
Universidade Federal de São Carlos
Centro de Ciências Exatas e Tecnologia
Departamento de Computação

Organização e Recuperação da Informação

Trabalho Prático 1

Alunos:

Eric Pereira Queiroz Moreira RA: 799662
Gabriel Meirelles Carvalho Orlando RA: 790728

(I) Introdução

Esse trabalho teve como objetivo aplicar diferentes métodos para representar os conceitos de banco e de registro. O banco de dados usado no projeto foi do seguinte formato:

```
KEY (i.e., número que identifica a pessoa)
LASTNAME (i.e., sobrenome da pessoa)
FIRSTNAME (i.e., primeiro nome da pessoa)
ADDRESS (i.e., endereço com logradouro, número e complemento)
CITY (i.e., cidade)
STATE (i.e., sigla do estado com 2 caracteres, tal como SP)
ZIP (i.e., CEP, tal como 22222-222)
PHONE (i.e., número do telefone com DDD, tal como (022)2222-2222)
```

Dois tipos de métodos foram aplicados na construção desse banco de dados: **campos e registros de tamanho fixo em bytes** e **campos e registros com indicadores de tamanho**. A linguagem utilizada no projeto foi a linguagem C. Todos os códigos referenciados neste documento estão disponibilizados em um repositório público do GitHub feito pelos autores, listado em [1].

(II) Campos e Registros de Tamanho Fixo (Em Bytes)

Compilação

A compilação do programa foi feita no sistema operacional Linux (Distro Ubuntu 20.04) utilizando o seguintes comando:

```
gcc -std=c99 -Wall -Werror -Werror=vla -pedantic-errors -g -lm main.c
t1.c t1.h -o 1
```

Formato do Registro

O registro é escrito no arquivo binário da seguinte forma:

```
#|e|c1|c2|...|cn#|e|...
```

- O caractere ‘#’ marca o início do registro.
- O caractere ‘|’ funciona como um delimitador, separando os diferentes campos do registro.
- A variável ‘e’ (Existe) serve para verificar se o registro existe logicamente (Ou não). O valor ‘1’ representa que ele existe, enquanto o valor ‘0’ representa que ele não existe logicamente.
- Os valores c1, c2, ..., cn representam os diferentes campos presentes no registro.

Inserção do Registro

1. Inicialmente, faz-se uma busca pela chave primária do registro, o qual planeja-se inserir.

2. Caso a chave primária já exista, o registro não é inserido. Caso contrário, o processo continua.
3. Verifica-se a existência de remoção lógica, verificando o valor do campo e todos os registros de forma sequencial
 - a. Caso **tenha havido remoção lógica**, verifica-se a posição onde a remoção ocorreu e troca o valor da variável 'e' de 0 para 1, e insere o registro na posição.
 - b. Caso **não tenha havido remoção lógica**, o novo registro é inserido ao final do arquivo.

Remoção do Registro

A remoção do registro é bem mais simples: Inicialmente é feita a busca da chave a qual deseja-se remover (Caso ela não exista, o processo retorna erro). Caso ela exista, o valor da variável 'e' é alterado de 1 para 0, representando a remoção lógica.

Inserção de Registro usando Nome (Chave Secundária)

A inserção da chave secundária no arquivo correspondente é feita de forma alfabeticamente ordenada (de forma a fazer com que a busca por chave secundária seja de complexidade $O(\log n)$ com busca binária) e com aproveitamento do espaço lógico do arquivo. Após encontrar uma posição que conserve a ordenação, caso a posição em questão possua um campo 'e' igual a 0, os outros campos do registro são substituídos - o tamanho do arquivo não é alterado. Caso nenhum espaço possa ser reaproveitado, o registro é inserido ao final do arquivo.

Remoção de Registro usando Nome (Chave Secundária)

A remoção de chave secundária ocorre utilizando o vetor de RRNs (De tamanho vinte) contido em cada nome. Caso tenha apenas um RRN, a variável 'e' do registro é alterada de 1 para 0, representando sua remoção lógica. Caso contrário, todos os RRNs são inseridos na posição anterior do vetor (O RRN do espaço $v[1]$ é copiado para $v[0]$, enquanto o do espaço $v[2]$ é copiado para $v[1]$, etc., enquanto o -1 é inserido no final para indicar o término da lista).

