

Estrutura de Dados (CCA410)

Aula 01 - TADs, Arrays e Listas Ligadas

Prof. Luciano Rossi

Prof. Leonardo Anjoletto Ferreira

Prof. Flavio Tonidandel

Ciência da Computação
Centro Universitário FEI

2º Semestre de 2025

Tipo de Dado Abstrato (TDA)

Tipo de Dado Abstrato (TDA)

Definição

Definição

- ▶ Modelo matemático ou conceitual que define um conjunto de **dados** e as **operações** que podem ser realizadas sobre esses dados, independentemente de sua implementação concreta.
- ▶ O TDA especifica o que o tipo de dado faz, mas não como ele é implementado em termos de estruturas de dados ou algoritmos.

Tipo de Dado Abstrato (TDA)

Componentes de um TDA

Um TDA é normalmente definido por:

- ▶ **Conjunto de valores:** os elementos que podem fazer parte do tipo (por exemplo, elementos de uma lista, números em uma fila, etc.).
- ▶ **Conjunto de operações:** funções ou procedimentos definidos sobre o tipo (por exemplo, inserir, remover, buscar, esvaziar).
- ▶ **Propriedades ou axiomas:** regras ou restrições que as operações devem obedecer.

Tipo de Dado Abstrato (TDA)

Exemplos clássicos de TDA

- ▶ **Fila (Queue):** operações típicas incluem enqueue (enfileirar), dequeue (desenfileirar), peek (consultar o primeiro).
- ▶ **Pilha (Stack):** operações como push (empilhar), pop (desempilhar), top (consultar o topo).
- ▶ **Lista:** operações para inserir, remover ou acessar elementos.
- ▶ **Dicionário / Tabela de dispersão:** operações para associar chaves a valores.

Tipo de Dado Abstrato (TDA)

Importância

O conceito de TDA permite:

- ▶ **Abstrair** os detalhes de implementação, focando no comportamento esperado.
- ▶ Projetar sistemas **modulares**, com separação clara entre interface e implementação.
- ▶ Facilitar a análise e o raciocínio formal sobre **corretude** e **complexidade**.

Tipo de Dado Abstrato (TDA)

Exemplo: Pilha - Definição

- ▶ **Conjunto de valores:** sequência ordenada de elementos com acesso restrito ao topo.
- ▶ **Operações:**
 - ▶ **criar()** - cria uma pilha vazia.
 - ▶ **empilhar(S, x)** - insere elemento no topo.
 - ▶ **desempilhar(S)** - remove o topo.
 - ▶ **topo(S)** - consulta o topo.
 - ▶ **vazia(S)** - verifica se a pilha está vazia.

Tipo de Dado Abstrato (TDA)

Exemplo: Pilha - Axiomas¹

As operações da pilha devem obedecer:

- ▶ **vazia(criar()) = True**
- ▶ **vazia(empilhar(S, x)) = False**
- ▶ **topo(empilhar(S, x)) = x**
- ▶ **desempilhar(empilhar(S, x)) = S**

Axioma é uma propriedade ou regra formal que descreve como as operações de um TAD devem se comportar, sendo aceita como ¹verdadeira para garantir a consistência do modelo abstrato.

Tipo de Dado Abstrato (TDA)

Exemplo: Pilha - Interface em Pseudocódigo

- ▶ **tipo Pilha**
- ▶ **função criar() → Pilha**
- ▶ **procedimento empilhar(ref S: Pilha, x: Elemento)**
- ▶ **função desempilhar(ref S: Pilha) → Elemento**
- ▶ **função topo(S: Pilha) → Elemento**
- ▶ **função vazia(S: Pilha) → Booleano**

TDA vs Estrutura de Dados

Relação Conceitual

- ▶ Um **Tipo de Dado Abstrato (TDA)** define o **comportamento** esperado e as operações permitidas sobre um conjunto de valores.
- ▶ Uma **estrutura de dados** é a forma **concreta** como os dados são organizados e armazenados na memória.
- ▶ O TDA descreve o **que** pode ser feito; a estrutura de dados define **como** isso é realizado.
- ▶ Um TDA **não é uma estrutura de dados**, mas pode ser **implementado** por diferentes estruturas de dados.

