08-03-2024

Jorge Fernandes e Hugo Oliveira

ISTEC

Gestão Biblioteca

Introdução à Programação

Índice

[Introdução 1](#_Toc192369461)

[Descrição do Problema 1](#_Toc192369462)

[Funcionalidades Implementadas 1](#_Toc192369463)

[Implementação 1](#_Toc192369464)

[Ambiente de Desenvolvimento 1](#_Toc192369465)

[Organização do Código 2](#_Toc192369466)

[Principais Desafios 2](#_Toc192369467)

[Testes 3](#_Toc192369468)

[Conclusão 3](#_Toc192369469)

[Código – Fonte 4](#_Toc192369470)

[Função Livro 4](#_Toc192369471)

[Função Salvar Livros 4](#_Toc192369472)

[Função Carregar Livros 5](#_Toc192369473)

[Função Salvar Utilizadores 5](#_Toc192369474)

[Função Carregar Utilizadores 6](#_Toc192369475)

[Função Adicionar Livro 7](#_Toc192369476)

[Função Pesquisar Livros 8](#_Toc192369477)

[Função Registrar Utilizador 9](#_Toc192369478)

[Função Emprestar Livro 10](#_Toc192369479)

[Função Devolver Livro 11](#_Toc192369480)

[Função Relatórios 12](#_Toc192369481)

[Função Menu 13](#_Toc192369482)

[Função Main 13](#_Toc192369483)

## Índice de Imagens

[Figura 1 - Estrutura de dados para armazenar informações do livro 4](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369090)

[Figura 2 - Salvar Livros 4](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369091)

[Figura 3 - Carregar Livros 5](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369092)

[Figura 4 - Salvar Utilizadores 5](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369093)

[Figura 5 - carregarUtilizadores 6](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369094)

[Figura 6 - AdicionarLivros 7](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369095)

[Figura 7 - PesquisarLivros 8](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369096)

[Figura 8 - RegistarUtilizador 9](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369097)

[Figura 9 - EmprestarLivro 10](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369098)

[Figura 10 - DevolverLivro 11](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369099)

[Figura 11 - Relatorios 12](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369100)

[Figura 12 - Menu 13](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369101)

[Figura 13 - Main 13](file:///C:\Users\Utilizador\Downloads\gestao_biblioteca.docx#_Toc192369102)

# Introdução

O objetivo deste projeto é desenvolver um sistema de gestão de biblioteca que permita aos utilizadores realizar operações comuns, como adicionar livros, pesquisar livros, emprestar e devolver livros, e gerar relatórios.

# Descrição do Problema

A gestão manual de bibliotecas pode ser ineficiente e suscetível a erros. Um sistema automatizado melhora a organização, rastreamento de livros e experiência do utilizador.

# Funcionalidades Implementadas

* **Registo de livros**: Permite adicionar informações sobre os livros, como título, autor, ISBN, gênero.
* **Pesquisa de livros**: Possibilita a busca por diferentes critérios, como título, autor ou gênero.
* **Empréstimo e devolução**: Registra empréstimos e devoluções com controle de datas.
* **Gestão de utilizadores**: Registra e gere utilizadores da biblioteca e os seus históricos.
* **Relatórios**: Gera relatórios de livros emprestados, disponíveis e histórico de empréstimos.
* Persistência de dados: Armazena informações em ficheiros CSV.