Busca pelo campo “name”

_____Para que a chave com o nome correspondente seja encontrada, realiza-se uma busca binária no arquivo de chaves secundárias. Visto que esse arquivo encontra-se sempre ordenada, a busca pode ser feita em complexidade $O(\log N)$, onde N é o número de chaves. Os registros encontrados como resultado da busca binária são retornados adequadamente.

Busca pelo campo “key”

A busca do campo *key* é feita diretamente no arquivo de dados de maneira sequencial, apenas buscando pela chave desejada. Caso a chave buscada seja encontrada, deve-se verificar se ela existe logicamente. Caso exista logicamente, sua posição correspondente deve ser retornada. Caso contrário, ela é tida como não-encontrada.

(III) Campos e Registros com Indicadores de Tamanho

Compilação

O programa foi compilado no sistema operacional Linux, dessa maneira:

```
gcc -std=c99 -Wall -Werror -Werror=vla -pedantic-errors -g -lm main.c t.c t.h -o l
```

Formato do registro

O registro é escrito no arquivo binário da seguinte forma, cada registro tem um indicador, #, que marca o início do registro, assim se segue:

```
#|e|tam|t_c1|c1|t_c2|c2|...|t_cn|cn#...
```

- O primeiro caractere ‘#’ marca o início do registro.
- O caractere ‘|’ funciona como um delimitador, separando os diferentes campos do registro.
- A variável ‘e’ (Existe) serve para verificar se o registro existe logicamente (Ou não). O valor ‘1’ representa que ele existe, enquanto o valor ‘0’ representa que ele não existe logicamente.
- A variável ‘tam’ representa o tamanho total do registro escrito (Contando com os delimitadores).
- As variáveis *t_c1*, *t_c2*, etc. são os indicadores do tamanho do campo adjacente.
- As variáveis *c1*, *c2*, ..., *cn* são os campos do registro em si, os quais possuem tamanhos e tipos diferentes.

Inserção do Registro

A inserção desse tipo de registro primeiramente consiste em checar se a chave primária a qual deseja-se inserir já existe no arquivo. Como a chave primária não pode se repetir, caso ela já exista a inserção não ocorre. Outra restrição para essa operação ocorre no caso do vetor de tamanho fixo (O qual armazena os bytes offset da chave secundária Name) estar cheio. O vetor possui 20 espaços. Caso ele não possua nenhum espaço restante, a inserção também não ocorre.

Após a checagem dessas restrições, é feita a pesquisa no registro principal para verificação da existência de espaços que possam ser reaproveitados. A estratégia utilizada é a mesma da seção (II), a *First Fit*. O detalhamento e motivação para escolha deste algoritmo estão descritos na seção (V).

Para reaproveitar o espaço existem duas alternativas: Caso o tamanho do registro a ser inserido seja *igual* ao do registro o qual será reaproveitado a inserção é feita diretamente, semelhante ao que foi explicado na inserção de chave secundária da seção (II).

Caso o registro a ser inserido seja *menor* que o registro atual, deve-se indicar o espaço restante (O que custa espaço). Desse modo, o registro a ser inserido deve possuir pelo menos **9 bytes** a menos do que o registro a ser reaproveitado. Dessa forma, o registro será:

```
#|1|tam|...#0|tamanho_reg_antigo - (tam + 8)|...
```

Deve-se analisar esse valor **-8** (Isso ocorre pois no espaço restante tem-se 4 caracteres, cada um com **1 byte**, e um inteiro, com **4 bytes**).

Obviamente, o reaproveitamento não ocorre caso o registro a ser inserido seja maior do que o registro atual.

Inserção de Registro usando Nome (Chave Secundária)

A inserção da chave secundária no arquivo correspondente é feita de forma alfabeticamente ordenada (de forma a fazer com que a busca por chave secundária seja de complexidade $O(\log n)$ com busca binária) e com aproveitamento do espaço lógico do arquivo. Após encontrar uma posição que conserve a ordenação, caso a posição em questão possua um campo 'e' igual a 0, os outros campos do registro são substituídos - o tamanho do arquivo não é alterado. Caso nenhum espaço possa ser reaproveitado, o registro é inserido ao final do arquivo.

