



INSTITUTO FEDERAL DO NORTE DE MINAS GERAIS  
CAMPUS MONTES CLAROS -MG  
CIÊNCIA DA COMPUTAÇÃO



JOÃO KENNEDY SOUZA SOARES  
HEULLER RAMOS

## **RELATÓRIO PRÁTICA DE LABORATÓRIO 4**

Relatório proposto pelo Professor Wagner Ferreira de Barros como parte das exigências de avaliação da disciplina de Organização e Sistemas de Arquivos (OSA) do Curso de Bacharelado em Ciência da Computação do Instituto Federal do Norte de Minas Gerais, Campus Montes Claros.

MONTES CLAROS - MG

2024

# Relatório Sobre o Gerenciamento de Livros

## 1. Objetivo

O objetivo deste trabalho foi a implementação de um algoritmo para o gerenciamento de registros de livros, utilizando conceitos de manipulação de arquivos binários e indexação. O foco principal foi armazenar registros de tamanho variável com descritores de tamanho, garantindo acesso rápido por meio de índices organizados em uma árvore binária. A proposta incluiu a leitura inicial de dados de um arquivo CSV, a criação de um arquivo binário contendo os registros, a construção de um arquivo de índices para facilitar buscas e atualizações, e a implementação de funcionalidades para inserção, atualização e pesquisa de dados.

## 2. Estrutura do Projeto

### 2.1. Classes

O projeto foi implementado utilizando as seguintes classes:

**Buffer:** Responsável por manipular os registros, realizando operações de escrita, leitura e buscas. A classe Buffer gerencia a interação com os arquivos binários e de índices, além de utilizar uma árvore binária para armazenar e buscar os índices de forma eficiente.

**Index:** Responsável por armazenar os índices dos registros. Cada índice contém o ID do livro e o endereço do registro no arquivo binário. A classe Index facilita a criação e manipulação dos índices.

**Registro:** Responsável por ler e armazenar os registros nos arquivos. A classe Registro contém métodos para empacotar e desempacotar os dados dos registros, além de carregar os registros a partir de um arquivo CSV.

Ademais, cada classe foi dividida em arquivos .h para armazenar a declaração das funções e atributos, e .cpp para armazenar a implementação. Além disso, foi criada a estrutura básica do Makefile para gerenciar a execução do projeto, facilitando a compilação e limpeza dos arquivos gerados, como também a estrutura fornecida ao projeto de Árvore.h para manipulação do ID e endereço do arquivo Index.

## 2.2. Arquivos

O projeto utiliza os seguintes arquivos para seu funcionamento:

**Arquivo binário principal:** Este arquivo armazena os registros de livros com tamanho variável. Cada registro é delimitado por "|" e inclui um descritor de tamanho que facilita operações de leitura e escrita, permitindo otimização no acesso aos dados.

**Arquivo de índices:** Este arquivo armazena os índices dos registros em formato de tamanho fixo, associando IDs aos endereços físicos dos registros no arquivo binário. Esses índices são organizados em uma árvore binária para possibilitar buscas rápidas e eficientes.

## 2.3. Pesquisa

Durante a criação do arquivo de índices, cada índice foi armazenado em uma estrutura de árvore binária, ordenados pelos IDs dos registros. A pesquisa é feita através de uma função que recebe um ID e realiza a busca na árvore binária. Caso o ID exista, a função verifica a localização daquele ID no arquivo principal e os dados do registro são retornados. Os resultados da busca são mostrados no terminal, permitindo ao usuário visualizar as informações dos livros encontrados.

## 2.4. Inserção e Atualização

Para a inserção de novos registros de livros, foram inseridos 3 novos registros aleatórios no final do arquivo principal. No mesmo momento, também foi realizada a atualização no arquivo de índices e na árvore binária. A inserção de novos registros é feita de forma eficiente, garantindo que os novos dados sejam corretamente indexados e possam ser buscados posteriormente.

## 3. Metodologia

A leitura inicial dos dados foi realizada a partir de um arquivo CSV, utilizando a classe Registro para transformar os dados em um vetor de registros estruturados. A manipulação desses registros foi gerida pela classe Buffer, onde foram implementadas operações de escrita no arquivo binário e criação do arquivo de índices. A árvore binária foi utilizada para armazenar e organizar os índices, permitindo buscas rápidas e precisas. Além disso, as operações foram projetadas para lidar com algumas aspas, que ocasionaram inconsistências nos delimitadores do CSV na leitura, e com isso foi

implementada uma função que conta quantas aspas e se caso elas forem pares significa que a próxima coisa a ser lida é o “;” e se caso forem ímpares deve procurar a próxima aspa até ser par o resultado dela e assim pegar o delimitador do CSV, essas funções são respectivamente chamadas countA e read.

