

Dominando Estruturas de Dados 1

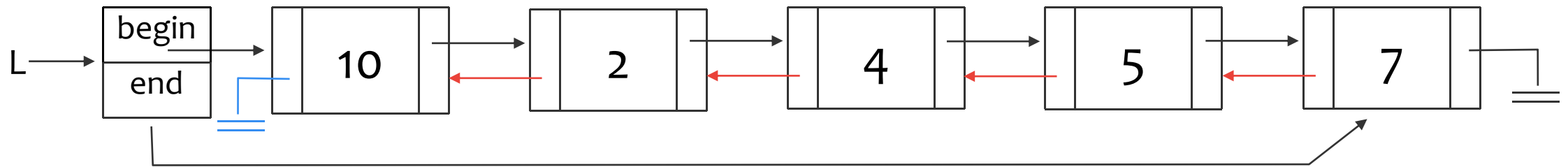
Listas Duplamente Encadeadas

Prof. dr. Samuel Martins (Samuka)
@xavecoding @hisamuka



Lista Duplamente Encadeada

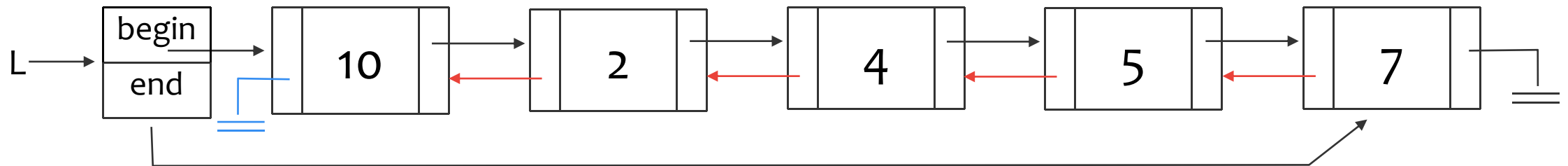
- Variação de **lista encadeada simples**;
- O nó agora também aponta para seu **antecessor**;
- O nó **anterior** do primeiro nó é NULL;



Definição

```
typedef struct _doubly_node {  
    int val;  
    struct _doubly_node *prev;  
    struct _doubly_node *next;  
} DoublyNode, Node;
```

```
typedef struct _doubly_linked_list {  
    DoublyNode *begin;  
    DoublyNode *end;  
    size_t size;  
} DoublyLinkedList, List;
```



Lista Duplamente Encadeada

Diversos tipos de operações:

- Inserção na cabeça (início) da lista;
- Impressão dos Elementos da Lista
- Inserção na cauda (fim) da lista;
- Remover elementos da lista;
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

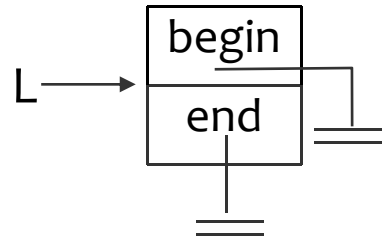
Lista Duplamente Encadeada

Diversos tipos de operações:

- **Inserção na cabeça (início) da lista;**
- Impressão dos Elementos da Lista
- Inserção na cauda (fim) da lista;
- Remover elementos da lista;
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

Inserção na cabeça (início) da lista

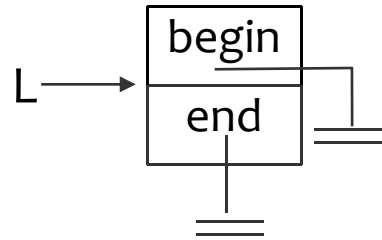
Caso 1: Lista está vazia



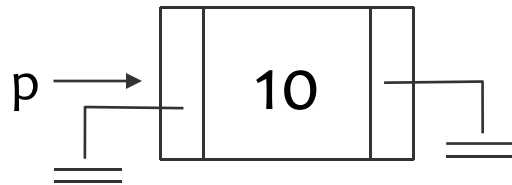
inserir 10

Inserção na cabeça (início) da lista

Caso 1: Lista está vazia



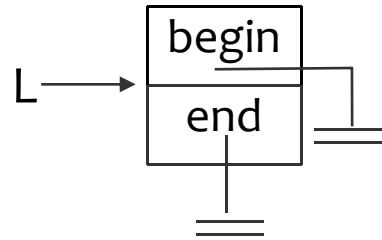
inserir 10



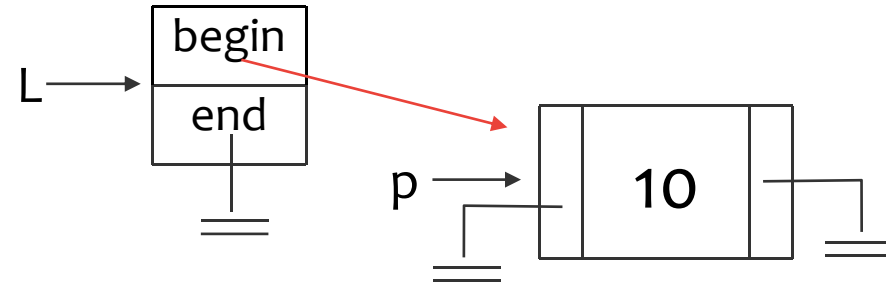
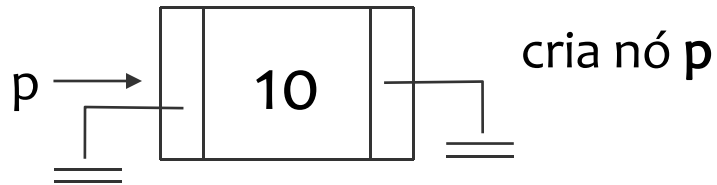
cria nó **p**

Inserção na cabeça (início) da lista

Caso 1: Lista está vazia

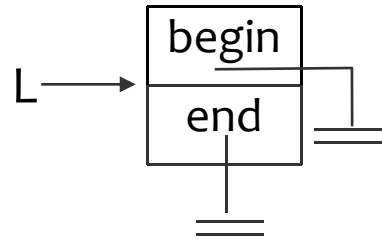


inserir 10

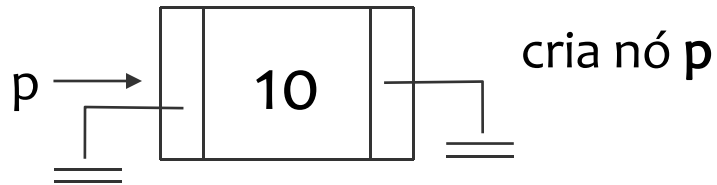


Inserção na cabeça (início) da lista

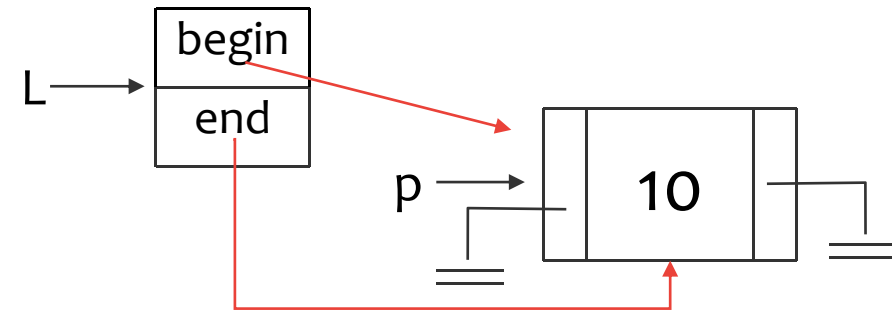
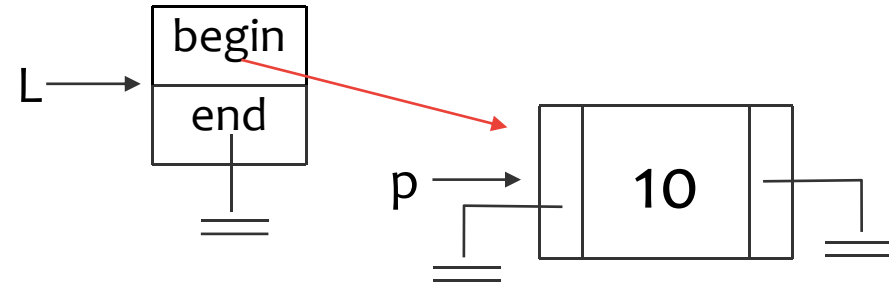
Caso 1: Lista está vazia