# Implementação

## Ambiente de Desenvolvimento

* Linguagem de programação utilizada: C
* Ferramentas: Dev-C++ (IDE)
* Ambiente: Windows

# Organização do Código

O código foi dividido em funções principais para garantir modularidade e organização:

* **salvarLivros()**: Salva a lista de livros no arquivo *livros.csv*.
* **carregarLivros()**: Carrega a lista de livros a partir do arquivo *livros.csv*.
* **salvarUtilizadores()**: Salva a lista de utilizadores no arquivo *utilizadores.csv*.
* **carregarUtilizadores()**: Carrega a lista de utilizadores a partir do arquivo *utilizadores.csv*.
* **adicionarLivro()**: Adiciona um novo livro à lista e salva os dados.
* **pesquisarLivro()**: Pesquisa livros.
* **registarUtilizador()**: Regista um novo utilizador.
* **emprestarLivro()**: Empresta um livro a um utilizador.
* **devolverLivro()**: Devolve um livro emprestado.
* **relatorios()**: Gera relatórios de livros emprestados, disponíveis ou históricos de utilizadores.
* **menu()**: Exibe o menu principal e permite escolher a ação a ser realizada.
* **main()**: Carrega dados de livros e utilizadores e exibe o menu principal.

# Principais Desafios

* Implementar a persistência dos dados nos ficheiros CSV.
* Garantir que um livro não seja emprestado mais de uma vez simultaneamente.
* Criar uma interface intuitiva e eficiente para os utilizadores.

# Testes

Foram realizados vários testes para garantir o funcionamento correto da gestão da biblioteca, incluindo:

* Registo e remoção de livros.
* Consulta de livros por título, autor e gênero.
* Registo de empréstimos e devoluções.
* Geração de relatórios.

Todos os testes foram concluídos com sucesso, demonstrando que o sistema é capaz de lidar corretamente com todas as funcionalidades implementas para o trabalho.

# Conclusão

O projeto de Sistema de Gestão de Biblioteca foi concluído com sucesso, permitindo uma gestão organizada e eficiente de livros e utilizadores. Ao longo do desenvolvimento, adquirimos conhecimento em manipulação de ficheiros CSV, organização de estrutura de software e boas práticas de programação.

# Código – Fonte

## Função Livro

Objetivo: Armazenar informações sobre um livro, incluindo o título, autor, ISBN, género, se está emprestado, o ID do utilizador que requisitou e a data.

Como funciona: A estrutura Livro em C++ agrupa esses dados relacionados, facilitando o gerenciamento de livros em um sistema, como, por exemplo, em uma biblioteca.

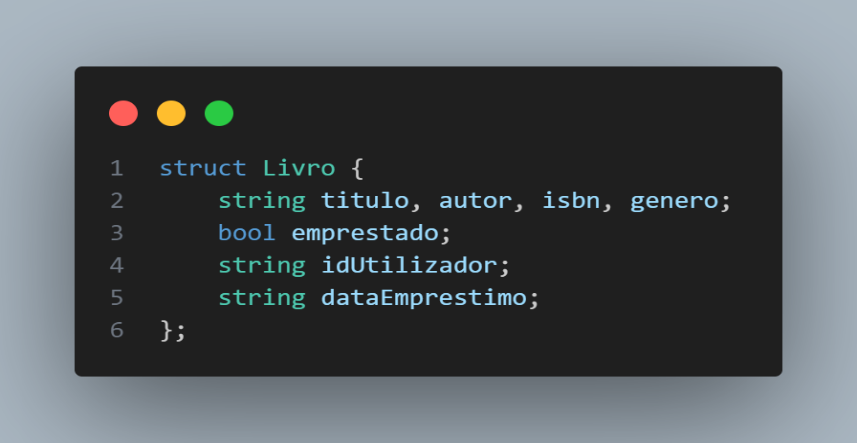


Figura 1 - Estrutura de dados para armazenar informações do livro

## Função Salvar Livros

Objetivo: Salvar os dados dos livros no arquivo livros.csv.

