

# SCC0203 - ALGORITMOS E ESTRUTURAS DE DADOS II

## TRABALHO PRÁTICO N. 2

### ARMAZENAMENTO SECUNDÁRIO E ÍNDICES

Profa. Débora Medeiros

#### 1. Descrição do Trabalho

Uma academia deseja manter um cadastro de seus clientes para enviar notícias, avisos, promoções etc., direcionados para grupos de clientes. Os grupos podem ser de clientes que treinam a mesma modalidade ou clientes do mesmo sexo. A academia já possui um servidor onde cada cliente possui um login associado a uma conta de email. Dessa forma, para cada cliente, a academia deseja manter os seguintes atributos:

- login no servidor da academia (chave primária);
- modalidade (musculação, aeróbica, lutas, etc) e
- sexo (f/m)

Você deverá desenvolver um sistema que permita à academia:

1. Inserir um novo cliente;
2. buscar clientes por modalidade;
3. buscar clientes por sexo e
4. buscar clientes por modalidade **e** sexo (match).

Para isso, devem ser utilizados 2 índices secundários (modalidade e sexo), 1 primário (login, que será consultado pelos índices secundários). Na próxima seção, será apresentado como devem ser os comandos correspondentes às funcionalidades requeridas. A Seção 3 contém a especificação dos índices e do arquivo de dados e a Seção 4 finaliza a descrição desse trabalho com um exemplo execução do sistema.

## 2. Comandos e saídas

Os comandos são representados por duas letras maiúsculas. Toda linha de entrada obrigatoriamente inicia com um comando. Só serão fornecidos como entrada comandos aqui especificados (não há necessidade de tratar comandos inválidos). O símbolo `_` denota um espaço em branco (nenhum dos campos conterá espaços).

### Inserir cliente

```
IC_<login>_<modalidade>_<sexo>
```

### Buscar clientes por modalidade

```
BM_<modalidade>
```

### Buscar clientes por sexo

```
BS_<sexo>
```

### Buscar clientes por modalidade e tipo

```
BD_<modalidade>_<sexo>
```

### Saída para cada busca (sem linhas em branco)

```
<número de resultados>  
<login>_<modalidade>_<sexo>  
...  
<login>_<modalidade>_<sexo>
```

### Finaliza a execução<sup>1</sup>

```
FM
```

Todas as chaves devem ser convertidas para a seguinte forma canônica: letras minúsculas sem acentos, nem cedilhas etc.

---

<sup>1</sup> Finaliza a execução do programa, gravando os arquivos de índice e dados no disco e fechando-os.

### 3. Arquivos de índices e dados

O processamento do arquivo de dados DEVE SER REALIZADO EM MEMÓRIA SECUNDÁRIA, já os arquivos de índice PODEM SER PROCESSADOS EM MEMÓRIA PRINCIPAL. Ao final da execução, deverão existir 4 arquivos (1 arquivo de dados, 1 arquivo de índice primário, 2 arquivos de índice secundário)<sup>2</sup>:

- **Dados:** deve ser ASCII e organizado em registros de TAMANHO VARIÁVEL, contendo o tamanho do registro (dois primeiros caracteres do registro indicando número total de caracteres do registro), login, modalidade e sexo. Este arquivo é *entry sequenced*, ou seja, os registros são adicionados sempre no final do arquivo. Os campos no registro devem ser separados pelo caractere delimitador | (pipe) (1 para cada campo, mesmo o último). Como o número de campos é fixo, não deve ser utilizado caractere separador de registros. Por exemplo:

tam1login1|modalidade1|sexo1|tam2login2|modalidade2|sexo2|...