inserir 10

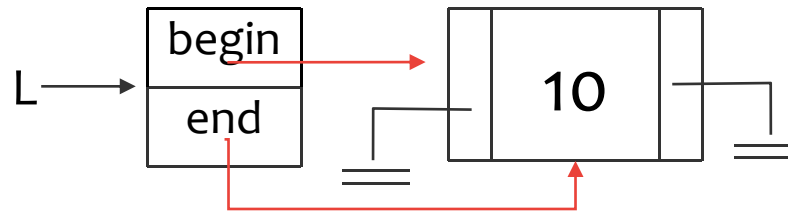


cria nó p



Inserção na cabeça (início) da lista

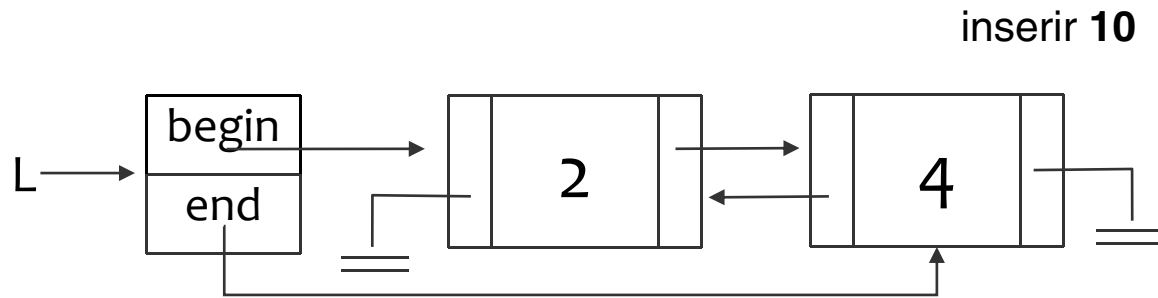
Caso 1: Lista está vazia



configuração final da lista
após a inserção

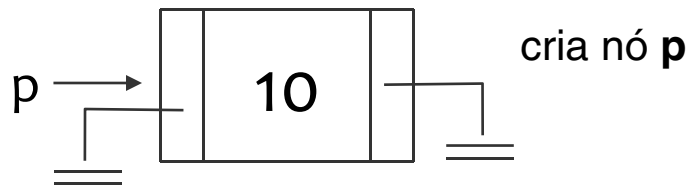
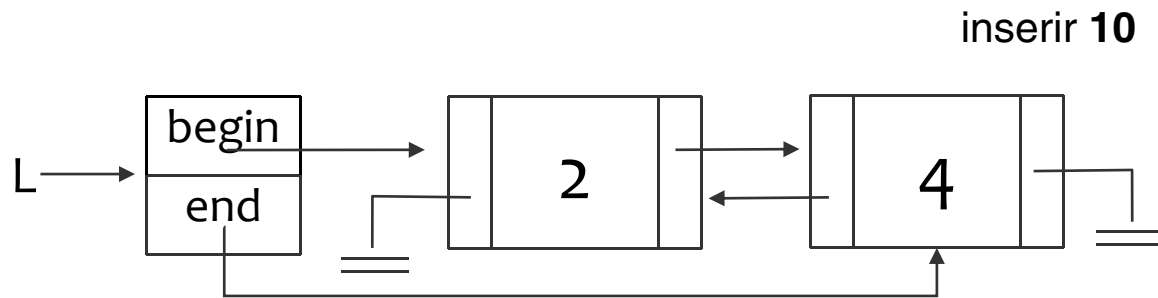
Inserção na cabeça (início) da lista

Caso 2: Lista possui elementos



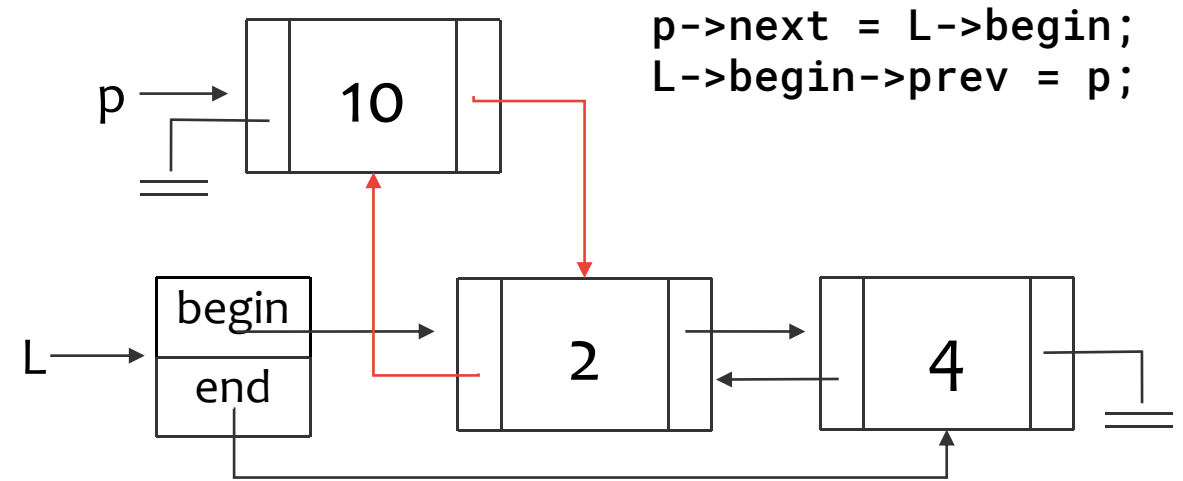
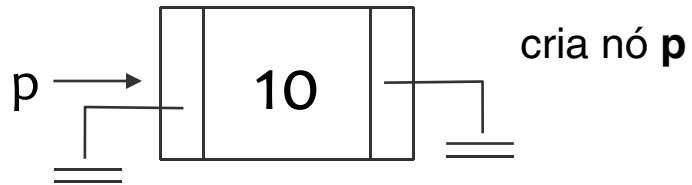
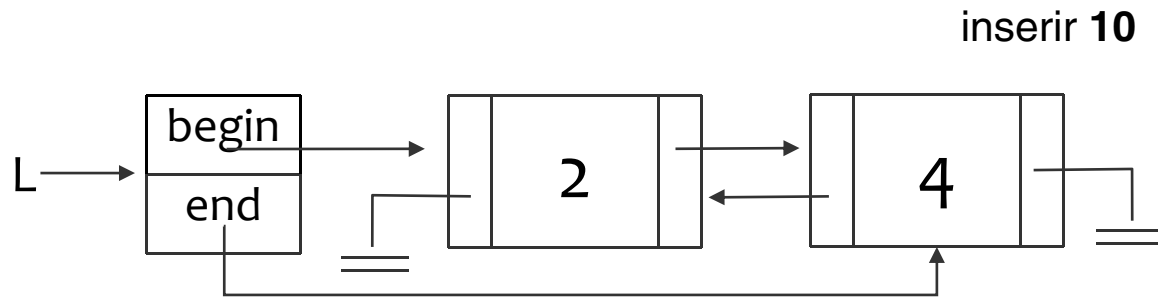
Inserção na cabeça (início) da lista

