

# Especificação do Projeto 1a: Gerenciador de Biblioteca

versão 1.0

## 1 Introdução

Este projeto visa criar um sistema de gerenciamento de uma biblioteca utilizando uma **árvore binária de busca** em **linguagem C**. A árvore será estruturada de forma a organizar os livros pelo **código do livro** e permitir a busca por **gênero**. Além disso, o sistema permitirá carregar um conjunto de dados de um arquivo CSV, bem como a inserção de novos livros na árvore.

## 2 Objetivos

- Organizar os livros de uma biblioteca utilizando uma árvore binária de busca, na qual os livros são organizados pelo código único de cada obra.
- Permitir a busca de livros por gênero, exibindo todos os livros daquele gênero específico.
- Carregar os dados de um arquivo CSV, onde cada linha contém informações sobre um livro.
- Permitir a inserção de novos livros na árvore, caso o usuário deseje adicionar novos itens ao sistema.

## 3 Estrutura de Dados

Para armazenar as informações dos livros, será utilizada uma árvore binária de busca. Cada nó da árvore conterá os dados de um livro e terá as seguintes características:

- **Código (inteiro)**: Identificador único do livro.
- **Título (string)**: Nome do livro.
- **Autor (string)**: Autor do livro.

- **Gênero (string):** Gênero do livro.
- **Ano (inteiro):** Ano de publicação do livro.
- **Editora (string):** Nome da editora responsável pela publicação.
- **Número de Páginas (inteiro):** Total de páginas do livro.

### 3.1 Árvore Binária de Busca

A árvore binária de busca será organizada utilizando o código do livro como chave. A inserção de novos livros na árvore seguirá as regras padrão de uma árvore binária de busca, ou seja:

- Se o código do livro a ser inserido for menor que o código do nó atual, o novo livro será inserido à esquerda.
- Se o código do livro a ser inserido for maior, o novo livro será inserido à direita.
- Se o código já existir, o livro não será inserido, evitando duplicação (esse caso deverá ser tratado).

## 4 Funções a serem Implementadas

O projeto será estruturado em funções que desempenham papéis específicos. A seguir, estão listadas as funções principais e suas finalidades.

### 4.1 Função para Criar a Árvore

Esta função será responsável por inicializar a árvore vazia. Quando a árvore for inicializada, ela não conterá nenhum nó.

```
No* inicializar_arvore();
```

### 4.2 Função para Inserir um Novo Livro

A função de inserção permitirá adicionar um livro à árvore. Ela receberá os dados do livro e o código do livro será utilizado para determinar a posição na árvore.

```
void inserir_livro(No** raiz, Livro livro);
```

### 4.3 Função para Buscar por Gênero

A função de busca receberá um gênero e retornará todos os livros que pertencem a esse gênero. A árvore será percorrida e os livros com o gênero correspondente serão exibidos.

```
void buscar_por_genero(No* raiz, char genero[]);
```

#### 4.4 Função para Carregar Livros do Arquivo CSV

Esta função lerá os dados de um arquivo CSV contendo os livros. Para cada linha do arquivo, um novo livro será inserido na árvore. O formato esperado do arquivo é o seguinte:

```
código,título,autor,gênero,ano,editora,número_de_páginas
```

```
No* carregar_livros(char* nome_arquivo, No* raiz);
```

#### 4.5 Função para Exibir a Árvore

Esta função exibirá os livros contidos na árvore.

```
void exibir_arvore(No* raiz);
```

#### 4.6 Função para Liberar a Memória da Árvore

Esta função será responsável por liberar a memória alocada para a árvore quando o programa terminar. Ela fará um percurso pós-ordem para garantir que todos os nós sejam desalocados corretamente.

```
void liberar_arvore(No* raiz);
```

### 5 Fluxo de Operação

1. O projeto deverá ser modularizado.
2. O programa iniciará com a leitura dos dados de livros a partir de um arquivo CSV.
3. A árvore binária será montada com os livros lidos, sendo que os livros serão inseridos na árvore de acordo com seu código.
4. O usuário poderá consultar livros por gênero. A busca será feita pela árvore e todos os livros do gênero solicitado serão exibidos.
5. O usuário poderá também adicionar novos livros à árvore. Para isso, será fornecido um menu que permite a inserção manual de livros.
6. O sistema também permitirá que o usuário visualize os livros presentes na árvore.
7. Ao finalizar, o sistema liberará a memória alocada para a árvore.
8. Opcionalmente é possível implementar outras funcionalidades ao projeto.

## 6 Regras a seguir

1. Este projeto vale 5,0 pontos.
2. O projeto poderá ser implementado em dupla.
3. Os nomes dos participantes deverão ser enviados via tarefa a ser informada.
4. Idem caso o projeto for implementado por apenas um aluno.
5. A data limite para envio do projeto será 30/11/24.
6. A forma de envio do projeto será via tarefa a ser informada.
7. O sorteio dos dias de apresentação será realizado no dia 01/12/24 e seu resultado divulgado no Sigaa.
8. Apenas um dos participantes ficará responsável por enviar o projeto compactado.
9. Este documento está sujeito a alterações e qualquer atualização será informada pelo Sigaa.