Array (Vetor)

Array (Vetor)

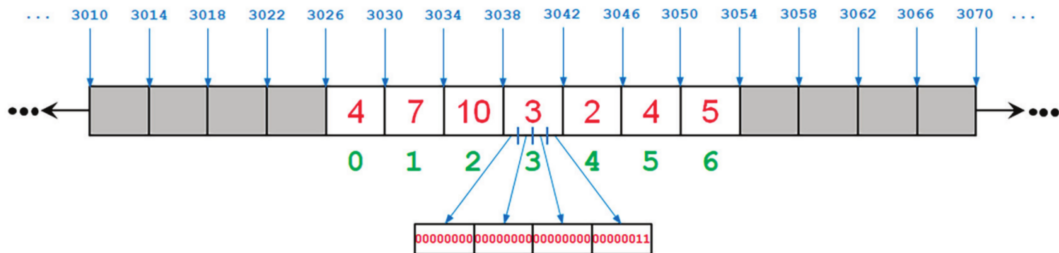
Definição

Definição

- ▶ Um **array** (ou vetor) é uma estrutura de dados concreta que armazena uma coleção de elementos do mesmo tipo em posições contíguas na memória.
- ▶ Permite o **acesso direto** a cada elemento por meio de um índice.
- ▶ O tamanho do array é geralmente fixado no momento da sua criação.

Array (Vetor)

Exemplo



Listas

Lista

Definição

Definição

- ▶ Uma lista é uma estrutura de dados que representa uma sequência de elementos em uma ordem específica;
- ▶ As listas podem ser implementadas de várias maneiras, mas as duas implementações mais comuns:
 - ▶ linked list (dinâmica);
 - ▶ array list (estática).

Lista Dinâmica Encadeada (LDE)

Definição

Definição

- ▶ Uma **lista dinâmica encadeada** é uma estrutura de dados em que os elementos (nós) são armazenados em posições não contíguas na memória.
- ▶ Cada nó contém um **valor** e uma **referência** (ponteiro) para o próximo nó da lista.
- ▶ O tamanho da lista pode variar dinamicamente durante a execução do programa.

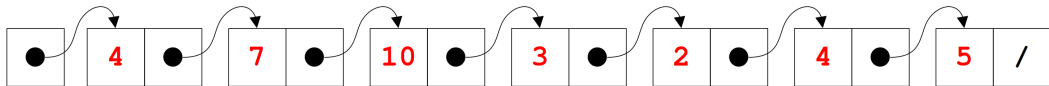
Lista Dinâmica Encadeada (LDE)

Propriedades

- ▶ Não há **limite fixo** de tamanho (cresce e diminui conforme necessário).
- ▶ Inserção e remoção de elementos são feitas ajustando os **ponteiros**.
- ▶ Acesso sequencial: para acessar um elemento, é necessário **percorrer** a lista desde o início.
- ▶ Pode ser usada para implementar TDAs como listas, pilhas e filas.

Lista Dinâmica Encadeada (LDE)

Exemplo



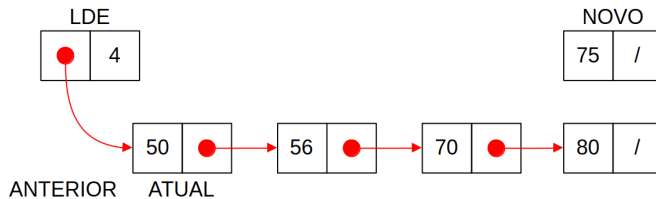
Lista Dinâmica Encadeada (LDE)

Pseudocódigo da interface (inserção ordenada)

- ▶ tipo ListaLigada
- ▶ função criar() → ListaLigada
- ▶ procedimento inserir_ordenado(ref L: ListaLigada, elemento: Elemento)
- ▶ procedimento remover(ref L: ListaLigada, ref nó: Nó)
- ▶ função buscar(L: ListaLigada, elemento: Elemento) → ref Nó
- ▶ função tamanho(L: ListaLigada) → Inteiro
- ▶ função primeiro(L: ListaLigada) → ref Nó
- ▶ função próximo(ref nó: Nó) → ref Nó

