Universidade Federal do Rio de Janeiro
Escola Politécnica
Departamento de Engenharia Eletrônica e de Computação
Engenharia Eletrônica e de Computação
Linguagens de Programação

TRABALHO DE LINGUAGENS DE PROGRAMAÇÃO

Alunos: Jose Guilherme Silva de Macedo e Karen dos Anjos Arcoverde

Professor: Miguel Elias Mitre Campista

Rio de Janeiro 2021

Sumário

6	Conclusão	7
5	Casos de Uso	5
4	Implementação do programa 4.1 Classes e funções implementadas	4 4 4 5
	Observações	4
2	Sobre o programa	3
1	Objetivo do programa	3
0	Introdução	3

0 Introdução

Esse relatório se refere a uma explicação de como funciona o sistema de uma biblioteca da Eletrônica e de Computação, contendo o programa desenvolvido, a sua implementação e o seu uso.

1 Objetivo do programa

O objetivo do programa é tentar mostrar como funcionaria um gerenciamento de uma biblioteca da Eletrônica e de Computação, usando duas linguagens de programação, C++ e SQL. O programa simula como acontece um empréstimo e uma devolução de livro, acesso ao acervo de livros, adição e remoção de livros, entre outros. Portanto, são usadas funções externas com o uso de SQL que fazem esse gerenciamento dos livros, em que o C++ será apenas o intermediário entre o usuário e as funções externas.

Na figura abaixo, é ilustrada a forma como isso acontece:

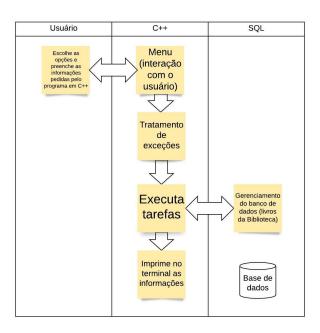


Figura 1: Esquema do programa construído.

2 Sobre o programa

O programa implementa o gerenciamento de livros de Eletrônica e de Computação de forma que o usuário possa escolher o que ele deseja fazer. Dessa forma, a cada escolha do usuário, uma função externa feita no ambiente SQL é chamada para executar tal operação.

O programa possui arquivos ".sql" que executam comando no ambiente SQL para fazer alterações no banco de dados. Caso não exista a tabela, o programa já cria uma tabela com as colunas já definidas para que o usuário não precise ter o trabalho de criar manualmente.

Ademais, o usuário poderá escolher a opção de inserir um livro ou remover um livro do banco de dados (biblioteca), pode pesquisar por um livro pela palavra-chave ou pelo gênero ou pelo email da pessoa que fez o empréstimo. Também pode realizar empréstimo ou devolução de um livro e ainda pode escolher para mostrar todos os livros e suas informações que estão no banco de dados (título do livro, autor do livro, gênero do livro - Eletrônica analógica, Eletrônica digital, Matemática, Computação etc., status do livro - Disponível ou Indisponível, alugado, e e-mail da pessoa que alugou. Quando algum livro for removido,

inserido ou alugado, a impressão do banco de dados já mostra as alterações feitas anteriormente pelo usuário.

Cabe ressaltar que a cada interação do usuário, os valores introduzidos por ele são verificados caso não corresponda aos pedidos.

3 Observações

Durante a construção do código, foi visto a necessidade de mostrar a tabela do banco de dados no terminal para o usuário, então foi acrescentada uma opção no menu que executa essa tarefa.

Além disso, foi acrescentada uma busca do livro por palavra, caso a palavra digitada por usuário contenha em algum título do livro, é mostrada na tela todos os livros que contém essa palavra. Caso não tenha a palavra em algum livro, nenhum livro é mostrado. Também foram adicionadas busca por e-mail e gênero, mas no caso dessas palavras chave, a palavra digitada deve ser exatamente a palavra que consta no banco de dados para que a busca tenha sucesso.

4 Implementação do programa

Nesta seção serão apresentadas as funções e as classes utilizadas.

4.1 Classes e funções implementadas

4.1.1 Library

Atributos:

- 1. sqlite3* library: Ponteiro para um objeto da estrutura sqlite3. Esta estrutura é definida na biblioteca sqlite3.h. Ela possibilita a chamada de funções que se relacionam com o ambiente SQL.
- 2. sqlite3_stmt* stmt: Uma instância do objeto stmt, *statement*, é simplesmente uma declaração de código SQL que já foi compilado para forma binária e está pronto para ser executado.

Métodos:

- void criaTabela(): Cria a tabela, caso não exista, com as seguintes colunas: Título, Autor, Gênero, Status e Email. Caso a tabela já exista, a tabela existente não é modificada. Essa função utiliza o arquivo "cria_tabela.sql" para criar a tabela para o banco de dados.
- 2. string abreSQL(string): Abre os arquivos ".sql"e retorna o que está escrito nele.
- 3. void inserirLivro (string, string, string, string): Insere um livro no banco de dados e suas informações de título, autor e gênero escolhidas pelo usuário. Caso o livro já exista na Biblioteca, uma mensagem de erro é retornada para o usuário.
- 4. void removerLivro (string): Remove um livro no banco de dados escolhido pelo usuário. Caso o livro não exista na Biblioteca, uma mensagem de erro é retornada para o usuário.
- 5. void mostraLivros (): Imprime na tela o conteúdo da Biblioteca.
- 6. bool pesquisaNomeLivro (string, bool): Pesquisa um determinado título de livro que foi fornecido pelo usuário do tipo string na Biblioteca e retorna uma variável booleana do tipo true livro existe, false livro não existe. Além disso, o segundo argumento indica se a impressão do(s) livro(s) procurado(s) será realizada ou não.