Remoção do Registro

_____A remoção se resume a trocar o valor da variável 'e' do registro a ser removido de 1 para 0 no arquivo principal.

Remoção de Registro usando Nome (Chave Secundária)

A remoção de chave secundária ocorre utilizando o vetor de RRNs (De tamanho vinte) contido em cada nome. Caso tenha apenas um RRN, a variável 'e' do registro é alterada de 1 para 0, representando sua remoção lógica. Caso contrário, todos os RRNs são inseridos na posição anterior do vetor (O RRN do espaço $v[1]$ é copiado para $v[0]$, enquanto o do espaço $v[2]$ é copiado para $v[1]$, etc., enquanto o -1 é inserido no final para indicar o término da lista).

Busca pelo campo “name”

_____Para que a chave com o nome correspondente seja encontrada, realiza-se uma busca binária no arquivo de chaves secundárias. Visto que esse arquivo encontra-se sempre ordenado, a busca pode ser feita em complexidade $O(\log N)$, onde N é o número de chaves. Os registros encontrados como resultado da busca binária são retornados adequadamente.

Busca pelo campo “key”

A busca do campo *key* é feita diretamente no arquivo de dados de maneira sequencial, apenas buscando pela chave desejada. Caso a chave buscada seja encontrada, deve-se verificar se ela existe logicamente. Caso exista logicamente, sua posição correspondente deve ser retornada. Caso contrário, ela é tida como não-encontrada.

(IV) Entrada do Programa

A entrada do programa deve ser um registro como utilizado nas seções (II) e (III). A ordem de inserção deve ser:

1. Chave (Key)
2. Nome (First Name)
3. Sobrenome (Last Name)
4. Cidade (City)
5. Sigla do estado (State)
6. CEP (Zip Code)
7. Telefone (Phone)
8. Número (Street Number)
9. Rua (Street)
10. Complemento (Complement)

Uma filtragem importante se trata de nomes de pessoa/cidade/estado compostos. Nomes compostos (Como por exemplo Rio de Janeiro) devem ter seus espaços alterados por caracteres ‘_’. No caso exemplo, Rio de Janeiro deve ser inserido como ‘Rio_de_Janeiro’. Um exemplo de entrada correta é:

```
> 195 Mia Conway Sobral SC 61367-737 (89)31822-8825 9273 Julio_Landing Apt_372
```

Além disso, cada campo possui restrições de tamanho e valor:

- Como o campo ‘key’ é um *long int*, seu valor deve ser um número de no máximo **8 bytes**.
- O campo ‘nome’ deve possuir no máximo **9 caracteres**.
- O campo ‘sobrenome’ deve possuir no máximo **10 caracteres**.
- O campo ‘cidade’ deve possuir no máximo **23 caracteres**.
- O campo ‘estado’ deve possuir no máximo **2 caracteres**.
- O campo ‘CEP’ deve possuir no máximo **9 caracteres**.
- O campo ‘telefone’ deve possuir no máximo **14 caracteres**.

Obs.: Os campos que representam strings possuem um valor a menos do que está declarado nos arquivos de cabeçalho pela presença da caractere ‘\0’ no final de toda *string* na linguagem C.

(V) Reaproveitamento do Espaço:

A estratégia utilizada nas seções (II) e (III) é a estratégia *First Fit*, a qual consiste em simplesmente procurar o primeiro espaço na lista que seja suficientemente grande para armazenar o registro. Caso o processo não ocupe todo o espaço, o espaço restante é disponibilizado como buraco na lista.

Essa estratégia foi escolhida pois trata-se de um algoritmo rápido e eficiente para o escopo do projeto, além de ter uma implementação simples e direta. Outras alternativas necessitariam ler o arquivo completo e seriam mais complexas e menos rápidas.