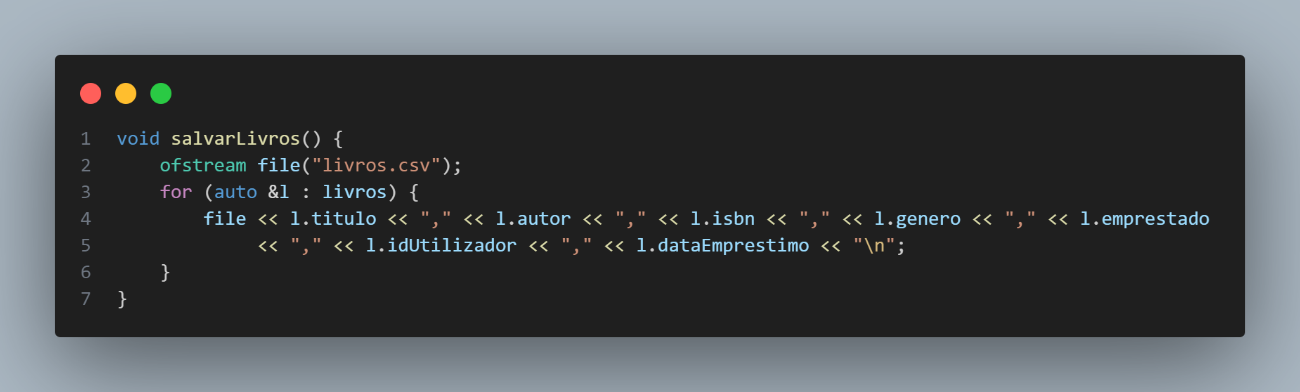
Como funciona: Para cada livro, a função escreve no arquivo os dados do livro (título, autor, ISBN, gênero, se está emprestado, ID do utilizador e data de empréstimo) separados por vírgulas. No final, os dados de todos os livros são gravados no arquivo.

Figura 2 - Salvar Livros

Função Carregar Livros

Objetivo: Carregar os dados dos livros a partir do arquivo livros.csv.

Figura 3 - Carregar Livros

Como funciona: A função lê o arquivo linha por linha. Para cada linha, ela divide as informações usando a vírgula como delimitador e atribui esses dados às variáveis correspondentes de um novo objeto Livro. Esse livro é então adicionado ao vetor livros.

## Função Salvar Utilizadores

Objetivo: Salvar os dados dos utilizadores no arquivo utilizadores.csv.

Como funciona: Funciona quase como a função salvarLivros(), essa função percorre a lista de utilizadores e escreve o nome, ID e o histórico de empréstimos de cada utilizador no arquivo, separando os itens com vírgulas.

Figura 4 - Salvar Utilizadores

## Função Carregar Utilizadores

Objetivo: Carregar os dados dos utilizadores a partir do arquivo utilizadores.csv.

Como funciona: A função lê o arquivo linha por linha, dividindo as informações usando vírgulas. Para cada linha, a função cria um objeto Utilizador e preenche seu nome, ID e histórico de empréstimos com os dados do arquivo. Esse utilizador é adicionado ao vetor utilizadores.

Figura 5 - carregarUtilizadores

## Função Adicionar Livro

Objetivo: Adicionar um novo livro à biblioteca.

Como funciona: A função solicita ao utilizador informações sobre o livro (título, autor, ISBN e gênero). Em seguida, cria um novo objeto Livro com essas informações, marca o livro como não emprestado, e adiciona ao vetor livros. Após isso, a função chama salvarLivros() para salvar os livros no arquivo

Figura 6 - AdicionarLivros

## Função Pesquisar Livros

Objetivo: Pesquisar livros na biblioteca com base num critério.

Como funciona: A função oferece três opções de critérios de pesquisa: título, autor e gênero. O utilizador escolhe um critério e insere um termo de pesquisa. A função então percorre o vetor de livros e verifica se o termo fornecido corresponde a qualquer parte do título, autor ou gênero do livro. Se encontrar um livro correspondente, ele exibe as informações do livro.



Figura 7 - PesquisarLivros

## Função Registrar Utilizador

Objetivo: Registrar um novo utilizador na biblioteca.

