega

A solução deve ser enviada pelo Moodle dentro de um arquivo .zip, contendo os seguintes arquivos:

- **integrantes.txt**: coloque o nome dos integrantes do grupo (até 2 pessoas) , com um nome por linha
- **readme.txt**: explique a função de hash e a como foi implementada a resolução de conflitos;
- **experimento997.txt**: estatísticas do experimento com tamanho de tabela 997
- **experimento1999.txt**: estatísticas do experimento com tamanho de tabela 1999
- **experimento3989.txt**: estatísticas do experimento com tamanho de tabela 3989
- **experimento7993.txt**: estatísticas do experimento com tamanho de tabela 7993
- código fonte correspondente a solução
- opcional: código e comprovante de aceite do problema desafio