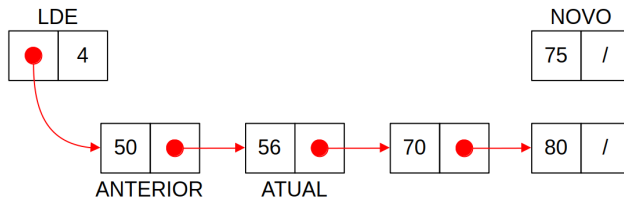
Lista Dinâmica Encadeada (LDE)

Operação de busca



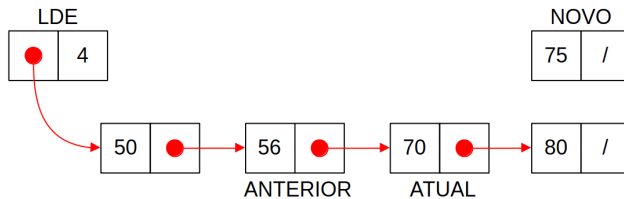
Lista Dinâmica Encadeada (LDE)

Operação de busca



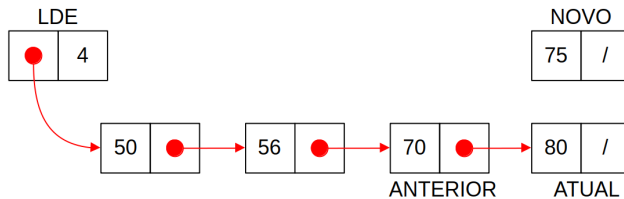
Lista Dinâmica Encadeada (LDE)

Operação de busca



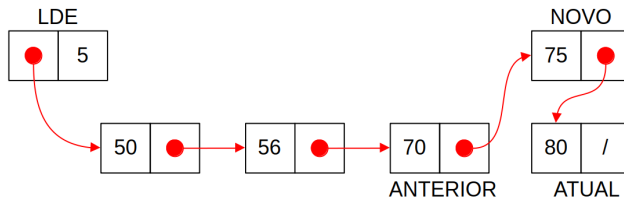
Lista Dinâmica Encadeada (LDE)

Operação de busca



Lista Dinâmica Encadeada (LDE)

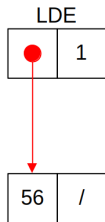
Operação de busca



Lista Dinâmica Encadeada (LDE)

Operação de inserção

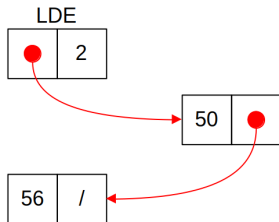
- Inserir no início de uma lista vazia):



Lista Dinâmica Encadeada (LDE)

Operação de inserção

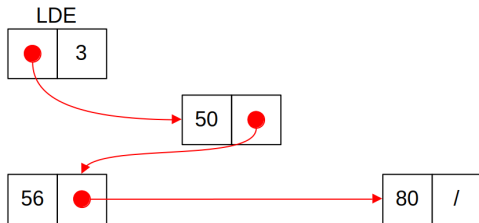
- Inserir no início de uma lista não vazia



Lista Dinâmica Encadeada (LDE)

Operação de inserção

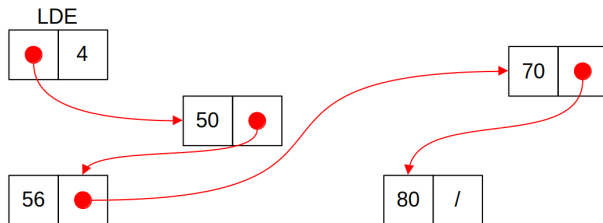
- Inserir no final de uma lista não vazia



Lista Dinâmica Encadeada (LDE)

Operação de inserção

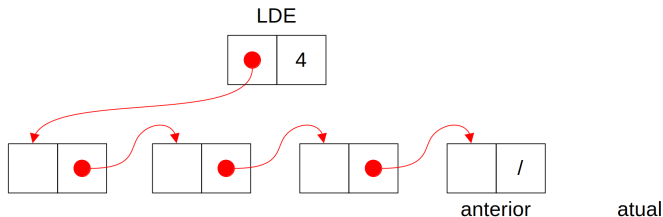
- Inserir no meio de uma lista não vazia



Lista Dinâmica Encadeada (LDE)

Operação de remoção

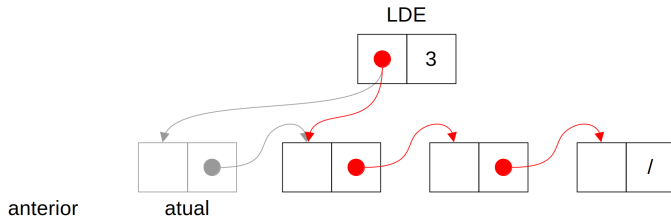
- ▶ Lista vazia ou elemento não encontrado
 - ▶ retornar sem executar nada.



