

MAC0121/0122 - 2025

Exercício de programação 3

22 de novembro de 2025

Este EP é uma adaptação do EP5 de MAC0121 em 2019 (oferecida pela professora Cristina Gomes Fernandes), que por sua vez é uma adaptação feita pelo professor José Coelho de Pina, do EP5 de MAC0122 em 2005 de autoria do professor José Augusto Ramos Soares e do EP5 de MAC0122 em 2012 de autoria do professor Carlos Hitoshi Morimoto. O enunciado da adaptação pode ser encontrado aqui.

1 Introdução

A grande quantidade de streamings na atualidade fornece uma vasta gama de opções para assistir filmes, no conforto do lar. Porém, dada a necessidade das empresas de preencherem seus catálogos, muitos filmes disponibilizados por esses serviços não valem muito a pena. Evidentemente, as sinopses não ajudam muito, já que as empresas querem te convencer a consumir esses produtos. Pois bem, seus problemas acabaram!! Neste EP vamos fazer um programa que imprime notas para filmes usando informações disponíveis na Internet!

Para isso vamos implementar um programa que lê de um (ou mais) arquivo(s) informações sobre filmes e permite fazer operações como consulta, ordenação, inserção e remoção de filmes, etc.

A ideia é que seu programa leia as notas dos filmes de arquivos. Esses arquivos são disponibilizados e estão no formato utilizado pelo site *The Internet Movie Database* (IMBD). O IMDB coleta essas informações a partir da participação de vários espectadores que se dispuseram a dar notas aos filmes.

1	0000000233	419191	8.7	Cidade de Deus (2002)
2	7000000.00	231	1.5	Xuxa Abracadabra (2003)

A primeira linha indica que 419191 pessoas votaram no filme, a média das notas foi 8,7, o nome do filme é Cidade de Deus e o ano do filme é 2002. Cada nota de um votante é um inteiro de 1 até 10. O string 0000000233 indica a distribuição de votos. Cada caractere representa a porcentagem de votos para cada possível nota, de acordo com a seguinte descrição.

1	"."	nenhum voto	"3"	30-39% dos votos	"7"	70-79% dos votos
2	"0"	1-9% dos votos	"4"	40-49% dos votos	"8"	80-89% dos votos
3	"1"	10-19% dos votos	"5"	50-59% dos votos	"9"	90-99% dos votos
4	"2"	20-29% dos votos	"6"	60-69% dos votos	"*"	100% dos votos

Portanto, no caso de Cidade de Deus, entre 30 a 39% dos votos foram 10, enquanto que entre 70 a 79% dos votos da Xuxa foram 1. Mas como o filme da Xuxa recebeu apenas 231 votos, fica a seu critério confiar ou não nessa nota.

Um esqueleto para você utilizar como ponto de partida para o EP3 está disponível no e-disciplinas. Essencialmente, o esqueleto é formado por:

- um programa principal parcialmente feito (`main.c`, `main.h`);
- funções de entrada e saída do EP5 (`iofilmes.c`, `iofilmes.h`);
- funções que manipulam listas de filmes (`filmes.c`, `filmes.h`);
- (OPCIONAL) funções que manipulam uma tabela de símbolos (`st.c`, `st.h`);
- funções de uso geral (`util.c`, `util.h`): uma função de nome `leiaString` que pode ser usada para ler cadeias de caracteres como, por exemplo, nomes de filmes; uma função de nome `strCmp` que pode ser utilizada na ordenação dos filmes por nome; etc.

2 Comportamento do Programa:

Veja os detalhes no arquivo `comportamentoFuncoes.pdf`.

3 Estruturas

1. **Lista de filmes:** Os dados de cada filme devem ser armazenados em uma lista duplamente encadeada circular com cabeça.

Os nomes dos filmes devem ser lidos com no máximo `TAM_STR` (`util.h`) caracteres. Suponha que `lst` é um objeto do tipo `ListaFilmes`. Assim, a célula cabeça é `lst->cab`. A primeira célula da lista é `lst->cab->prox` e a última é `lst->cab->ant`. Quando a lista `lst` está vazia, temos que

```
1 lst->cab->prox == lst->cab  &&  lst->cab->ant == lst->cab
```

ou, de maneira equivalente,

```
1 lst->nFilmes == 0
```

Veja mais detalhes da estrutura no arquivo `filmes.h`.

2. **Tabela de símbolos (opcional)** O programa utilizará uma tabela de símbolos que tem como chaves as palavras que ocorrem nos nomes dos filmes e como itens filmes cujo título que contém as palavras.

A função `hash()` que calcula o código de dispersão (= hash code) de cada palavra deverá ser definida no arquivo `st.c`. As colisões serão resolvidas através de separate chaining: para cada código de dispersão ou índice `h` da tabela, há uma lista encadeada que armazena palavras (chaves) em nomes de filmes que a função de dispersão leva em `h`. A célula que representa cada palavra/chave possui um ponteiro `iniListaPtr*` para a lista ligada dos (ponteiros para os) filmes que possuem a chave no seu nome.

A interface dessa implementação se encontra em `st.h` e a correspondente implementação em `st.c`.

4 Formato do arquivo de entrada

O arquivo com os dados de entrada tem, além dos dados a serem lidos, várias linhas com comentários que devem ser ignorados. As linhas que devem ser lidas começam com 6 espaços em branco () seguidos de um dígito ('0'..'9') ou do caractere ponto ('.').