(VI) Bateria de Testes do Programa:

1. Na bateria de testes da seção (II), o arquivo é criado, alguns registros são inseridos e, por fim, todos os registros são exibidos na tela. Além disso, também testa-se a busca pelo registro número N existente no arquivo (Lembrando que somente são contabilizados registros os quais existam logicamente):

Caso 1: Inserção de registros.

```
-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:0
Arquivo criado com sucesso

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:1
Digite a entrada da seguinte forma:
Chave Name lname City State Zip-Code Telefone Numero Rua Complemento
->195 Mia Conway Sobral SC 61367-737 (89)31822-8825 9273 Julio_Landing Apt_372

Registro adicionado ao arquivo arquivo.bin

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:1
Digite a entrada da seguinte forma:
Chave Name lname City State Zip-Code Telefone Numero Rua Complemento
->767 Zenaida Dennis Novo_Hamburgo CE 22713-313 (41)01716-6435 09280 Neva_Springs Apt_146

Registro adicionado ao arquivo arquivo.bin
```

Caso 2: Mostra dos registros adicionados

```
-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:2

Registros do arquivo arquivo.bin
-----Registro: 0-----
KEY = 195
LAST NAME = Conway
FIRST NAME = Mia
CITY = Sobral
STATE = SC
CEP = 61367-737
PHONE = (89)31822-8825
ENDERECO = Julio_Landing,9273,Apt_372
-----

-----Registro: 1-----
KEY = 767
LAST NAME = Dennis
FIRST NAME = Zenaída
CITY = Novo_Hamburgo
STATE = CE
CEP = 22713-313
PHONE = (41)01716-6435
ENDERECO = Neva_Springs,9280,Apt_146
-----

-----Registro: 2-----
KEY = 245
LAST NAME = Watts
FIRST NAME = Fay
CITY = Sobral
STATE = SC
CEP = 24937-515
PHONE = (40)05787-6013
ENDERECO = Pollich_Hills,71782,Suite_335
-----
```


Caso 3: Busca por registro específico

```
-----Registro: 3-----
KEY = 647
LAST NAME = Mcmillan
FIRST NAME = Regina
CITY = Ribeirão_Preto
STATE = CE
CEP = 83857-374
PHONE = (26)88234-1645
ENDERECO = Shannon_Court,3037,Suite_356
-----

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:3
Digite o numero do registro:4

OPS, algo deu errado, nao existe o registro 4...

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:3
Digite o numero do registro:2

Registro encontrado:
-----Registro: 2-----
KEY = 245
LAST NAME = Watts
FIRST NAME = Fay
CITY = Sobral
STATE = SC
CEP = 24937-515
PHONE = (40)05787-6013
ENDERECO = Pollich_Hills,71782,Suite_335
-----
```

2. Além disso, também foram testadas a busca utilizando a chave secundária **NAME** (Testando busca por chaves existentes e inexistentes). Por fim, também testou-se a remoção de um registro com a chave primária 767 (E sua posterior busca para confirmar que a remoção ocorreu corretamente).

Caso 4: Busca por chave secundária (Nome)

```
-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:5
Digite o primeiro nome:Gabriel
Procurando pelo nome Gabriel

OPS, algo deu errado, nao existe o nome Gabriel...

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:5
Digite o primeiro nome:Regina
Procurando pelo nome Regina

Registro(s) com o nome: Regina abaixo:
Registro encontrado:
-----Registro: 3-----
KEY = 647
LAST NAME = Mcmillan
FIRST NAME = Regina
CITY = Ribeirão_Preto
STATE = CE
CEP = 83857-374
PHONE = (26)88234-1645
ENDERECO = Shannon_Court,3037,Suite_356
-----
```

Caso 5: Remoção do registro 767

```
-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:6
Digite a chave para remover:767
Registro com a chave 767 removido

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:2

Registros do arquivo arquivo.bin
-----Registro: 0-----
KEY = 195
LAST NAME = Conway
FIRST NAME = Mia
CITY = Sobral
STATE = SC
CEP = 61367-737
PHONE = (89)31822-8825
ENDERECO = Julio_Landing,9273,Apt_372
-----

-----Registro: 1-----
KEY = 245
LAST NAME = Watts
FIRST NAME = Fay
CITY = Sobral
STATE = SC
CEP = 24937-515
PHONE = (40)05787-6013
ENDERECO = Pollich_Hills,71782,Suite_335
-----
```

```

-----Registro: 2-----
KEY = 647
LAST NAME = Mcmillan
FIRST NAME = Regina
CITY = Ribeirão_Preto
STATE = CE
CEP = 83857-374
PHONE = (26)88234-1645
ENDERECO = Shannon_Court,3037,Suite_356
-----

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:4
Digite o numero da chave:767

OPS, algo deu errado, nao existe a chave 767...

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro n
4)Mostrar registro com a KEY x
5)Mostrar registro com FNAME s
6)Remover registro
7)Sair
-----
Digite o digito:5
Digite o primeiro nome:Zenaida
Procurando pelo nome Zenaida

OPS, algo deu errado, nao existe o nome Zenaida...