Como funciona: A função solicita ao utilizador o nome e ID do utilizador. Em seguida, cria um objeto Utilizador com essas informações e o adiciona ao vetor de utilizadores. A função também chama salvarUtilizadores() para atualizar o arquivo com os novos dados.

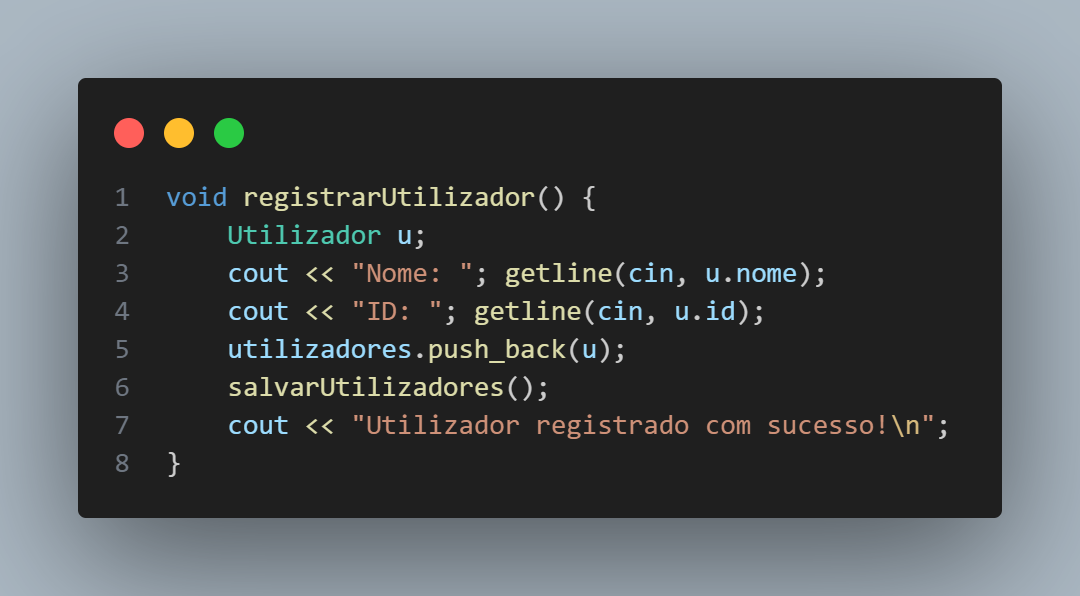


Figura 8 - RegistarUtilizador

## Função Emprestar Livro

Objetivo: Emprestar um livro para um utilizador.

Como funciona: O utilizador fornece o ISBN do livro e o ID do utilizador. A função verifica se o livro com o ISBN fornecido existe e se não está emprestado. Se o livro estiver disponível, ele marca o livro como emprestado, armazena o ID do utilizador e a data do empréstimo. Além disso, o histórico de empréstimos do utilizador é atualizado com a informação do livro emprestado. Por fim, a função chama salvarLivros() e salvarUtilizadores() para atualizar os dados nos arquivos.



Figura 9 - EmprestarLivro

## Função Devolver Livro

Objetivo: Devolver um livro que foi emprestado.

Como funciona: O utilizador fornece o ISBN do livro a ser devolvido. A função encontra o livro correspondente e verifica se ele está emprestado. Se o livro estiver emprestado, ele é marcado como disponível novamente (não emprestado), o histórico de devoluções do utilizador é atualizado, e a função chama salvarLivros() e salvarUtilizadores() para gravar os dados atualizados nos arquivos.



Figura 10 - DevolverLivro

## Função Relatórios

Objetivo: Gerar relatórios sobre o estado da biblioteca.

Livros Emprestados: Exibe todos os livros que estão emprestados, juntamente com o ID do utilizador que os pegou.

Livros Disponíveis: Exibe todos os livros que não estão emprestados.

Histórico por Utilizador: Solicita o ID do utilizador e exibe o histórico de empréstimos e devoluções desse utilizador.