## 4. Testes Realizados

Nos testes realizados, foram utilizados blocos try/catch para tratar exceções, principalmente durante a abertura e escrita nos arquivos para verificação de erros. Foram submetidas 3 buscas de livros existentes no arquivo principal e no arquivo de índice, retornando no terminal os resultados da busca. Além disso, foram inseridos 3 novos registros, que foram adicionados aos arquivos conforme descrito anteriormente.

### 4.1. Testes de Busca

Foram realizados testes de busca utilizando IDs previamente conhecidos. Cada ID foi consultado na árvore binária para localizar o índice correspondente no arquivo de índices. Uma vez encontrado, o endereço foi utilizado para acessar o registro no arquivo binário. As informações retornadas foram exibidas no terminal, comprovando a funcionalidade do sistema como mostrado abaixo:

```
buffer.buscarRegistro(105);  
buffer.buscarRegistro(203);  
buffer.buscarRegistro(102005);
```

Foram encontrados os seguintes dados sobre os IDs buscados.

```
+-----+  
Registros buscados e encontrados com sucesso e mostrados abaixo!  
+-----+  
  
Índice encontrado: ID = 105, Endereço = 2293991  
=====  
ID: 105  
Title: Dark Fields of the Republic: Poems 1991-1995  
Authors: Rich, Adrienne Cecile  
Publish Year: 1995  
Category: Poetry , American , General  
=====  
Índice encontrado: ID = 203, Endereço = 10905907  
=====  
ID: 203  
Title: Vision: A Personal Call to Create a New World  
Authors: Carey, Ken  
Publish Year: 1992  
Category: Body, Mind & Spirit , General  
=====  
Índice encontrado: ID = 102005, Endereço = 11551932  
=====  
ID: 102005  
Title: You've Got Mad Libs  
Authors: Price, Roger and Stern, Leonard  
Publish Year: 2004  
Category: Juvenile Nonfiction , Games & Activities , General  
=====
```

## 4.2. Testes de Inserção

Três novos registros foram adicionados ao final do arquivo binário principal. Após a inserção, os IDs desses registros foram buscados para verificar se estavam corretamente indexados e armazenados, como mostrados abaixo:

```
vector<Registro> novosRegistros = {  
    Registro(123457, "Zen and the Art of Guitar", "Harper, James", 2010, "Music , Instruction & Study , Techniques"),  
    Registro(123458, "Zero Hour: A Novel", "Smith, Caroline", 2018, "Fiction , Thrillers , Action & Adventure"),  
    Registro(123459, "Zodiac Signs: An Illustrated Guide", "Moore, Abigail", 2021, "Self-Help , Astrology , Illustrated")  
};
```

Após a inserção foram buscados os IDs com objetivo de verificar se realmente foram inseridos como mostra abaixo:

```
+-----+  
Registros novos foram adicionados com sucesso e mostrados abaixo!  
+-----+  
  
Índice encontrado: ID = 123457, Endereço = 11568367  
=====  
ID: 123457  
Title: Zen and the Art of Guitar  
Authors: Harper, James  
Publish Year: 2010  
Category: Music , Instruction & Study , Techniques  
=====  
Índice encontrado: ID = 123458, Endereço = 11568464  
=====  
ID: 123458  
Title: Zero Hour: A Novel  
Authors: Smith, Caroline  
Publish Year: 2018  
Category: Fiction , Thrillers , Action & Adventure  
=====  
Índice encontrado: ID = 123459, Endereço = 11568556  
=====  
ID: 123459  
Title: Zodiac Signs: An Illustrated Guide  
Authors: Moore, Abigail  
Publish Year: 2021  
Category: Self-Help , Astrology , Illustrated  
=====  
Todas operações concluída com sucesso!
```

E após isso é mostrado uma mensagem onde todas as operações foram realizadas com sucesso indicando que tudo foi realmente feito.

## 5. Resultado

O projeto cumpriu todas as exigências fornecidas na descrição. Foram utilizadas, em sua grande maioria, as estruturas recomendadas pelo professor. Algumas alterações foram realizadas para otimizar processos e adequar o trabalho. Os testes realizados demonstraram que o sistema é capaz de ler registros de um arquivo CSV, armazená-los em um arquivo binário, criar um arquivo de índices e realizar buscas eficientes utilizando uma árvore binária.

## **6. Conclusão**

Houve uma dificuldade considerável para entender a estrutura do projeto. Demorou mais para entender e adaptar a implementação do que entender o que devia ser feito. Algumas funções ficaram sobrecarregadas devido à complexidade de implementar de forma diferente garantindo a eficiência. Contudo, o projeto foi realizado com sucesso e o conhecimento adquirido foi de grande importância. A implementação de estruturas de classes e a manipulação de arquivos binários proporcionaram um aprendizado importante sobre a manipulação de arquivos de forma mais rápida usando o conteúdo visto em sala.