- 7. void pesquisaNomeLivroLike (string): Pesquisa os títulos do livro que contenham a palavra-chave escolhida pelo usuário e imprime na tela. Caso o usuário digite uma palavra que não esteja em nenhum título de livro, nada é impresso na tela para evitar erros.
- 8. void pesquisaGenero (string): Pesquisa livros que pertençam ao gênero fornecido pelo usuário em forma de string. Esses livros são impressos na tela.
- 9. void pesquisa Email (string): Pesquisa livros que foram emprestados ao usuário da biblioteca cujo e-mail foi digitado por quem está manipulando o programa. Os livros encontrados são impressos na tela.
- 10. void updateEmail(string, string): Método privado que atualiza o e-mail e o status de determinado livro. Esse método é usado pelos métodos de realização e devolução de empréstimos.
- 11. void realizarEmprestimo(string, string): Esse método recebe as strings do usuário em relação ao pedido de empréstimo e após validar os dados, ela chama o método updateEmail().
- 12. void realizarDevolucao(string): Análogo ao método anterior, mas validando os dados referentes a devolução. Obviamente os argumentos passados a *updateEmail()* também são diferentes.

4.1.2 Main

Nessa função, o Menu com as opções que o usuário deve escolher é implementado. Além disso, os atributos e os métodos da library.cpp são chamados de acordo a necessidade de cada opção do Menu.

- 1. Library livros: É um objeto para a classe Library. Serve para acessar todas os atributos e métodos da classe Library.
- 2. int opcao_menu: É uma variável inteira que guarda o número que o usuário digitou para escolher a opção do Menu. Ela é usada no switch.
- 3. string tituloLivro: Guarda a informação que o usuário digitou sobre o título do livro.
- 4. string autorLivro: Guarda a informação que o usuário digitou sobre o autor do livro.
- 5. string generoLivro: Guarda a informação que o usuário digitou sobre o gênero do livro.
- 6. string email: Guarda a informação que o usuário digitou sobre o email.
- 7. bool livro_existe: Essa variável booleana (true ou false) serve para indicar se o livro já existe na biblioteca ou não.

5 Casos de Uso

O usuário deve escolher uma das 9 opções mostradas na tela ao executar o programa. Um exemplo de uma das opções escolhidas pelo usuário (número 3 - Mostrar livros) pode ser observada a seguir:

Figura 2: Menu impresso na tela para o usuário.

- 1. Inserir livro: Digita título, autor e gênero do livro. Uma mensagem é mostrada na tela de livro inserido, caso dê algum problema ao inserir, mostrará uma mensagem de erro.
- 2. Remover livro: Digita título do livro. Uma mensagem é mostrada na tela de livro removido, caso dê algum problema ao remover, mostrará uma mensagem de erro.
- 3. Mostrar livros: Somente mostra na tela todos os livros contidos na biblioteca.
- 4. Pesquisar livro por título: o usuário digita o título completo ou um trecho dele e então, em caso da pesquisa encontrar algum livro, os dados deste(s) livros serão impressos na tela. Caso haja algum problema em abrir o banco de dados, uma mensagem de erro será exibida. Erros desse tipo poderiam acontecer caso duas threads tentassem usar o mesmo lock ao mesmo tempo, contudo o programa não foi feito para ser multi-threading e tomou-se o cuidado durante a programação para se evitar deadlocks. Caso o programa fosse multi-thread, cuidados adicionais como usado de variáveis de condição ou semáforos deveriam ser utilizados, além da biblioteca lpthreads.
- 5. Realizar empréstimo: O usuário digita o título exato do livro que ele pegar emprestado, assim como seu e-mail. Caso o livro seja encontrado na biblioteca e caso ele esteja disponível pra empréstimo, o status do livro será mudado para Indisponível e o e-mail associado ao livro será mudado para o e-mail do usuário. Caso estes testes falhem, mensagens de erro serão impressas na tela.
- 6. Realizar devolução: Análogo o uso da função anterior, mas nesse caso não se faz necessário digitar o e-mail do usuário. Caso os testes tenham sucesso, o e-mail do usuário será removido do cadastro do livro, assim como o status voltará para Disponível.
- 7. Pesquisar livro por gênero: Nessa pesquisa, o gênero do livro deverá ser digitado exatamente como está nas entradas do banco de dados. Caso a pesquisa encontre um ou mais livros, eles serão impressos na tela.
- 8. Pesquisar livro por email: Análogo ao caso de uso anterior, mas agora a busca é feita por e-mail, sendo útil caso o bibliotecário queira analisar um usuário específico para gerar, por exemplo, um Nada Consta para o usuário.
- 9. Sair: Para finalizar o programa.

6 Conclusão

Este relatório contém informações sobre um possível gerenciamento de uma Biblioteca de Engenharia Eletrônica e de Computação, com o uso da linguagem SQL. Ademais, A junção de C++ e SQLite foi satisfatória e bastante útil para gerenciar a nossa biblioteca. Com isso, conseguimos alcançar o que tínhamos proposto e incrementar funções. Além disso, foi interessante aprender a linguagem SQL e saber interligar C++ com SQLite durante o semestre de 2020.2.

Apesar de termos adicionado bastante funções, percebemos que ainda poderia ser melhorado o programa, por exemplo, utilizar conhecimentos de machine learning para sugerir ao usuário determinados tipos de livros de acordo com o que ele pesquisa ou aluga.