

UFES Music

Aluno: Gabriel Silva Simoura

Disciplina: Tópicos Especiais em Programação I

Professor: Rodrigo Laiola Guimarães

1. Estruturas de dados utilizadas

1.1) Mídias:

```
struct midia{
    int tipo;
    int qtdArtista;
    int qtdCompositor;
    char *nomeMidia;
    char *nomeArtista[2];
    char *nomeCompositor[2];
    char *genero;
    char *produzidoPor;
    float duracao;
};
```

- I. **tipo:** Define se é uma música (01), vídeo(02) ou um podcast(03);
- II. **qtdArtista:** Define a quantidade de compositores existentes na Mídia;
- III. **nomeMidia:** String que contém o nome da Mídia;
- IV. **nomeArtista[2]:** Vetor de String que contém os nomes dos artistas;
- V. **nomeCompositor[2]:** Vetor de String que contém os nomes dos compositores;
- VI. **genero:** String que contém o gênero da Mídia;
- VII. **produzidoPor:** String que contém o nome da produtora/gravadora;
- VIII. **duracao:** Float contendo a duração da Mídia;

No TAD Midia há funções que: Alocam um ponteiro para a struct mídia, imprime o conteúdo e os dados de uma mídia, cria uma mídia, fazem atribuições de dados da mídia, nome, duração, nome de artistas, etc. Da mesma forma, há funções que pesquisam por informações da mídia, deletam a mídia do programa, que retornam dados da mídia, e que fazem a leitura e a escrita dos dados da mídia para um arquivo;

1. 2) Album:

```
struct album{
```

```

char *nomeAlbum;
char *nomeArtista;
char *genero;
char *produzidoPor;
Midia *midia[21];
int qtdMidias;
int qtdMidiasTotal;
int ano;
};

```

- I. **nomeAlbum**: String que contém o nome do Álbum;
- II. **nomeArtista**: String que contém o nome do artista Álbum;
- III. **genero**: String que contém o gênero do Álbum;
- IV. **produzidoPor**: String que contém o nome da produtora/gravadora;
- V. **midia[21]**: Vetor de ponteiros para a struct mídia;
- VI. **qtdMidias**: Inteiro que contém a quantidade de mídias atuais;
- VII. **qtdMidiasTotais**: Inteiro que contém a quantidade de mídias totais;
- VIII. **ano** : Inteiro que contém o ano de lançamento do álbum;

No TAD Álbum há funções que: Alocam um ponteiro para a struct álbum, imprime o conteúdo e os dados de um Álbum, cria um álbum, fazem atribuições de dados do mesmo, nome, duração, nome de artistas, etc. Da mesma forma, há funções que pesquisam por informações do Álbum, que retornam dados do Álbum, e que fazem a leitura e a escrita dos dados do Álbum para um arquivo;

1. 3) Playlist :

```

struct playlist{
    int tipo; // (0)--> publica e (1)--> privada;
    char *nomePlaylist;
    char *colaboradores[2];
    int qtdColaboradores;
    int qtdMidias;
    Midia *midia[51];
};

```

- I. **tipo**: Inteiro que define se a playlist é (0) pública ou (1) privada;
- II. **nomePlaylist**: String que contém o nome da Playlist;
- III. **colaboradores[2]**: Vetor de String que contém o nome dos colaboradores da playlist;
- IV. **qtdColaboradores**: Inteiro que contém a quantidade de colaboradores da playlist;

- V. **qtdMídias:** Inteiro que contém a quantidade de mídias atuais;
- VI. **mídia[51]:** Vetor de ponteiros para structs mídia;

No TAD Playlist há funções que: Alocam um ponteiro para a struct playlist, imprime o conteúdo e os dados de uma playlist, cria uma playlist, fazem atribuições de dados da mesma, nome, duração, nome de colaboradores, mídias, etc. Da mesma forma, há funções que trocam posições das mídias na playlist, que apagam mídias, e que fazem a leitura e a escrita dos dados da Playlist para um arquivo;

1. 4) Usuário:

```
struct usuario{
    int idusuario;
    int favorito;
    int qtdTotalPlaylist;
    int qtdplay;
    int qtdPseguindo;
    char *login;
    char *senha;
    Playlist *play[51];
    Playlist *favoritos;
    Playlist *seguindo[5];
};
```

- I. **idusuario:** Inteiro contendo o identificador do usuário;
- II. **favorito:** Inteiro verificando se já foi criado ou não uma playlist favorita;
- III. **qtdTotalPlaylist:** Inteiro contendo a quantidade máxima de playlists
- IV. **qtdplay:** Inteiro contendo a quantidade de playlists atuais;
- V. **qtdPseguindo:** Inteiro contendo a quantidade de playlists seguidas;
- VI. **login:** String contendo o login do usuário;
- VII. **senha:** String contendo a senha do usuário;
- VIII. **play[51]:** Vetor de ponteiros para a struct Playlist;
- IX. **favoritos:** ponteira para a struct Playlist contendo as mídias favoritas;
- X. **seguindo[5]:** ponteiros para a struct Playlist que estão sendo seguidas pelo usuário;

No TAD Usuário há funções que: Alocam um ponteiro para a struct Usuário, imprime o conteúdo e os dados de um Usuário, cria um Usuario, fazem atribuições de dados do mesmo, login e senha etc. Da mesma forma, há funções que apagam e adicionam playlists do Usuário, Trocam posições das playlist na conta do Usuário, que retornam informações do Usuário, e que fazem a leitura e a escrita dos dados do Usuário para um arquivo;

1. 5) Sistema:

```

struct sistema{
    Album *albuns[11];
    Usuario *usuarios[11];
    int qtdUsuario;
    int qtdAlbuns;
    int senhaDev;
};

```

- I. albuns[11]:** Vetor de ponteiros para albuns do programa;
- II. usuarios[11]:** Vetor de ponteiros para usuarios do programa;
- III. qtdUsuario:** Inteiro com a quantidade de usuários atual do programa;
- IV. qtdAlbuns:** Inteiro com a quantidade de Álbuns atual do programa;
- V. senhaDev:** Senha do Desenvolvedor do Programa, para poder fazer edições, adicionar álbuns, editar nomes, etc; SENHA= 130511;

No TAD Sistema há funções que: Alocam um ponteiro para a struct Sistema, imprime o conteúdo e os dados do Sistema, fazem atribuições de dados do mesmo, quantidade de albuns, usuários, playlists, álbuns, etc. Da mesma forma, há funções que retornam informações do Sistema, que fazem a pesquisa por informações no programa e que fazem a leitura e a escrita dos dados do Sistema para um arquivo;

-- menu.c/menu.h : arquivos que funcionam para Imprimir o menu em si para o usuário/desenvolvedor do programa;

--- programa.txt: arquivo onde é feita a leitura e a escrita de dados do programa;