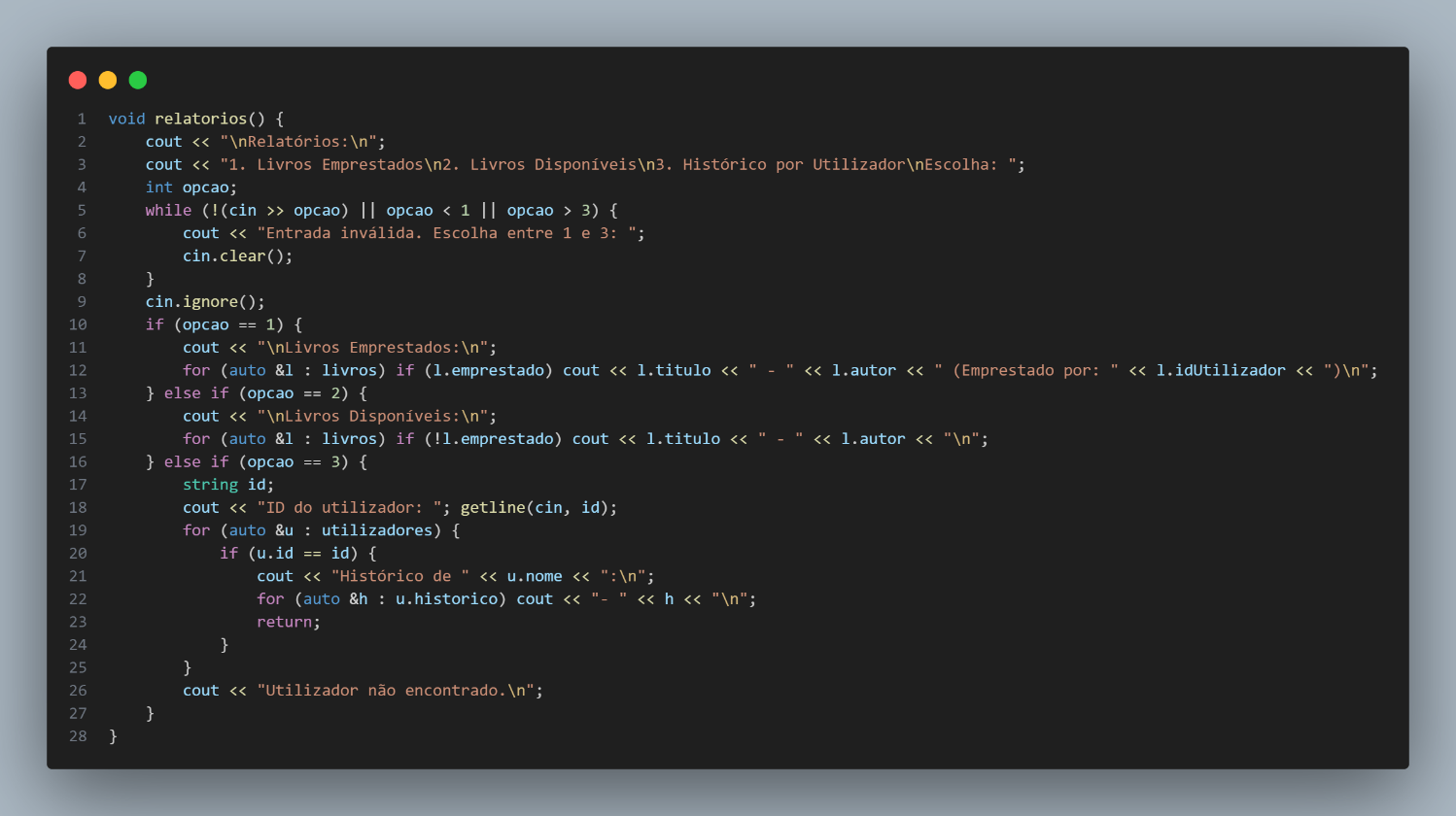
Dependendo da opção escolhida, a função exibe as informações correspondentes.