Lista Dinâmica Encadeada (LDE)

Operação de remoção

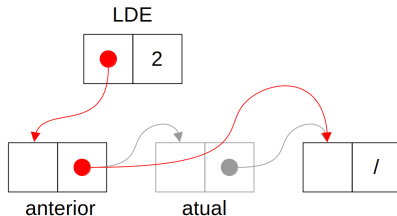
► Remover no início



Lista Dinâmica Encadeada (LDE)

Operação de remoção

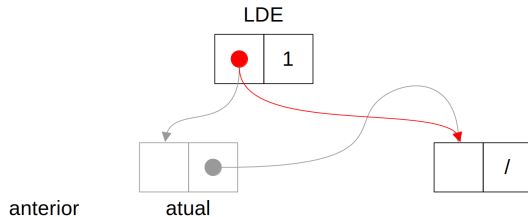
- Remover no meio



Lista Dinâmica Encadeada (LDE)

Operação de remoção

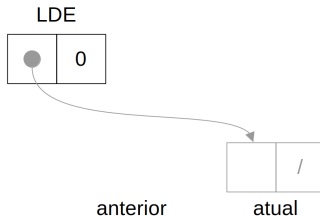
- Remover no início (novamente)



Lista Dinâmica Encadeada (LDE)

Operação de remoção

- Remover elemento único



Lista Dinâmica Encadeada (LDE)

Pseudocódigo: Inserção ordenada

INSERIR(*L*, *x*)

```
1  novo ← criar novo nó com valor x
2  anterior ← NULL
3  atual ← L.primeiro
4  enquanto atual ≠ NULL e atual.valor < x faça
5      anterior ← atual
6      atual ← atual.proximo
7  se anterior == NULL então
8      novo.proximo ← L.primeiro
9      L.primeiro ← novo
10 senão
11     anterior.proximo ← novo
12     novo.proximo ← atual
13 L.qtde ← L.qtde + 1
```

Lista Dinâmica Encadeada (LDE)

Pseudocódigo: Remoção de elemento

REMOVER(*L*, *x*)

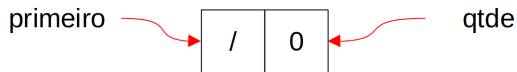
```
1  anterior ← NULL
2  atual ← L.primeiro
3  enquanto atual ≠ NULL e atual.valor ≠ x faça
4      anterior ← atual
5      atual ← atual.proximo
6  se atual == NULL então retorne
7  se anterior == NULL então
8      L.primeiro ← atual.proximo
9  senão
10     anterior.proximo ← atual.proximo
11  libere atual
12  L.qtde ← L.qtde - 1
```

Implementação

Lista Dinâmica Duplamente Encadeada (LDDE)

Lista Dinâmica Duplamente Encadeada (LDDE)

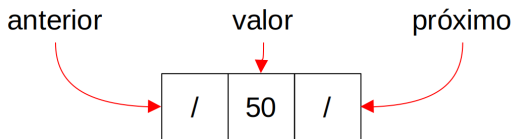
Estrutura



Estrutura típica de uma LDDE.

Lista Dinâmica Duplamente Encadeada (LDDE)

Estrutura

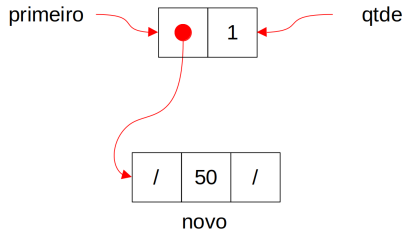


Estrutura típica de uma célula.

Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de inserção

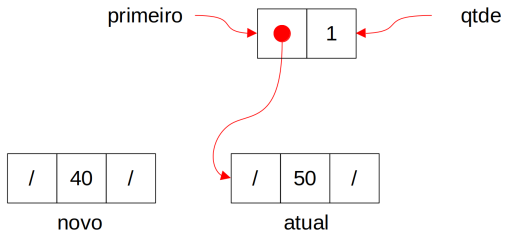
- Inserir em uma lista vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de inserção

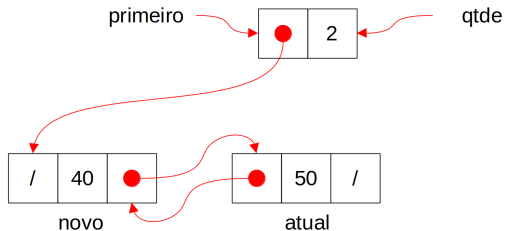
- Inserir no início de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de inserção

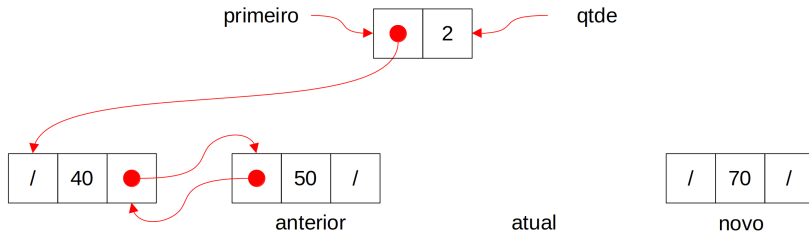
- Inserir no início de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de inserção

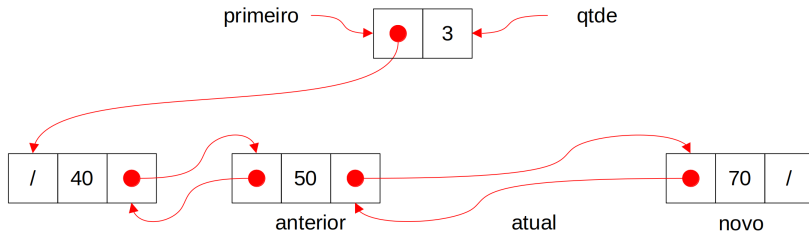
- Inserir no final de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de inserção

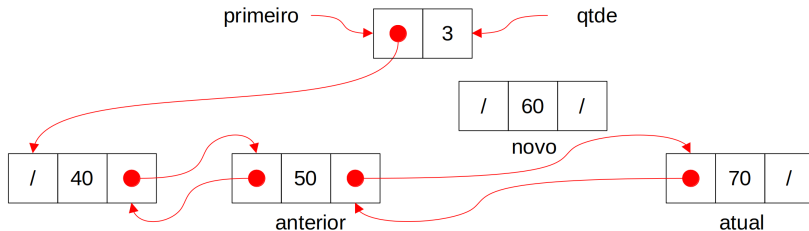
- Inserir no final de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de inserção

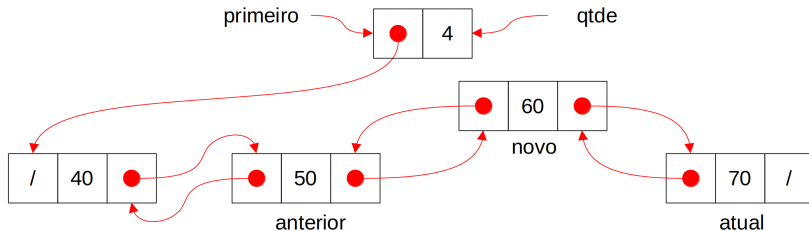
- Inserir no meio de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de inserção

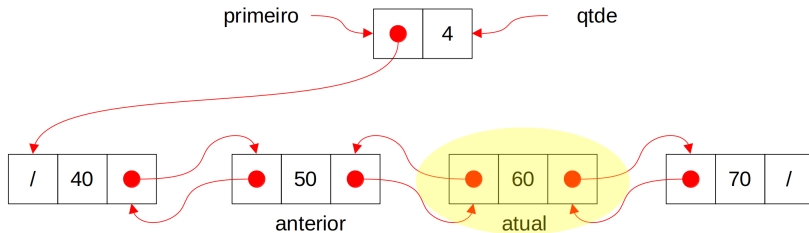
- Inserir no meio de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de remoção