Cada linha com uma avaliação de um filme contém, na ordem,

1. 6 espaços em branco;
2. 10 caracteres representando a distribuição dos votos do filme (por exemplo, 0000000133);
3. um inteiro com o número de votantes (por exemplo, 279506);
4. um float representa a nota do filme (por exemplo, 8.7);
5. uma string com o nome do filme (por exemplo, `Cidade de Deus`); e
6. um inteiro entre parênteses, que é o ano do filme (por exemplo, 2002).

Podem aparecer filmes duplicados no arquivo de entrada. Por exemplo, a seguinte entrada aparece duplicada.

```
1      0000000124  72392  8.8  Casablanca (1942)
2      0000000124  72392  8.8  Casablanca (1942)
```

Somente uma entrada para cada filme deve ser inserida na lista pela função de leitura. Para isto consideraremos que **dois filmes são iguais** se eles têm a mesma nota, nome e ano.

Assim, devemos observar que os seguintes filmes são diferentes.

```
1      0000000124  72392  8.8  Casablanca (1942)
2      4...0..03    11  5.5  Casablanca (1961)
3      1.000.0015   36  4.9  Casablanca (2002)
4      0.0.021210   60  7.0  Casablanca 50th Anniversary Special: You Must Remember This (1992) (
5      0000212000   71  5.6  Casablanca Driver (2004)
6      2121000000   84  3.4  Casablanca Express (1989)
7      3...11.1.2    8  5.4  Casablanca revisitada (1992)
8      ...021310.   25  5.9  Casablanca, Casablanca (1985)
9      2..111..02   13  5.2  Casablanca, nid d'espions (1963)
10     .1...133.1    9  6.3  Casablancais, Les (1999)
11     00.1023000   55  6.2  Cirkus Casablanca (1981)
12     4.....1..4    7  5.3  Juego sucio en Casablanca (1985)
13     0000012211   802  7.0  Night in Casablanca, A (1946)
```

Para complicar um pouco a vida, às vezes a linha termina com outro par de parênteses com uma das palavras: mini, TV ou V, ou com comentários entre chaves (caracteres " e "). Essa parte deve ser ignorada.

Outro problema: às vezes o ano é desconhecido e é representado por "????", como mostra o exemplo abaixo.

```
1      2000001002    107  4.3  "Gigolos" (????)
```

Vocês não precisam se preocupar com esses problemas, pois a função para a leitura desse arquivo (`carregueListaFilmes`) é totalmente fornecida em `iofilmes.c`.

5 Funções

1. **Protótipos das funções fornecidas e parcialmente fornecidas:** O comportamento das funções fornecidas ou parcialmente fornecidas e que você utilizará em algum dos trechos de código que deverá escrever está descrito nos comentários que precedem as funções nos arquivos do esqueleto.
2. **Protótipos das funções que você deve implementar:** O comportamento das funções que você deverá escrever está descrito nos comentários que precedem as funções nos arquivos do esqueleto ou ainda no meio da função, como no caso das funções `main()` no `main.c`. Você (sempre!) pode escrever mais funções auxiliares se achar necessário.

6 Descrição dos arquivos do EP

Aqui temos uma visão geral dos arquivos que compõem o EP. Você deverá depositar na página de MAC0121 apenas um arquivo `tar.gz` contendo todos os arquivos com extensão `.h` e `.c` do seu EP5.

Este exercício-programa é formado por 10 arquivos. O diretório com o esqueleto do EP contém, além desses 10 arquivos, um Makefile para gerar o executável `imdb` do EP e `.zip` com todos os arquivos do EP. Para compilar o programa e gerar o executável basta, na linha de comando, digitar **make**.

Vamos às descrições dos arquivos:

1. **main.c:** Contém a função `main()` que está parcialmente escrita.
2. **main.h:** Contém apenas a declaração de algumas constantes utilizadas pela função `main()`. Não modifique.
3. **iofilmes.c:** Contém as definições de funções de entrada e saída. Algumas funções deste módulo estão por fazer.
4. **iofilmes.h:** É a interface para as funções de entrada e saída do módulo `iofilmes.c`. Não modifique.
5. **filmes.c:** Deve ser o arquivo principal deste EP com as implementações de funções de criam, comparam, buscam, inserem, removem e ordenam os objetos do filmes (tipo Filme) e lista de filmes (tipo ListaFilmes).
6. **filmes.h:** É a interface para as funções em `filmes.c`. Não modifique.
7. **util.c:** Contém as definições das funções de uso geral declaradas em
8. **util.h:** Para achar um filme com nome contendo o “string” definido pelo usuário, você deve implementar a função `achePalavra()`. Está é a única função deste arquivo que ainda precisa ser escrita.
9. **util.h** Interface para `util.c`.
10. **st.c** Contém as definições das funções que manipulam a tabela de símbolos.
11. **st.h** Interface para `st.c`.
12. **Makefile** Para compilar e gerar o executável `imdb` do EP5 basta digitar **make** no prompt.

7 Entrega

Você deve entregar os arquivos 1–11 listados na Seção 6. Esses arquivos devem estar dentro de um diretório cujo nome é o seu NUSP. Compacte o diretório e envie um **único arquivo** no e-disciplinas com extensão **.tar.gz**.

Envios com extensão incorreta ou que gerem diretório com nome distinto do solicitado terão desconto de 1 ponto na nota final. Similarmente, **tar** bombs, arquivos além dos solicitados, etc, poderão acarretar desconto na nota.

Como mencionado, a implementação da tabela de símbolos é **opcional**. Contudo, mesmo que você opte por não implementar, envie os arquivos **st.c** e **st.h**. A implementação da tabela de símbolos vale 5 pontos na nota do EP. Portanto, se você implementar corretamente tudo que foi solicitado, a nota final do EP será 15.