```

1. A bateria de testes da seção (III) é semelhante à anterior: Alguns registros são inseridos no arquivo e, em seguida, exibidos na tela de forma sequencial.

Caso 1: Inserção e Mostra de Registros

```
-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----
Digite o digito:0
Arquivo criado com sucesso

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----
Digite o digito:1
Digite a entrada da seguinte forma:
Chave Name lname City State Zip-Code Telefone Numero Rua Complemento
->195 Mia Conway Sobral SC 61367-737 (89)31822-8825 9273 Julio_Landing Apt_372

Registro adicionado ao arquivo arquivo.bin

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----
Digite o digito:2

Registros do arquivo arquivo.bin
-----Registro: 0-----
#|1|140|
KEY = 195
LAST NAME = Conway
FIRST NAME = Mia
CITY = Sobral
STATE = SC
CEP = 61367-737
PHONE = (89)31822-8825
ENDERECO = Julio_Landing,9273,Apt_372
-----
```

2. Outros testes também foram feitos com as operações *busca_key* e *busca_name*, as quais simbolizam a busca tanto usando chave primária quanto secundária:

Caso 2: Busca por Chave Primária e Chave Secundária (Nome)

```
-----Registro: 2-----
#|1|141|
KEY = 245
LAST NAME = Watts
FIRST NAME = Fay
CITY = Sobral
STATE = SC
CEP = 24937-515
PHONE = (40)05787-6013
ENDERECO = Pollich_Hills,71782,Suite_335
-----
-----Registro: 3-----
#|1|156|
KEY = 647
LAST NAME = Mcmillan
FIRST NAME = Regina
CITY = Ribeirão_Preto
STATE = CE
CEP = 83857-374
PHONE = (26)88234-1645
ENDERECO = Shannon_Court,3037,Suite_356
-----
-----Registro: 4-----
#|1|150|
KEY = 873
LAST NAME = Miller
FIRST NAME = Joan
CITY = Petrópolis
STATE = PA
CEP = 58128-105
PHONE = (67)07144-2986
ENDERECO = Gottlieb_Shoals,4545,Suite_972
-----
-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----
Digite o digito:3
Digite o numero da chave:873
-----
```

```

#|1|150|
KEY = 873
LAST NAME = Miller
FIRST NAME = Joan
CITY = Petrópolis
STATE = PA
CEP = 58128-105
PHONE = (67)07144-2986
ENDereco = Gottlieb_Shools,4545,Suite_972
-----

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----

Digite o digito:4
Digite o primeiro nome:Regina
Procurando pelo nome Regina

Registro(s) com o nome: Regina abaixo:
-----

#|1|156|
KEY = 647
LAST NAME = Mcmillan
FIRST NAME = Regina
CITY = Ribeirão Preto
STATE = CE
CEP = 83857-374
PHONE = (26)88234-1645
ENDereco = Shannon_Court,3037,Suite_356
-----

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----

Digite o digito:5
Digite a chave para remover:647

```

3. Por fim, também foi testada a remoção de um registro de tamanho 156 bytes - e em seguida a inserção de um registro de tamanho menor. Ao final da execução é exibida a lista ordenada de chaves secundárias com os bytes offset.