Caso 2: Lista possui elementos



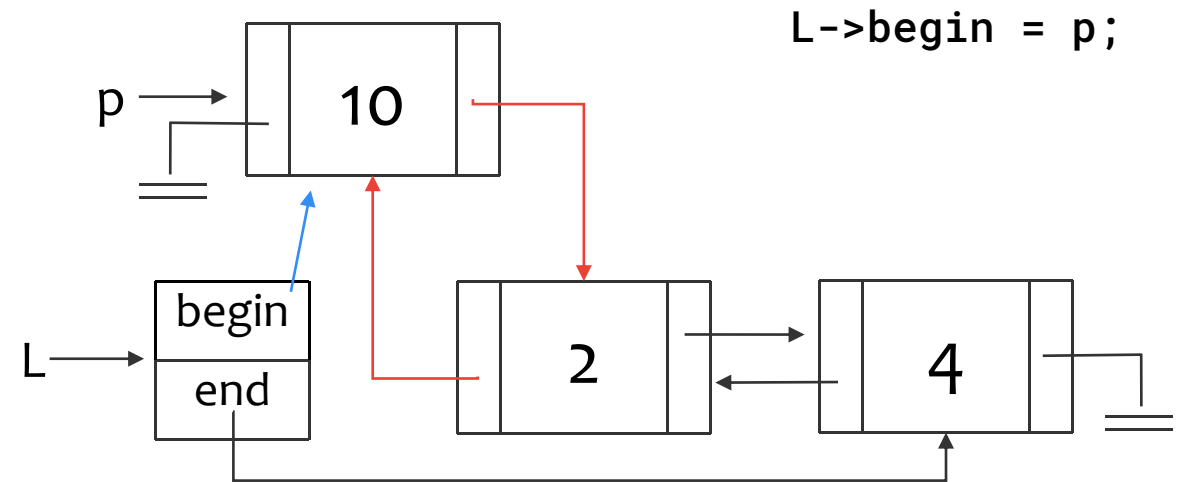
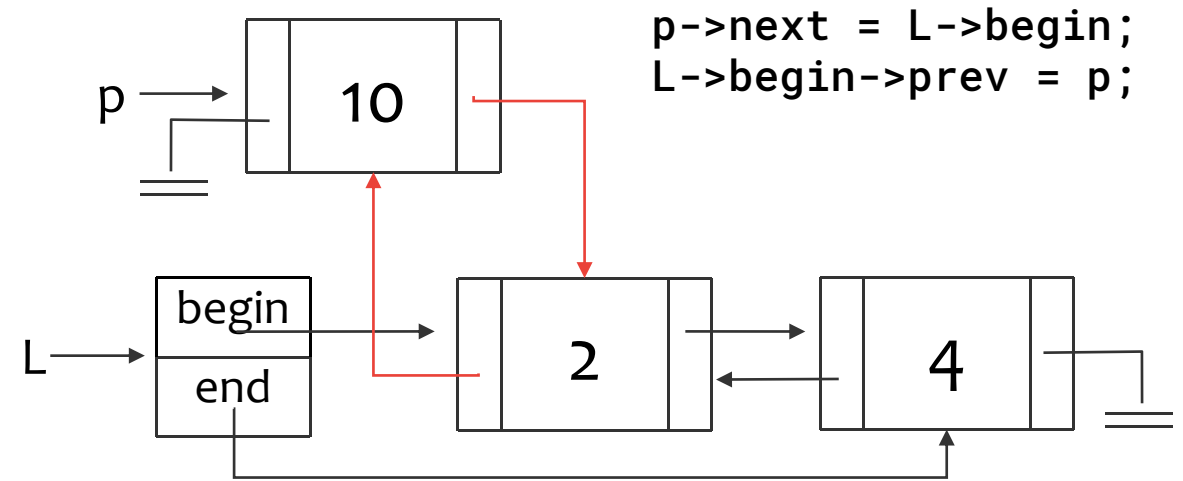
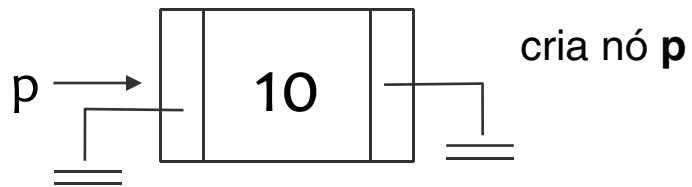
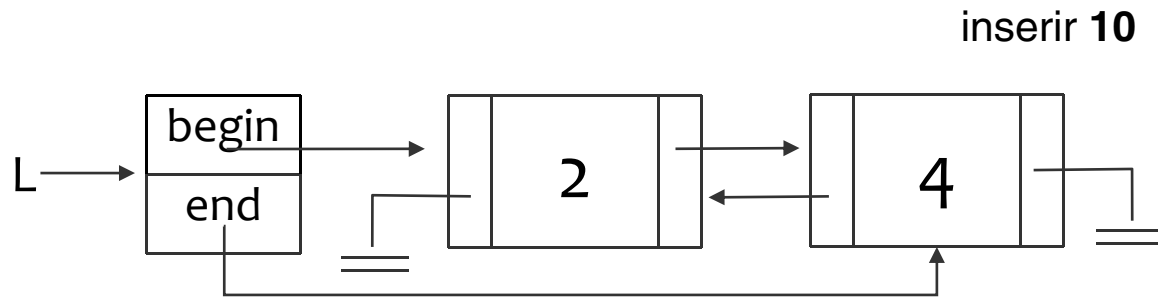
Inserção na cabeça (início) da lista

Caso 2: Lista possui elementos



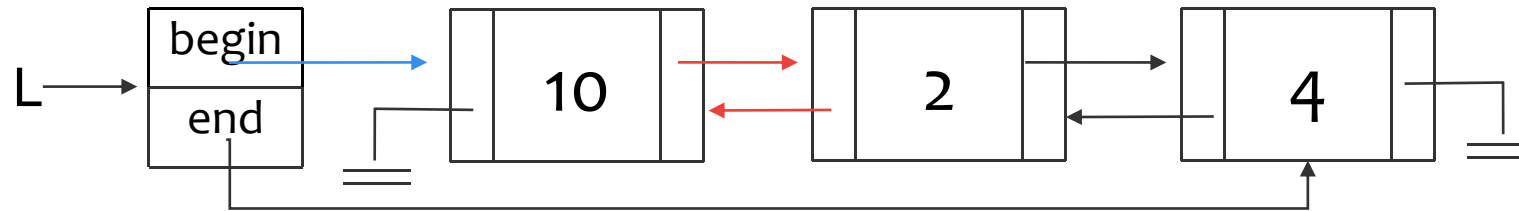
Inserção na cabeça (início) da lista

Caso 2: Lista possui elementos



Inserção na cabeça (início) da lista

Caso 2: Lista possui elementos



configuração final da
lista após a inserção

Lista Duplamente Encadeada

Diversos tipos de operações:

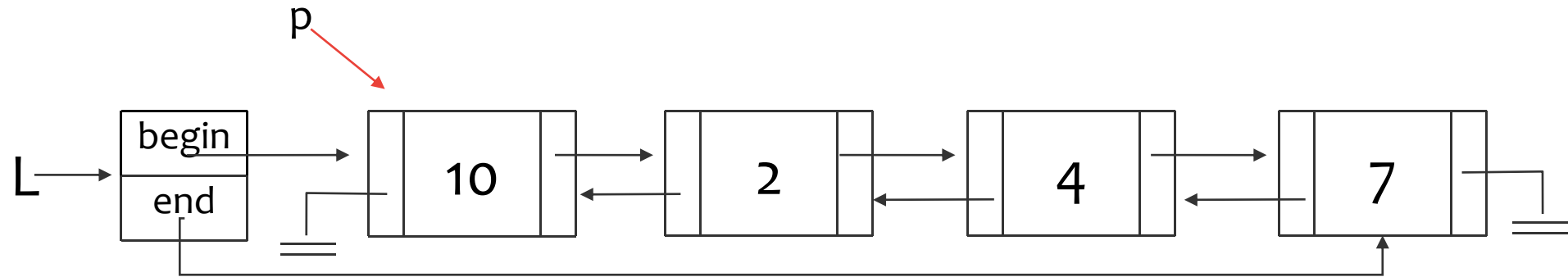
- **Inserção na cabeça (início) da lista;**
- Impressão dos Elementos da Lista
- Inserção na cauda (fim) da lista;
- Remover elementos da lista;
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

Lista Duplamente Encadeada

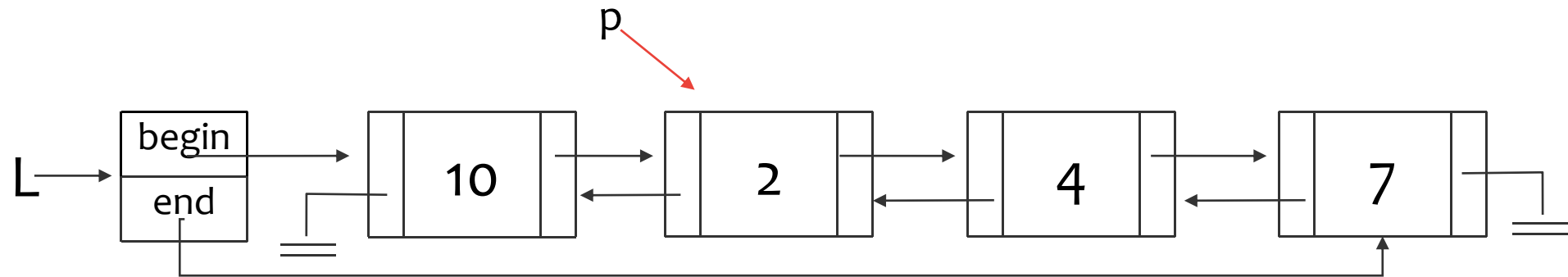
Diversos tipos de operações:

- Inserção na cabeça (início) da lista;
- **Impressão dos Elementos da Lista**
- Inserção na cauda (fim) da lista;
- Remover elementos da lista;
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