Figura 11 - Relatorios

## Função Menu

Objetivo: Exibir o menu principal do sistema e permitir que o utilizador escolha a ação desejada.

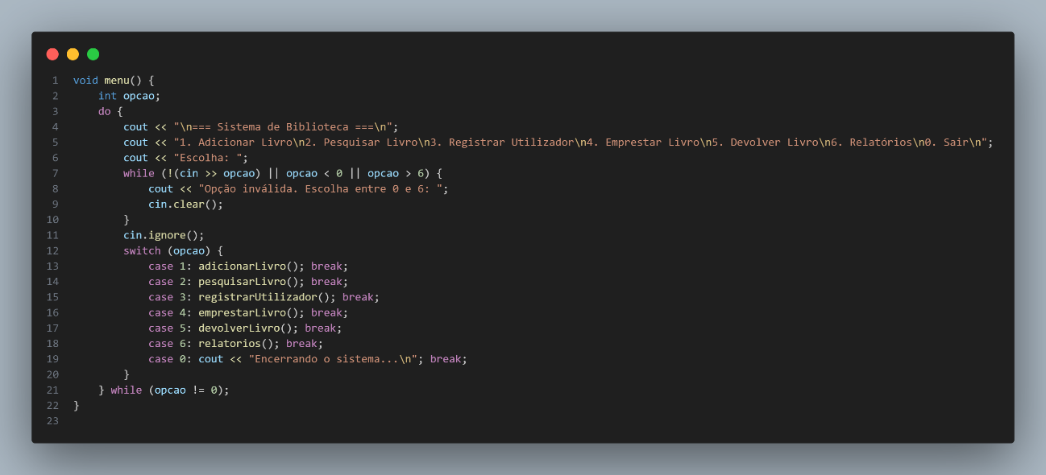
Como funciona: A função apresenta um menu com várias opções, como adicionar livro, pesquisar livro, registrar utilizador, emprestar livro, devolver livro e gerar relatórios. O programa continuará a executar o menu até que o utilizador escolha a opção "Sair" (opção 0). A função chama as funções correspondentes para executar a ação selecionada.

Figura 12 - Menu

## 

## Função Main

Objetivo: Inicializar o sistema e chamar as funções apropriadas.

Como funciona: A função principal carrega os dados dos livros e utilizadores através das funções carregarLivros() e carregarUtilizadores(), e em seguida chama a função menu() para iniciar a interação com o utilizador.

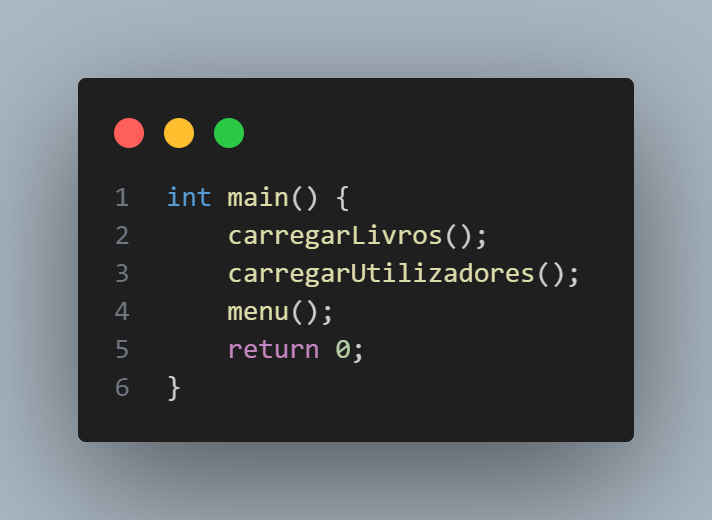


Figura 13 - Main