Caso 3: Remoção e inserção de registro com aproveitamento de espaço

```
#|1|150|
KEY = 873
LAST NAME = Miller
FIRST NAME = Joan
CITY = Petrópolis
STATE = PA
CEP = 58128-105
PHONE = (67)07144-2986
ENDEREÇO = Gottlieb_Shoals,4545,Suite_972
-----

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----

Digite o digito:4
Digite o primeiro nome:Regina
Procurando pelo nome Regina

Registro(s) com o nome: Regina abaixo:
-----

#|1|156|
KEY = 647
LAST NAME = Mcmillan
FIRST NAME = Regina
CITY = Ribeirão Preto
STATE = CE
CEP = 83857-374
PHONE = (26)88234-1645
ENDEREÇO = Shannon_Court,3037,Suite_356
-----

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----

Digite o digito:5
Digite a chave para remover:647
```



```
-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----
Digite o digito:2

Registros do arquivo arquivo.bin
-----Registro: 0-----
#|1|140|
KEY = 195
LAST NAME = Conway
FIRST NAME = Mia
CITY = Sobral
STATE = SC
CEP = 61367-737
PHONE = (89)31822-8825
ENDERECO = Julio_Landing,9273,Apt_372
-----
-----Registro: 1-----
#|1|150|
KEY = 767
LAST NAME = Dennis
FIRST NAME = Zenaida
CITY = Novo_Hamburgo
STATE = CE
CEP = 22713-313
PHONE = (41)01716-6435
ENDERECO = Neva_Springs,9280,Apt_146
-----
-----Registro: 2-----
#|1|141|
KEY = 245
LAST NAME = Watts
FIRST NAME = Fay
CITY = Sobral
STATE = SC
CEP = 24937-515
PHONE = (40)05787-6013
ENDERECO = Pollich_Hills,71782,Suite_335
-----
```

```

-----Registro: 3-----
#|1|150|
KEY = 873
LAST NAME = Miller
FIRST NAME = Joan
CITY = Petrópolis
STATE = PA
CEP = 58128-105
PHONE = (67)07144-2986
ENDERECO = Gottlieb_Shoals,4545,Suite_972
-----

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----

Digite o digito:1
Digite a entrada da seguinte forma:
Chave Name lname City State Zip-Code Telefone Numero Rua Complemento
->481 Hedda Gill Sousa GO 88155-383 (27)57813-2087 4235 Laury_Drive Suite_713

Registro adicionado ao arquivo arquivo.bin

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----

Digite o digito:2

Registros do arquivo arquivo.bin
-----Registro: 0-----
#|1|140|
KEY = 195
LAST NAME = Conway
FIRST NAME = Mia
CITY = Sobral
STATE = SC
CEP = 61367-737
PHONE = (89)31822-8825
ENDERECO = Julio_Landing,9273,Apt_372
-----

```

```

-----Registro: 1-----
#|1|150|
KEY = 767
LAST NAME = Dennis
FIRST NAME = Zenaida
CITY = Novo_Hamburgo
STATE = CE
CEP = 22713-313
PHONE = (41)01716-6435
ENDEREÇO = Neva_Springs,9280,Apt_146
-----
-----Registro: 2-----
#|1|141|
KEY = 245
LAST NAME = Watts
FIRST NAME = Fay
CITY = Sobral
STATE = SC
CEP = 24937-515
PHONE = (40)05787-6013
ENDEREÇO = Pollich_Hills,71782,Suite_335
-----
-----Registro: 3-----
#|1|139|
KEY = 481
LAST NAME = Gill
FIRST NAME = Hedda
CITY = Sousa
STATE = GO
CEP = 88155-383
PHONE = (27)57813-2087
ENDEREÇO = Laury_Drive,4235,Suite_713
-----
-----Registro: 4-----
#|1|150|
KEY = 873
LAST NAME = Miller
FIRST NAME = Joan
CITY = Petrópolis
STATE = PA
CEP = 58128-105
PHONE = (67)07144-2986
ENDEREÇO = Gottlieb_Shoals,4545,Suite_972
-----

```

```

-----MENU-----
0)Criar arquivo
1)Inserir
2)Mostrar todos os registros
3)Mostrar registro com a KEY x
4)Mostrar registro com FNAME s
5)Remover registro
6)Sair
-----
Digite o digito:6

Encerrando o programa...
|1| 306 Fay
-1
|1| 455 Hedda
-1
|1| 619 Joan
-1
|1| 0 Mia
-1
|1| 148 Zenaida
-1

```

A bateria de testes completa está disponível em [1], nas pastas “T1A/Prints_Execucao” e “T1B/Prints_Execucao”.

(VII) Referências

[1] Orlando, G., & Moreira, E. (2021). Organização e Recuperação da Informação (ENPE 3 - 2021): Trabalho Prático 1 (Version 1.0.0) [Computer software]. <https://github.com/ericpqmor/ori-1>