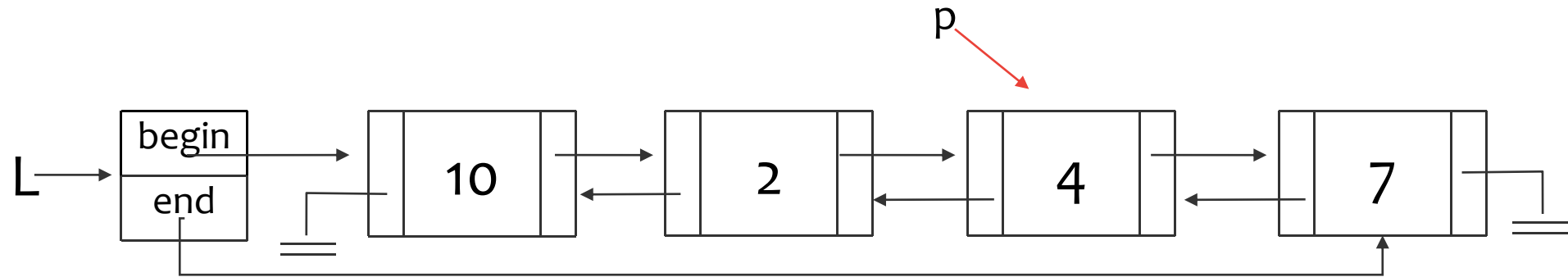
Impressão dos Elementos da Lista



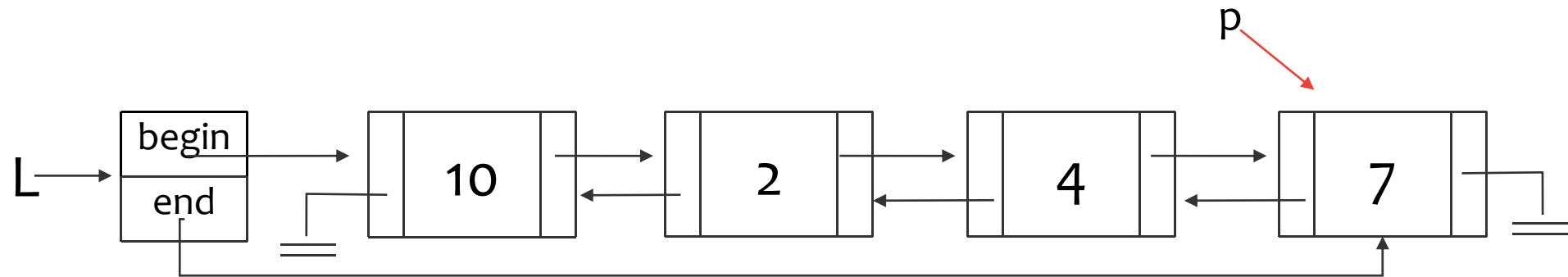
Impressão dos Elementos da Lista



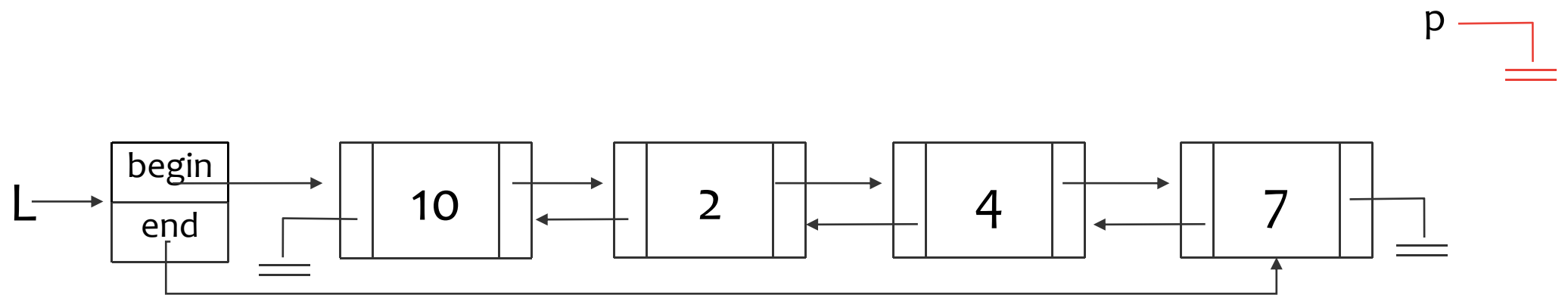
Impressão dos Elementos da Lista



Impressão dos Elementos da Lista



Impressão dos Elementos da Lista



Lista Duplamente Encadeada

Diversos tipos de operações:

- Inserção na cabeça (início) da lista;
- **Impressão dos Elementos da Lista**
- Inserção na cauda (fim) da lista;
- Remover elementos da lista;
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

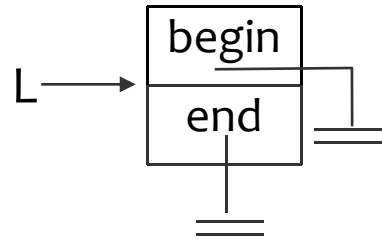
Lista Duplamente Encadeada

Diversos tipos de operações:

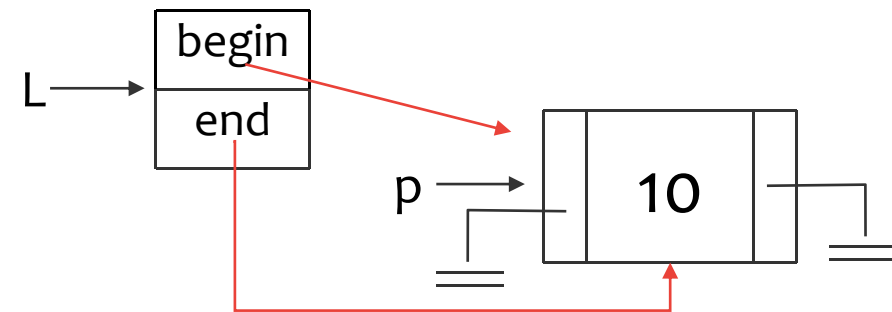
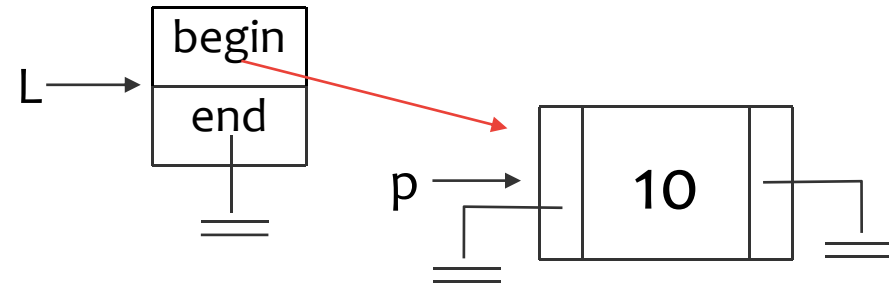
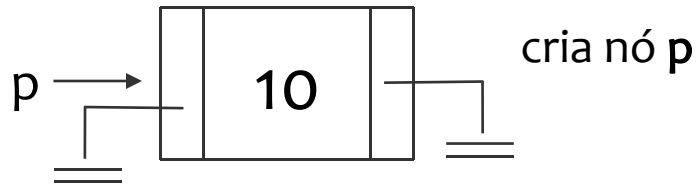
- Inserção na cabeça (início) da lista;
- Impressão dos Elementos da Lista
- **Inserção na cauda (fim) da lista;**
- Remover elementos da lista;
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

Inserção na cauda (fim) da lista

Caso 1: Lista está vazia



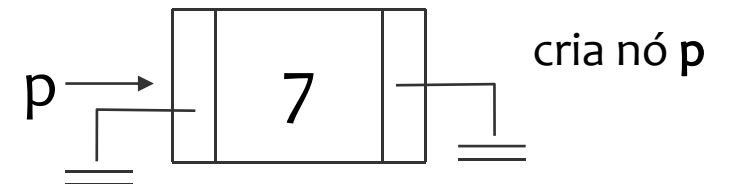
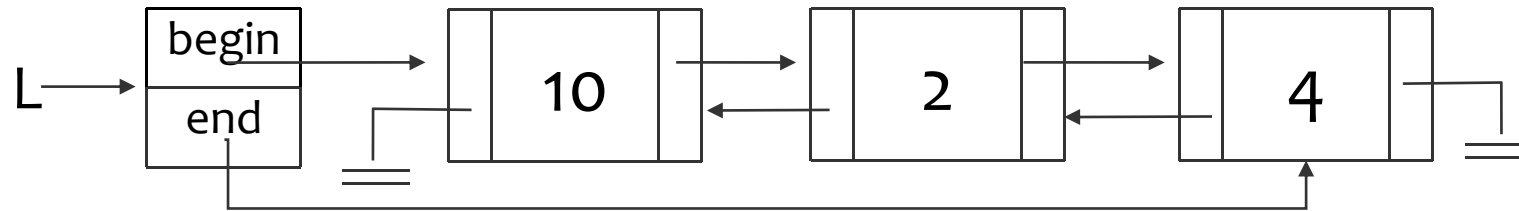
inserir 10