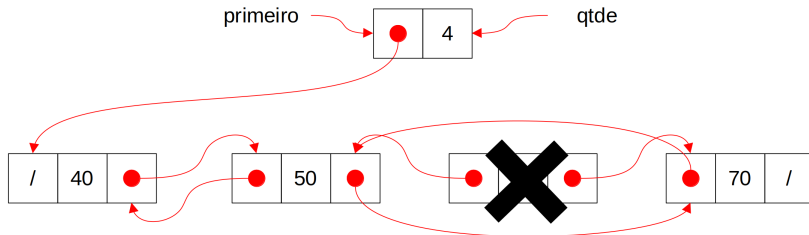
- Buscar por um valor na lista (60):



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de remoção

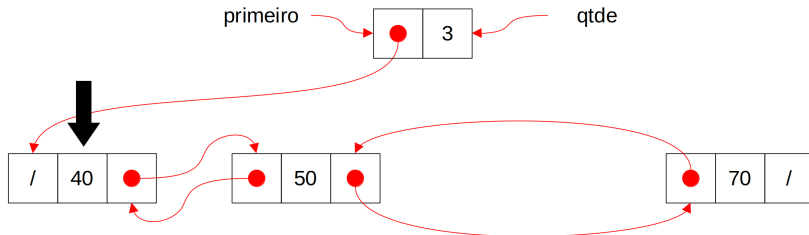
- Remover no meio de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de remoção

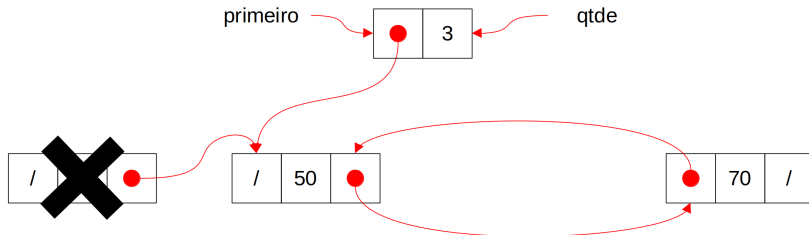
- Remover no início de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de remoção

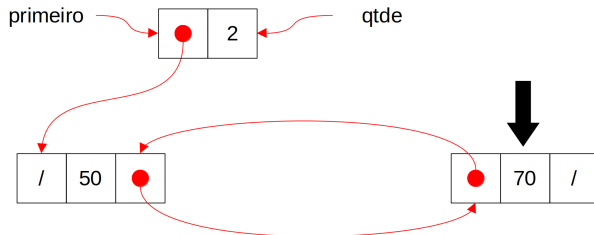
- Remover no início de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de remoção

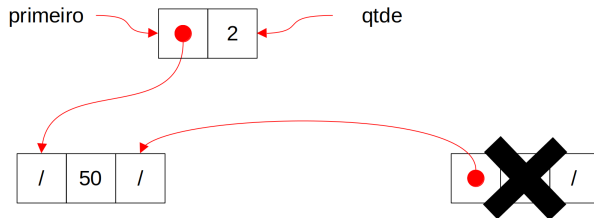
- Remover no final de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de remoção

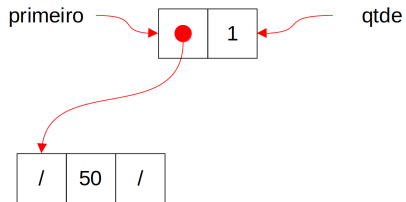
- Remover no final de uma lista não vazia:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de remoção

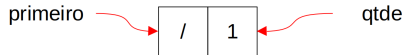
- Remover de uma lista unitária:



Lista Dinâmica Duplamente Encadeada (LDDE)

Operação de remoção

- Remover de uma lista unitária:



Laboratório

Estrutura de Dados (CCA410)

Aula 01 - TADs, Arrays e Listas Ligadas

Prof. Luciano Rossi

Prof. Leonardo Anjoletto Ferreira

Prof. Flavio Tonidandel

Ciência da Computação
Centro Universitário FEI

2º Semestre de 2025