

# Pesquisa e Classificação de Dados

2023/1 - Trabalho da disciplina

## Enunciado

O Banco Central precisa da sua ajuda para implementar o *Píques*, um novo sistema de pagamentos instantâneos. Neste sistema, cada pessoa cadastrada tem duas *chaves* que a identifica: uma chave aleatória (uma *string* de letras minúsculas) e um CPF (uma sequência de 11 dígitos). A partir de qualquer uma das chaves de uma pessoa, o sistema deve conseguir determinar todos os seus dados bancários (seu nome, seu banco, sua agência e sua conta-corrente).

Considere que os cadastros estão em um arquivo de texto chamado `banco.txt`. A primeira linha deste arquivo contém  $N$ , o número de pessoas cadastradas. Cada uma das próximas  $N$  linhas contém o cadastro de uma pessoa, contendo seu nome, seu banco, o número da agência, o número da conta-corrente, sua chave aleatória e seu CPF, nesta ordem. Como exemplo, considere o arquivo `banco.txt` com o seguinte conteúdo:

```
5
JoaodaSilva BancodoBrasil 39732 153038 jdpupz 62476478527
AlanTunisiao Itau 1506 94761 chavedoalan 62476479214
NazareCarminha BancodoBrasil 89703 70486 jdpup 90794576842
JiminJHope BancoTudoSerto 117788 28659 shiningthroughthecitywithalittlefunkandsoul 33722985054
FredericoMercurio QueensBank 62010 07291 bohemianrhapsody 15937081601
```

Neste exemplo há 5 pessoas cadastradas. A primeira é o JoaodaSilva, que tem conta no BancodoBrasil (agência 39732, conta-corrente 153038), sua chave aleatória é `jdpupz` e seu CPF é 62476478527; a segunda pessoa cadastrada é o AlanTunisiao, que tem conta no Itau (agência 1506, conta-corrente 94761), sua chave aleatória é `chavedoalan` e seu CPF é 62476479214; e assim por diante.

Escreva um programa em C que lê um arquivo chamado `banco.txt` com os cadastros no formato acima, e coloca seus dados na estrutura de dados descrita na próxima seção. Em seguida, seu programa deve ler uma sequência de comandos do usuário. Cada comando pode ser:

- `? [chave]`: Busca a chave dada na estrutura de dados. A chave pode ser uma chave aleatória ou um CPF. Seu programa deve imprimir uma linha contendo (nessa ordem) o nome, o banco, a agência e a conta-corrente da pessoa com a chave dada. Caso nenhuma pessoa tenha essa chave, imprima `Chave [chave] nao cadastrada.` ou `CPF [chave] nao cadastrado.`, de acordo com o caso.
- `+ [nome] [banco] [agencia] [conta] [chave] [cpf]`: Inclui um novo cadastro na estrutura de dados. Após a inserção, seu programa deve imprimir a linha `[nome] cadastrado com sucesso..` Você pode assumir que a chave e o CPF cadastrados não serão iguais a nenhuma chave ou CPF já cadastrados na estrutura.
- `- [chave]`: Remove da estrutura de dados o cadastro da pessoa com a chave dada. A chave pode ser uma chave aleatória ou um CPF. Após a remoção, seu programa deve imprimir `[nome] removido com sucesso..` Você pode assumir que sempre existirá uma pessoa com a chave dada no momento em que este comando é executado.
- `F`: Termina o programa. Neste momento, o arquivo `banco.txt` deve ser atualizado para conter todos (e apenas) os cadastros que estão na estrutura de dados no momento.

Veja o seguinte exemplo de execução, que considera que o conteúdo do arquivo `banco.txt` é, inicialmente, o dado acima (comandos `?`, `+`, `-` e `F` são a entrada do usuário, enquanto as demais linhas são a saída gerada pelo programa):

```
? 62476478527
JoaodaSilva BancodoBrasil 39732 153038

? chavedoalan
AlanTunisio Itau 1506 94761

? chavedojoao
Chave chavedojoao nao cadastrada.

? 12345678901
CPF 12345678901 nao cadastrado.

- shiningthroughthecitywithalittlefunkandsoul
JiminJHope removido com sucesso.

? 33722985054
CPF 33722985054 nao cadastrado.

- 62476479214
AlanTunisio removido com sucesso.

+ MarcoBF BancodoMato 0001 234561 welctometothemato 12345678901
MarcoBF cadastrado com sucesso.

? 12345678901
MarcoBF BancodoMato 0001 234561

+ Kleber WheyBank 424 24242 birll 15975382469
Kleber cadastrado com sucesso.

? birl
Chave birl nao cadastrada.

? birll
Kleber WheyBank 424 24242

+ Bowser KoopaCoin 666 6666 peachespeachespeaches 6666666666
Bowser cadastrado com sucesso.

F
```

Após a execução do programa, o arquivo `banco.txt` deve ser atualizado com os seguintes cadastros (em qualquer ordem):

```
6
JoaodaSilva BancodoBrasil 39732 153038 jdpupz 62476478527
FredericoMercurio QueensBank 62010 07291 bohemianrhapsody 15937081601
NazareCarminha BancodoBrasil 89703 70486 jdpup 90794576842
MarcoBF BancodoMato 0001 234561 welctometothemato 12345678901
Kleber WheyBank 424 24242 birll 15975382469
Bowser KoopaCoin 666 6666 peachespeachespeaches 6666666666
```

- Todas as strings (nomes, bancos e chaves aleatórias) têm até 100 caracteres e nenhuma tem espaços;



## Orientações

- O trabalho pode ser feito por equipes de *até* 2 (dois) estudantes;
- Submeta, via *Moodle*, um pacote **zip** ou **tar.gz** contendo todo o código-fonte necessário para compilar e executar sua solução, além de um arquivo de texto (txt) onde conste:
  - O nome de todos os integrantes da equipe;
  - Toda informação que a equipe julgar relevante para a correção (como *bugs* conhecidos, detalhes de implementação, escolhas de projeto, etc.)
- Comente adequadamente seus códigos para facilitar a correção;
- Atenção: a correção será parcialmente automatizada, e a saída do programa será testada com outras entradas além das fornecidas como exemplo. *Siga **fielmente** o formato de saída dado nos exemplos*, sob pena de grande redução da nota;
- Certifique-se que seu programa compila e funciona antes de submetê-lo;
- O trabalho deve ser entregue até **18 de Junho de 2023, 23:59**, apenas via *Moodle*. Trabalhos entregues por outros meios ou fora do prazo não serão aceitos. É suficiente que o trabalho seja submetido por apenas um estudante da equipe;
- Trabalhos detectados como cópia, plágio ou comprados receberão **todos** a nota 0 (**ZERO**) e estarão sujeitos a abertura de Processo Administrativo Disciplinar Discente.