Inserção na cauda (fim) da lista

Caso 2: Lista possui elementos

inserir 7

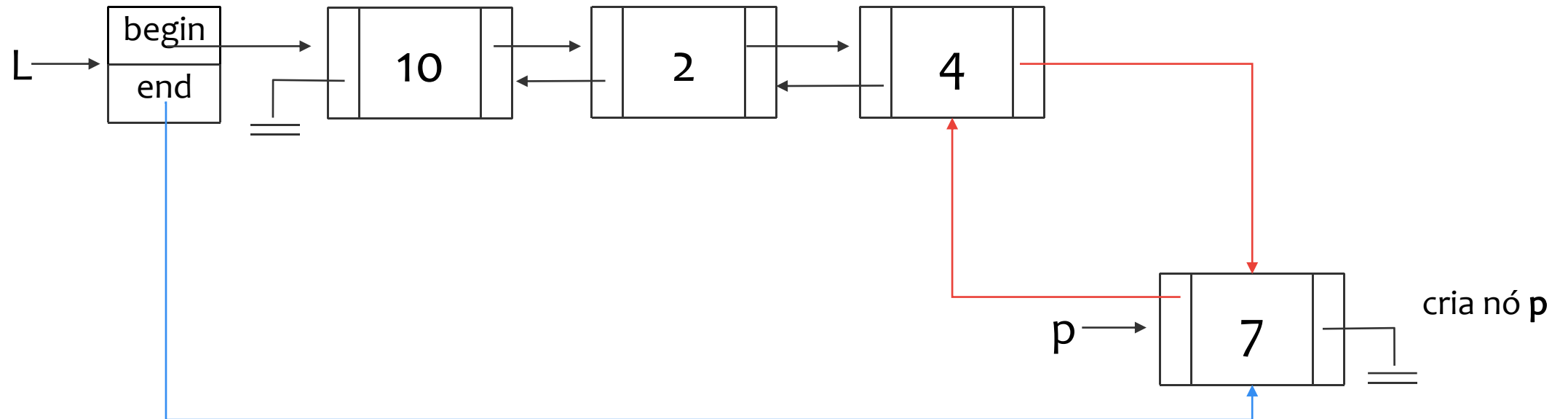


Inserção na cauda (fim) da lista

Caso 2: Lista possui elementos

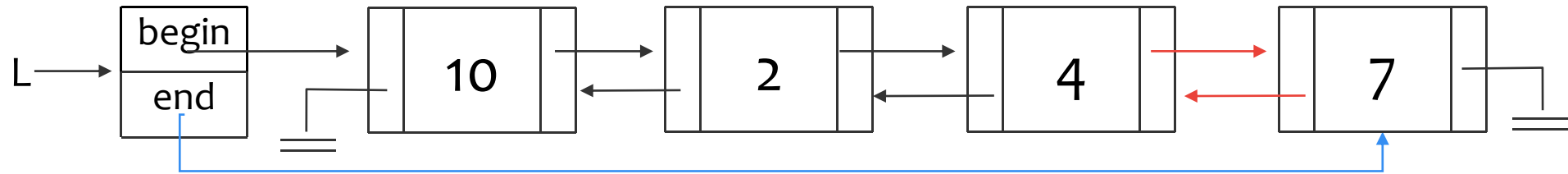
inserir 7

```
L->end->next = p;  
p->prev = L->end;  
L->end = p;
```



Inserção na cauda (fim) da lista

Caso 2: Lista possui elementos



configuração final da
lista após a inserção

Lista Duplamente Encadeada

Diversos tipos de operações:

- Inserção na cabeça (início) da lista;
- Impressão dos Elementos da Lista
- **Inserção na cauda (fim) da lista;**
- Remover elementos da lista;
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

Lista Duplamente Encadeada

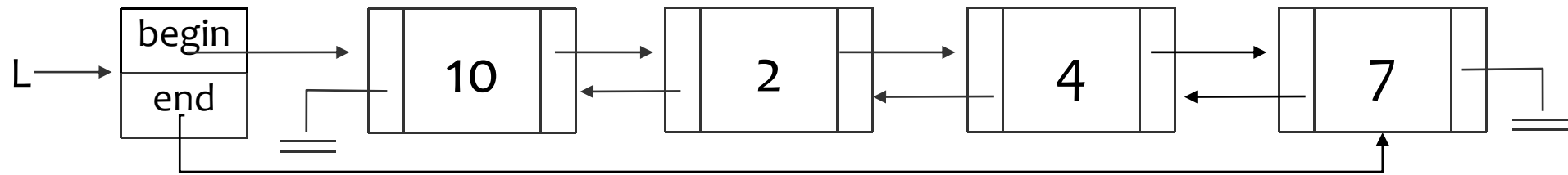
Diversos tipos de operações:

- Inserção na cabeça (início) da lista;
- Impressão dos Elementos da Lista
- Inserção na cauda (fim) da lista;
- **Remover elementos da lista;**
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

