

**CESED – CENTRO DE ENSINO SUPERIOR E DESENVOLVIMENTO
CENTRO UNIVERSITÁRIO UNIFACISA**

**APLICAR ALGORITMOS DE APOIO À PESQUISA OPERACIONAL
PROF. JONHNANTHAN OLIVEIRA**

**DEFININDO A ESTRUTURA DE DADOS PARA
ADMINISTRAÇÃO DE UMA BIBLIOTECA**

**ALUNOS DO GRUPO
GUSTAVO TOMIO MAGALHÃES KUBO
SÉRGIO MAGNO CASTOR PINHEIRO
THIAGO LIMEIRA DE ALENCAR**

**CAMPINA GRANDE – PB
2024.2**

Objetivo:

Neste projeto, nosso grupo – composto por Gustavo Tomio Magalhaes Kubo, Thiago de Alencar Limeira e Sérgio Magno Castor Pinheiro – escolheu implementar uma **lista encadeada** como estrutura de dados principal para gerenciar os livros de uma biblioteca. Esta decisão foi tomada considerando as características do projeto, no qual a adição e remoção de livros são operações frequentes e não há um limite previamente definido para a quantidade de livros a ser armazenada.

Diferente de uma lista sequencial (ou estática), a lista encadeada nos permite adicionar e remover elementos com mais flexibilidade e eficiência, sem a necessidade de reconfigurar ou deslocar elementos na memória. Essa abordagem evita problemas de desperdício de espaço e fragmentação de memória, características comuns em listas sequenciais que precisam ser redimensionadas.

Com orientação do professor Jonhnanthan Oliveira na disciplina **Aplicar Algoritmos de Apoio à Pesquisa Operacional**, decidimos por essa estrutura de dados por sua flexibilidade e pela eficiência em operações de gerenciamento de memória, essenciais para atender os requisitos do projeto de forma prática e otimizada.

Explicando a estrutura de dados escolhida

Vantagens e Desvantagens:

1. Lista Encadeada:

- **Vantagens:**

- Muito boa para situações onde é preciso adicionar ou remover elementos com frequência, pois isso é rápido (tempo constante) apenas ajustando os ponteiros.
- Não desperdiça espaço na memória, pois ocupa só o espaço necessário para os elementos que são adicionados.

- **Desvantagens:**

- Usa mais memória, pois cada elemento precisa de um ponteiro extra que aponta para o próximo (ou anterior, se for uma lista duplamente encadeada).
- Procurar um elemento específico pode ser mais lento, já que precisamos percorrer a lista inteira.

2. Lista Sequencial (Estática):

- **Vantagens:**

- Acesso rápido aos elementos, pois eles estão todos em sequência na memória, o que também facilita buscas diretas se soubermos a posição.
- Muito eficiente para elementos de tamanho fixo, pois não exige ponteiros adicionais.

- **Desvantagens:**

- Inserir e remover elementos no início ou no meio é complicado, pois é preciso deslocar os elementos para abrir ou fechar espaço, o que toma mais tempo.
- O tamanho precisa ser decidido logo no início, ou ajustado depois, o que limita a flexibilidade e pode causar fragmentação de memória.

Justificativa da Decisão:

Como o projeto envolve adicionar livros com frequência e não temos um limite fixo de quantos livros serão armazenados, a **lista encadeada** é a mais indicada. Ela é mais flexível e permite adicionar e remover livros sem precisar ajustar a memória toda vez, o que a lista sequencial exigiria.

Dificuldades ao usar uma Lista Sequencial para o projeto:

Se usarmos uma lista sequencial (ou seja, uma lista com tamanho fixo ou que precisa ser ajustado), podem surgir algumas dificuldades:

1. Inserção e Remoção de Elementos:

- Para inserir um livro em uma posição específica ou remover um do meio, é preciso mover todos os elementos que estão depois dessa posição. Isso toma mais tempo, especialmente se a lista for grande.

- Adicionar um livro no começo da lista é ainda mais trabalhoso, pois precisamos mover todos os elementos um espaço para frente.

2. Tamanho Fixo:

- A lista sequencial precisa de um tamanho fixo. Isso significa que, se ela encher, teremos que programar uma forma de expandir a lista, o que geralmente envolve criar uma lista maior e copiar tudo para a nova.

- Se a lista for muito grande e não preenchermos tudo, teremos espaço desperdiçado.

3. Complexidade de Memória:

- Quando aumentamos o tamanho da lista, isso pode criar fragmentação de memória, especialmente se o uso de inserções e remoções for constante.

4. Ordenação:

- Organizar uma lista sequencial exige mover elementos, enquanto na lista encadeada podemos apenas reorganizar os ponteiros dos nós.

5. Acesso a Elementos:

- Embora a lista sequencial seja rápida para acessar elementos por posição, se precisarmos procurar um elemento específico, ainda precisamos percorrer toda a lista. A lista encadeada pode ser implementada para ter um acesso mais prático em operações de ordenação e busca.

Conclusão:

- Apesar de a lista sequencial ser simples e rápida para algumas operações, suas limitações de flexibilidade, o custo de inserções e remoções e a dificuldade em ajustar o tamanho fazem da lista encadeada a escolha mais prática para o projeto da biblioteca. Como vamos adicionar e remover livros com frequência, a lista encadeada facilita bastante o desenvolvimento, pois não exige ajustes constantes de espaço na memória.