

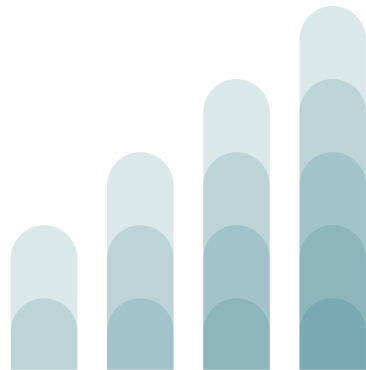
INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus Campos do Jordão

ESTRUTURA DE DADOS

Listas: Tipos e Pesquisa Sequencial

Lista Simples, Lista Encadeada, Lista Duplamente Encadeada, Lista Circular, Pesquisa Sequencial

Professor Mestre Igor de Moraes Sampaio
igor.sampaio@ifsp.edu.br





Lista Simples





O que é uma Lista?

Lista é uma estrutura de dados linear usada para armazenar uma coleção ordenada de elementos (como em um vetor/array), onde:

- Características:
 - Acesso direto por índice.
 - Inserção e remoção no meio exigem deslocamento de elementos.
 - Boa para buscas rápidas se souber o índice.



Array x Vetor x Lista

Estrutura	Tamanho	Armazenamento	Tipo dos Elementos	Acesso	Flexibilidade
Array	Fixo	Contíguo na memória	Mesmo tipo	Direto (índice)	Pouca
Vetor	Fixo (1D)	Contíguo na memória	Mesmo tipo	Direto (índice)	Pouca
Lista	Dinâmico	Pode ser não contíguo	Pode variar	Sequencial	Alta



Tipos de Listas

- Lista Estática

- Tamanho fixo (definido na criação).
- Exemplo clássico: vetores (arrays).

- Lista Dinâmica


- Tamanho variável (pode crescer ou diminuir).
- Implementada com ponteiros em C++ (ex: lista encadeada).

Exemplo completo
disponível no Classroom.
Arrays.cpp


```
int numeros[5] = {1, 2, 3, 4, 5};  
cout << numeros[2]; // Acessa o 3º elemento
```



Exemplo de Lista Linear Estática (Array) em C++



Pesquisa Sequencial





O que é uma Lista?

- O algoritmo mais simples de busca: percorre a lista do início ao fim.

```
para cada elemento na lista:  
    se elemento == buscado:  
        retorna posição  
retorna "não encontrado"
```




Desafio





Desafio

Implemente em menos de 10 minutos o algoritmo de pesquisa sequencial em C++.

Implemente usando o for e depois usando o for each.

```
// For tradicional: controla o índice manualmente
for (int i = 0; i < (int)numeros.size(); i++) {
    // Acessa elemento pelo índice i
    std::cout << numeros[i] << std::endl;
}

// Foreach (range-based for): itera diretamente sobre cada elemento
for (int num : numeros) {
    // 'num' é uma cópia do elemento atual da coleção
    std::cout << num << std::endl;
}
```



Lista Encadeada





Lista Encadeada Simples

Uma Lista Encadeada é uma estrutura de dados dinâmica formada por nós.

- **Estrutura:**

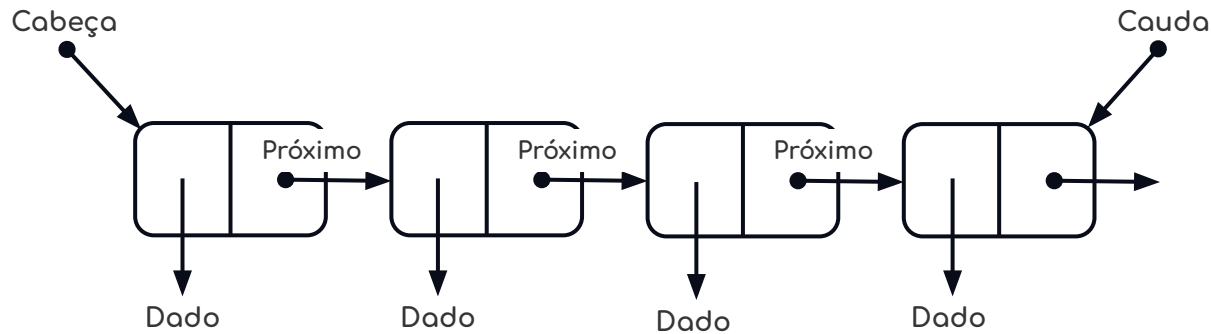
- Cada nó possui:
 - Dado – valor armazenado.
 - Ponteiro – referência para o próximo nó.