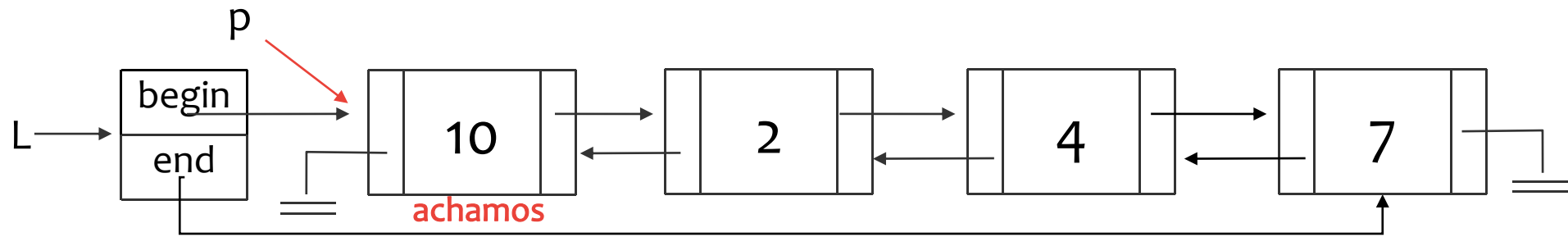
remover 10



Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

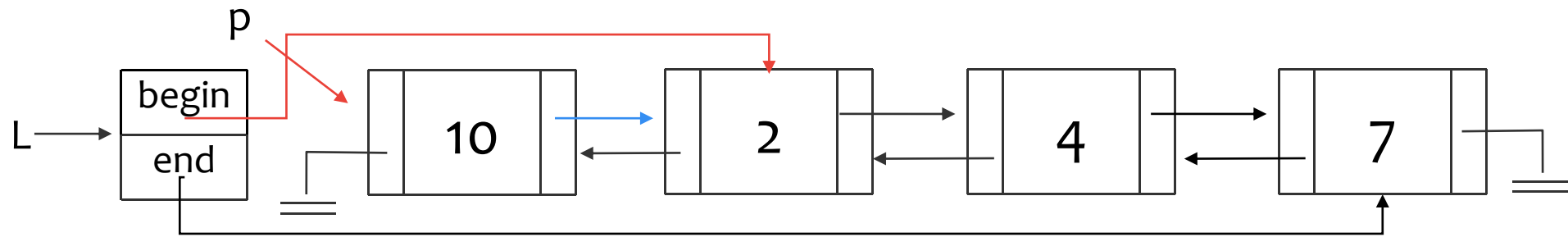
remover 10



Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

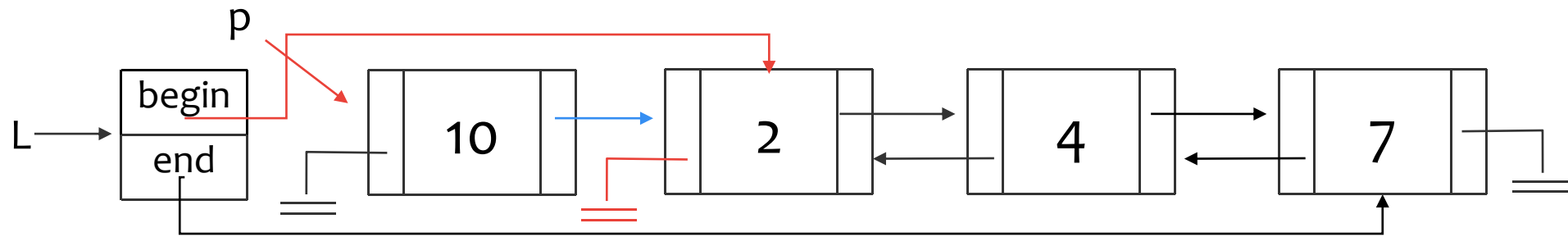
remover 10



Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

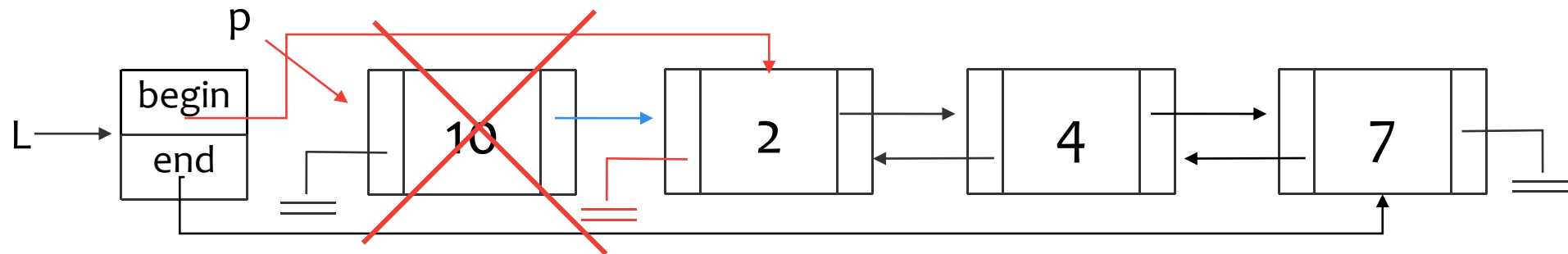
remover 10



Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

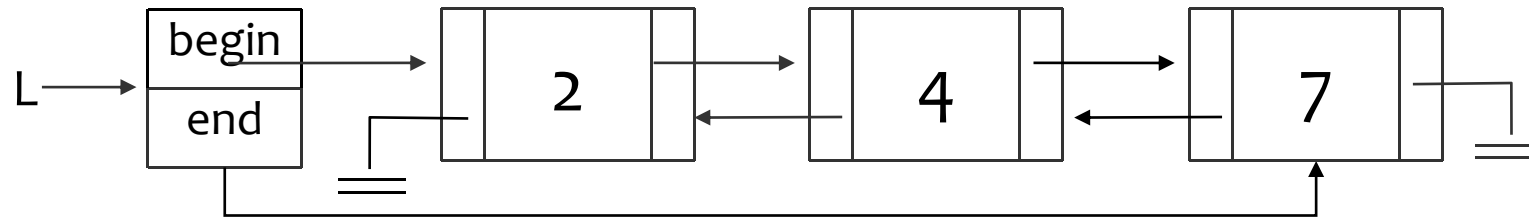
remover 10



Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

remover 10



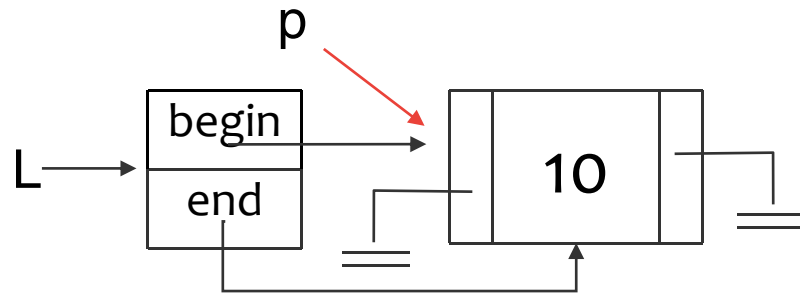
configuração final da
lista após a inserção

Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

- A lista só tem um elemento atualizar o ponteiro **end**

remover 10

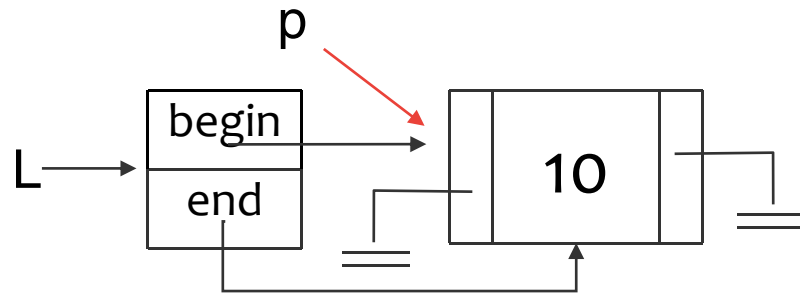


Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

- A lista só tem um elemento atualizar o ponteiro **end**

remover 10

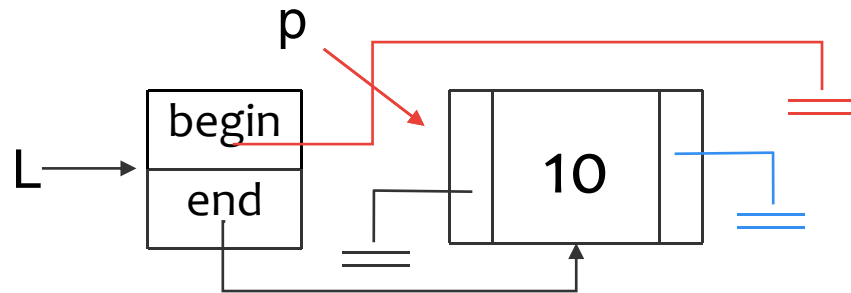


Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

- A lista só tem um elemento atualizar o ponteiro **end**

remover 10

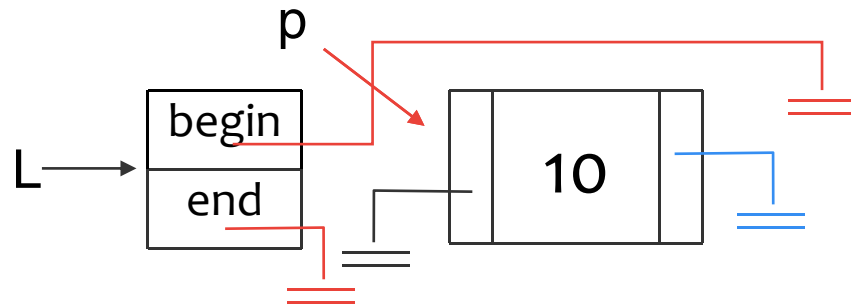


Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

- A lista só tem um elemento atualizar o ponteiro **end**

remover 10

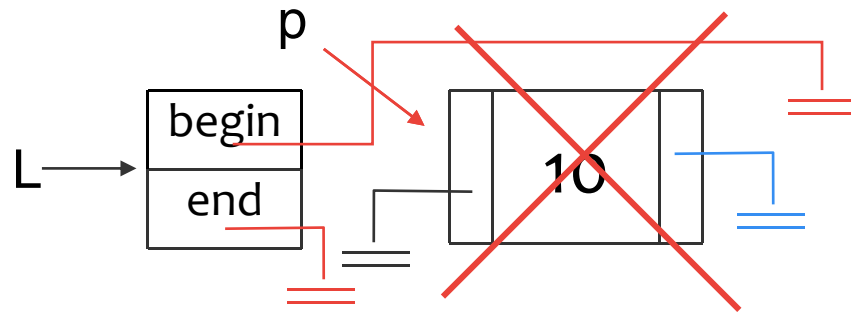


Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

- A lista só tem um elemento atualizar o ponteiro **end**

remover 10

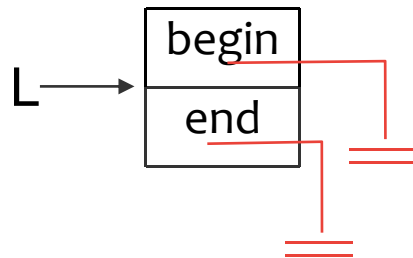


Remoção de Elementos da Lista

Caso 1: Elemento está na **cabeça (início)** da lista

- A lista só tem um elemento atualizar o ponteiro **end**

remover 10

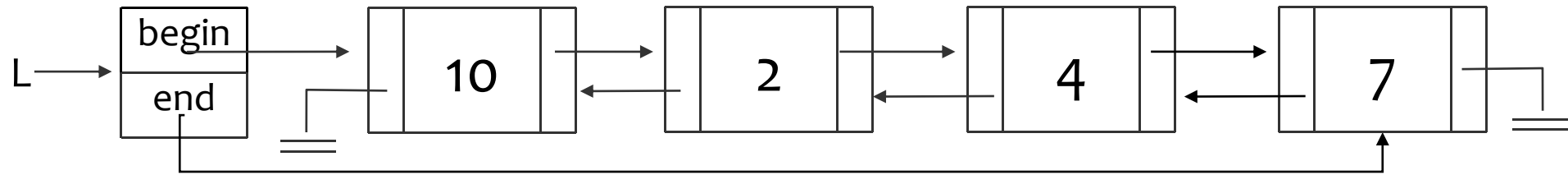


configuração final da lista

Remoção de Elementos da Lista

Caso 2: Elemento está no **meio** da lista

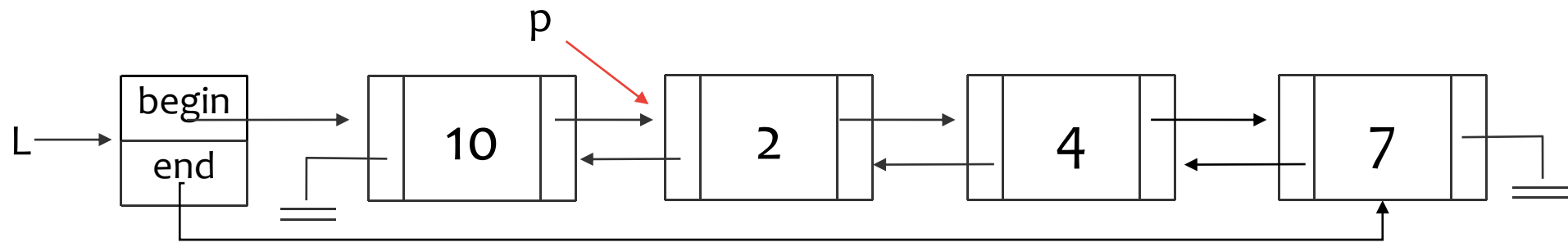
remove 4



Remoção de Elementos da Lista

Caso 2: Elemento está no **meio** da lista

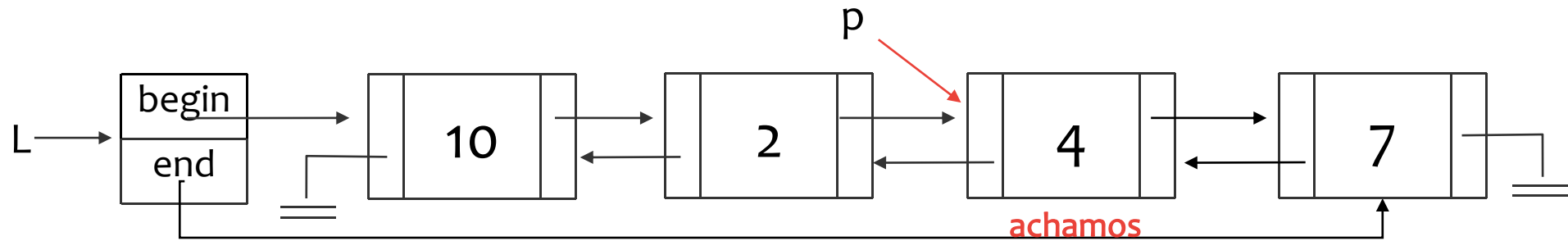
remove 4



Remoção de Elementos da Lista

Caso 2: Elemento está no **meio** da lista

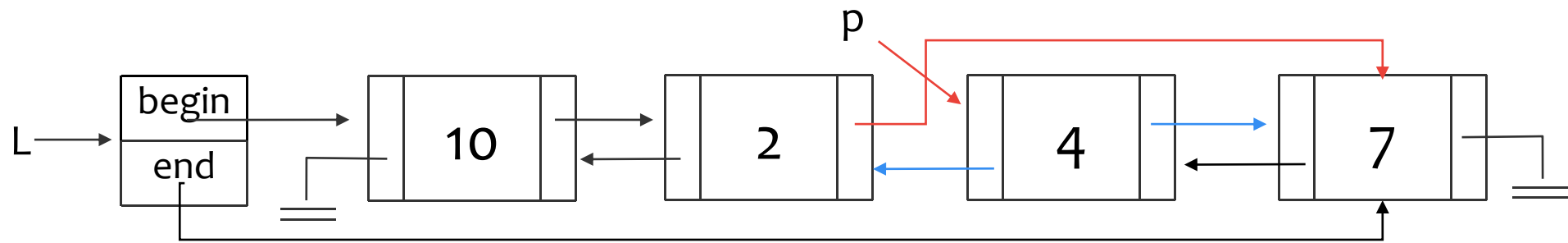
remove 4



Remoção de Elementos da Lista

Caso 2: Elemento está no **meio** da lista

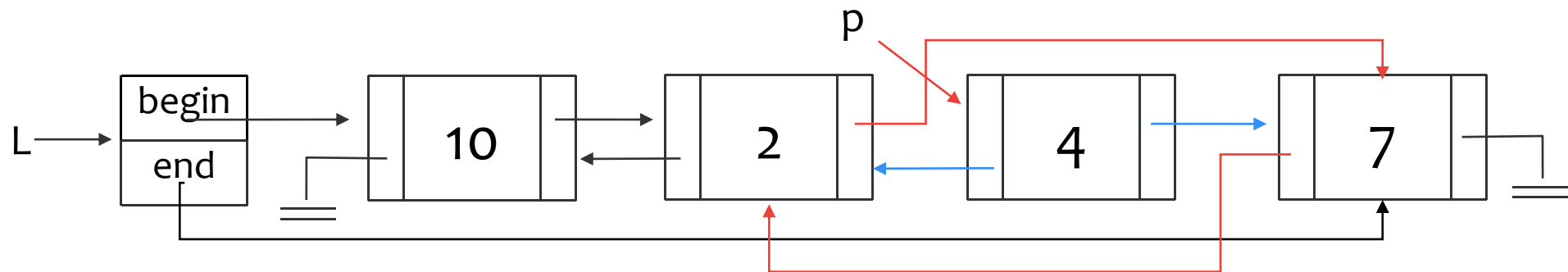
remove 4



Remoção de Elementos da Lista

Caso 2: Elemento está no **meio** da lista

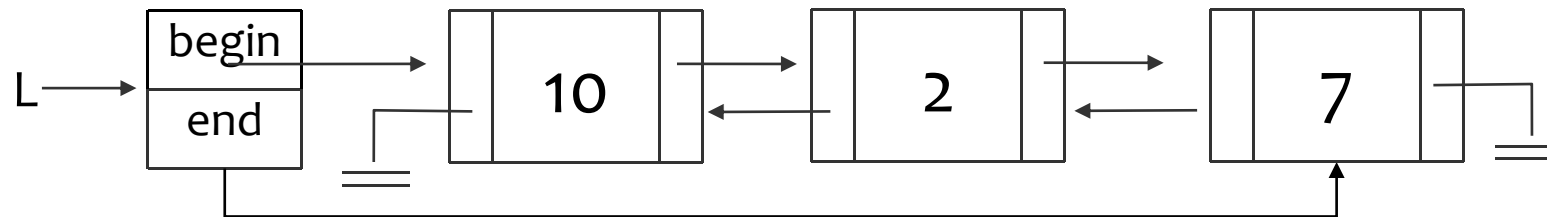
remove 4



Remoção de Elementos da Lista

Caso 2: Elemento está no **meio** da lista