**Obrigatoriamente, o arquivo de dados deverá ser chamado de “data.dat”.**

- **Índice primário:** deve ser binário e armazenar registros com 2 campos, um para chave primária e outro para o offset no arquivo de registros. Este índice deve ser MANTIDO ORDENADO por chave primária. A chave primária deve ser declarada como um vetor de 21 caracteres (20 para o login, 1 para o ‘\0’). O endereço do registro (offset), deve ser do tipo UNSIGNED INT. **Obrigatoriamente, o arquivo de índice primário deverá ser chamado de “index.dat”.**
- **Índice secundário (modalidade):** deve ser binário e armazenar registros com 2 campos, um para a chave secundária e outro para um vetor ordenado de chaves primárias<sup>3</sup>. Este índice deve ser MANTIDO ORDENADO por chave secundária. A chave secundária deve ser declarada como um vetor de 21 caracteres (20 para a modalidade, 1 para o ‘\0’). O campo das chaves primárias deve ser um vetor de 15 chaves primárias ou lista (ORDENADOS) representadas da mesma forma que no índice primário. **Obrigatoriamente, o arquivo de índice secundário de modalidade deverá ser chamado de “index1.dat”.**
- **Índice secundário (sexo):** deve ser binário e armazenar registros com 2 campos, um para a chave secundária e outro para um vetor ordenado de chaves primárias<sup>3</sup>. Este índice deve ser MANTIDO ORDENADO por chave secundária. A chave secundária é representada por 1 caractere. O campo das chaves primárias deve ser um vetor de 50 chaves primárias ou lista (ORDENADOS) representadas da mesma forma que no índice primário. **Obrigatoriamente, o arquivo de índice secundário de tipo deverá ser chamado de “index2.dat”.**

<sup>2</sup> Nenhum dos arquivos deverá ter cabeçalho.

<sup>3</sup> Supõe-se que, na academia, o número de alunos por modalidade e por sexo seja balanceado e exista uma quantidade limitada de vagas.

#### 4. Exemplo de execução

A seguir é apresentado um exemplo de entrada e saída seguindo o padrão apresentado:

Entrada	Saída										
IC ana aeróbica f IC joao musculação m IC maria musculação f IC jose natação m BM musculação BM lutas BD musculação M FM	2 joao musculacao m maria musculacao f 0 1 joao musculacao m										
Arquivo final de índice <sup>4</sup> secundário - modalidade (representação lógica)	Arquivo final de índice <sup>4</sup> secundário - sexo (representação lógica)										
<table border="1"> <tr> <td>aerobica</td> <td>ana</td> </tr> <tr> <td>musculacao</td> <td>joao maria</td> </tr> <tr> <td>natacao</td> <td>jose</td> </tr> </table>	aerobica	ana	musculacao	joao maria	natacao	jose	<table border="1"> <tr> <td>f</td> <td>ana maria</td> </tr> <tr> <td>m</td> <td>joao jose</td> </tr> </table>	f	ana maria	m	joao jose
aerobica	ana										
musculacao	joao maria										
natacao	jose										
f	ana maria										
m	joao jose										
Arquivo final de índice <sup>4</sup> primário	Arquivo final de dados										
<table border="1"> <tr> <td>ana</td> <td>0</td> </tr> <tr> <td>joao</td> <td>17</td> </tr> <tr> <td>jose</td> <td>58</td> </tr> <tr> <td>maria</td> <td>37</td> </tr> </table>	ana	0	joao	17	jose	58	maria	37	17ana aerobica f 20joao musculacao m  21maria musculacao f 17jose natacao m		
ana	0										
joao	17										
jose	58										
maria	37										

#### 5. Observações importantes

- O programa deverá respeitar exatamente os padrões de entrada e saída, pois a correção será parcialmente realizada de maneira automática;
- Os códigos serão verificados para se detectar ocorrências de plágio.
- Implemente usando a linguagem C.
- Vazamento de memória, referência a valores de variáveis não inicializadas e outros defeitos serão levados em conta na avaliação;
- Os trabalhos deverão ser feitos em duplas.
- Elabore um RELATÓRIO (envie preferencialmente em formato PDF) que discuta em linhas gerais a estrutura lógica usada para desenvolver o trabalho, sem apresentar código. **Não** entregue o código com o relatório. O relatório deve permitir compreender o trabalho sem a necessidade do código. Entregar o relatório pelo mesmo endereço de entrega do trabalho (a ser especificado posteriormente).
- Trabalhos estão sujeitos a avaliação oral segundo critério do professor e do estagiário PAE.
- Serão anulados aqueles trabalhos nos quais forem detectados quaisquer tipos de cópia ou plágio, não importa a origem.
- O sistema estará aberto para submissões a partir do dia 24/05/2011.
- A data limite para entrega é 06/06/2011.

<sup>4</sup> Arquivo binário.