- **Vantagens:**

- Crescimento dinâmico
- Inserção/remoção no início em $O(1)$

- **Desvantagens:**

- Acesso sequencial apenas (não há índice)
- Mais uso de memória (guarda ponteiros)



```
struct Node {  
    int data;  
    Node* next;  
};
```

Exemplo disponível no
Classroom.
ListaEncadeada.cpp

Exemplo de Nó de Lista Encadeada Simples em C++



Operações e Complexidades

Operação	Lista Simples (array)	Lista Encadeada
Acesso direto	$O(1)$	$O(n)$
Busca sequencial	$O(n)$	$O(n)$
Inserção no início	$O(n)$	$O(1)$
Inserção no fim	$O(1)$	$O(n)$ (ou $O(1)$ se mantiver ponteiro)
Inserção no meio	$O(n)$	$O(n)$
Remoção no início	$O(n)$	$O(1)$
Remoção no fim	$O(1)^*$	$O(n)$
Remoção no meio	$O(n)$	$O(n)$



Desafio





Desafio

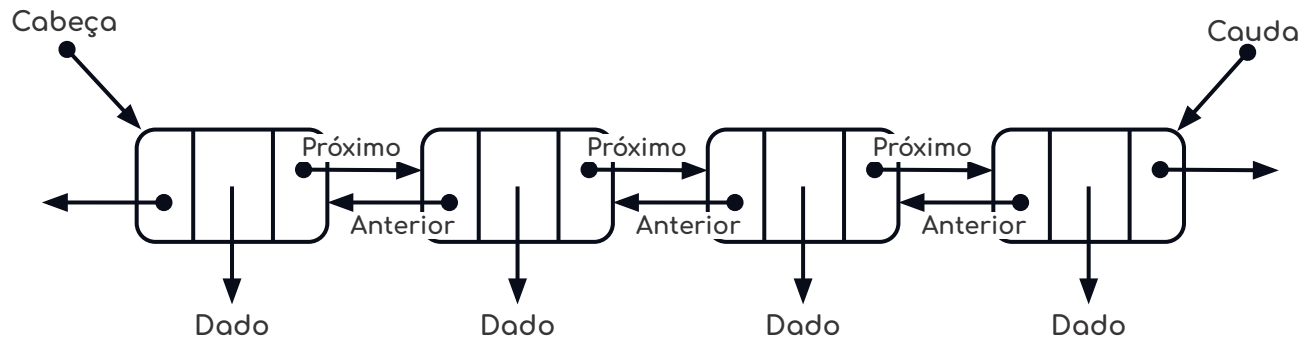
A partir do exemplo desenvolvido em sala faça a operações que faltam:

- Inserção no início
- Inserção no meio
- Remoção no início
- Remoção no fim
- Remoção no meio



Lista Duplamente Encadeada

- **Estrutura:**
 - Cada nó possui:
 - Dado
 - Ponteiro para o próximo
 - Ponteiro para o anterior
- **Vantagens:**
 - Navegação em ambas as direções
 - Remoções/inserções mais flexíveis
- **Desvantagens:**
 - Maior uso de memória (dois ponteiros)
 - Implementação mais complexa



```
struct Node {  
    int data;  
    Node* prev;  
    Node* next;  
};
```

Exemplo de Nó de Lista Duplamente Encadeada em C++



Lista Circular

- **Variantes:**
 - Simplesmente encadeada circular: último aponta para o primeiro
 - Duplamente encadeada circular: primeiro e último se conectam nos dois sentidos
- **Uso comum:**
 - Estrutura de fila circular
 - Simulação de sistemas circulares (ex: jogos, round robin)



Sugestão de Exercícios





Exercício

Usando sua implementação de lista encadeada simples, implemente:

- A lista duplamente encadeada.
- A lista circular encadeada simples.
- A lista circular duplamente encadeada.