

# Implementação 1

Bruno Braga Guimarães Alves

## Dados do Aluno

- Nome: Bruno Braga Guimarães Alves
- Matrícula: 767681
- Curso: Ciência da Computação

## Descrição do Projeto

Este projeto busca implementar o que foi solicitado no exercício "Implementação 1".

## Lista Encadeada

### Estrutura do Código

- **Classe No:**
  - Representa um nó da lista encadeada.
  - Cada nó possui dois atributos:
    - \* **elemento:** Armazena o valor do nó.
    - \* **proximo:** Um ponteiro para o próximo nó na lista.
- **Classe Lista:**
  - Gerencia a lista encadeada como um todo.
  - Atributo:
    - \* **raiz:** Um ponteiro para o primeiro nó (cabeça) da lista.
  - Métodos:
    - \* Possui 3 métodos para Inserção:
      - **Inserir:** Insere o elemento em ordem crescente.
      - **InserirInicio:** Insere o elemento no início.
      - **InserirFim:** Percorre a lista para inserir o elemento no fim.
    - \* Possui 3 métodos para Remoção:
      - **Remove:** Remove o nó na posição especificada.
      - **RemoveInicio:** Muda o elemento da raiz para o nó "próximo", removendo e retornando o valor do primeiro nó.
      - **RemoveFim:** Percorre a lista até remover e retornar o valor do último nó.
    - \* **Mostrar:** Retorna os valores contidos na lista.
    - \* **Buscar:** Procura e retorna o valor desejado.

# Pilha

## Estrutura do Código

- **Classe No:**
  - Representa um nó da pilha.
  - Atributos:
    - \* **elemento:** Armazena o valor do nó.
    - \* **proximo:** Um ponteiro para o próximo nó na pilha.
- **Classe Pilha:**
  - Atributo:
    - \* **raiz:** Um ponteiro para o primeiro nó da pilha.
  - Métodos:
    - \* **Inserir:** Insere um novo elemento no topo da pilha.
    - \* **Remover:** Remove o elemento do topo da pilha.
    - \* Outros métodos podem existir para operações adicionais, como visualizar o topo da pilha ou verificar se a pilha está vazia.
    - \* **Mostrar:** Retorna os valores contidos na pilha.
    - \* **Buscar:** Procura e retorna o valor desejado.

# Matriz

## Estrutura do Código

- **Definições e Includes:**
  - **NUM.LIN** e **NUM.COL** definem as dimensões da matriz (3x3).
  - Inclui bibliotecas padrão como **iostream**, **string** e **stdexcept**.
- **Classe Matriz:**
  - Atributo:
    - \* **matriz[NUM.LIN][NUM.COL]:** Um array 2D para armazenar os valores da matriz.
  - Métodos:
    - \* **preencherMatriz:** Preenche a matriz com valores inseridos pelo usuário.
    - \* **exibirMatriz const:** Exibe os valores da matriz no console.
    - \* **identificaElemento:** Identifica a posição de um elemento específico na matriz.
    - \* **RemoveElemento:** Remove o elemento especificado da matriz, substituindo-o por -1.

# Fila

## Estrutura do Código

- A classe **No** representa cada elemento da fila.
  - Atributos:
    - \* **int elemento:** Armazena o valor do nó.
    - \* **No \*proximo:** Ponteiro para o próximo nó na fila.
  - Construtor:
    - \* **No(int elemento):** Inicializa o nó com um valor e define o ponteiro **proximo** como **nullptr**.
- A classe **Fila** representa a estrutura de dados fila, utilizando uma lista encadeada.

- Atributos:
  - \* **No** \***raiz**: Ponteiro para o primeiro nó da fila.
- Métodos:
  - \* **Inserir**: Insere um novo elemento na fila, criando um novo nó.
  - \* **Remover**: Remove o primeiro elemento da fila. Salva o elemento do primeiro nó, avança a raiz para o próximo nó e deleta o nó antigo.
  - \* **Mostrar**: Exibe todos os elementos da fila.
  - \* **Buscar**: Procura e retorna o valor desejado.