remove 4

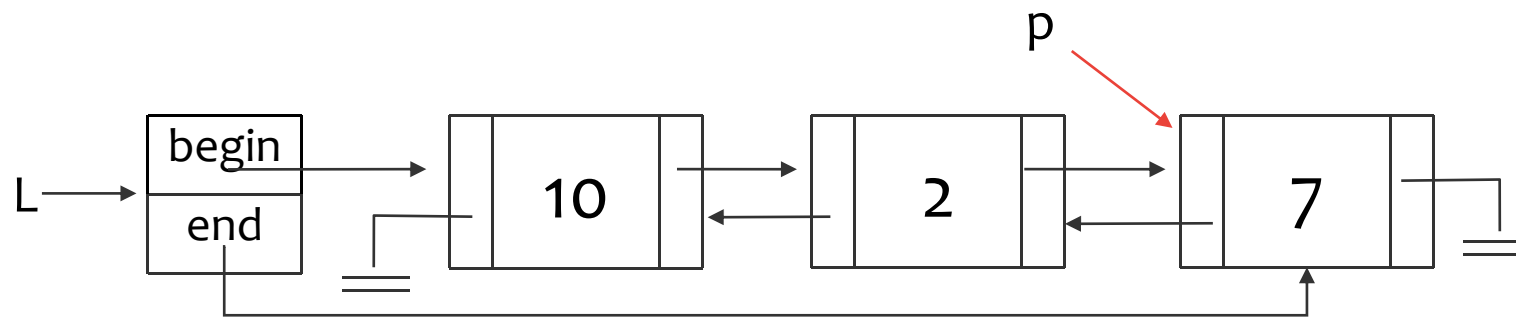


configuração final da lista

Remoção de Elementos da Lista

Caso 3: Elemento está na **cauda (final)** da lista

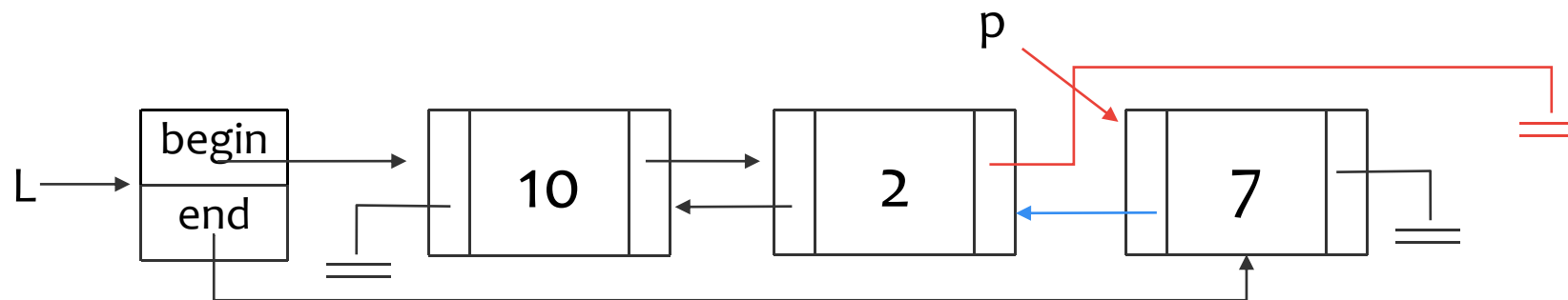
remove 7



Remoção de Elementos da Lista

Caso 3: Elemento está na **cauda (final)** da lista

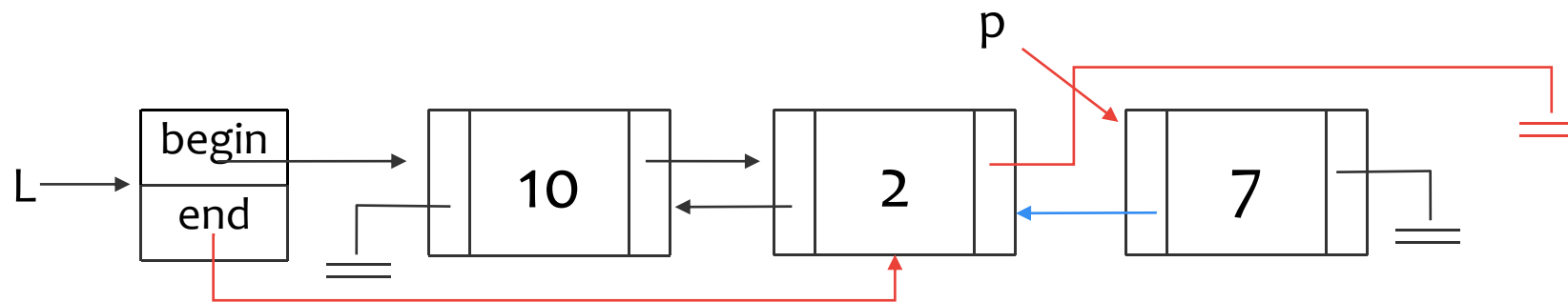
remover 7



Remoção de Elementos da Lista

Caso 3: Elemento está na **cauda (final)** da lista

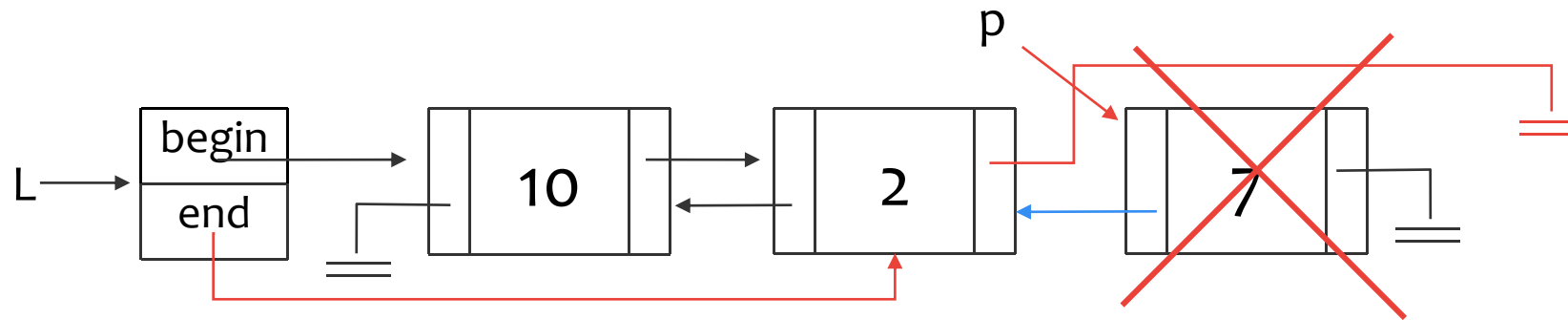
remove 7



Remoção de Elementos da Lista

Caso 3: Elemento está na **cauda (final)** da lista

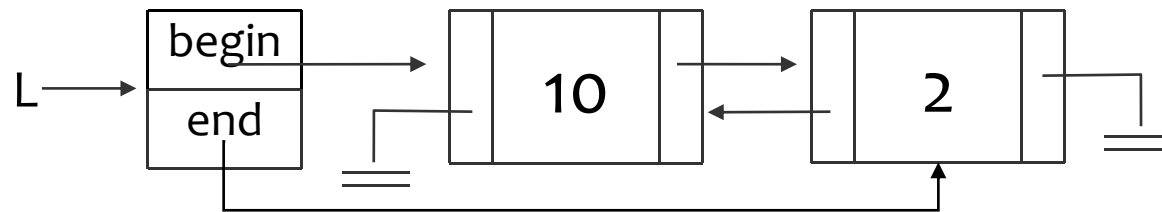
remover 7



Remoção de Elementos da Lista

Caso 3: Elemento está na **cauda (final)** da lista

remover 7



configuração final da lista

Lista Duplamente Encadeada

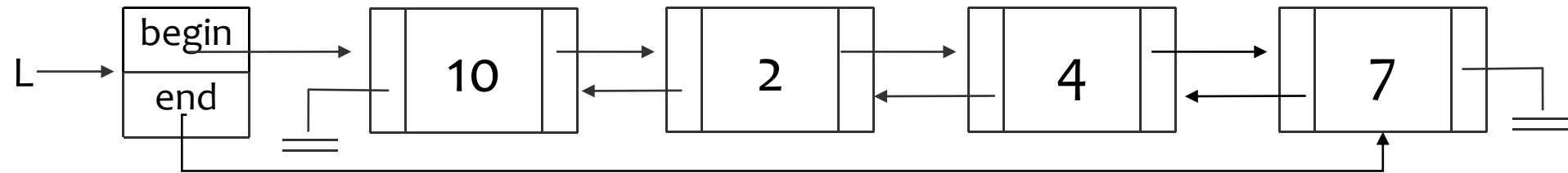
Diversos tipos de operações:

- Inserção na cabeça (início) da lista;
- Impressão dos Elementos da Lista
- Inserção na cauda (fim) da lista;
- **Remover elementos da lista;**
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i

Lista Duplamente Encadeada

Diversos tipos de operações:

- Inserção na cabeça (início) da lista;
- Impressão dos Elementos da Lista
- Inserção na cauda (fim) da lista;
- Remover elementos da lista;
- Contar o número de elementos da Lista;
- Verificar se a lista está vazia e retornar verdadeiro/falso
- Retornar o primeiro elemento;
- Retornar o último elemento;
- Retornar um elemento na posição i



Exercícios

- Inserir um elemento em uma dada posição
- Inverter uma lista
- Copiar/clonar uma lista;
- Apagar todos os elementos da lista
- Concatenar de duas listas;
- Ordenação de uma lista;
- Buscar um dado elemento na lista e retornar seu ponteiro;
- Inserção ordenada;

Dominando Estruturas de Dados 1

Listas Duplamente Encadeadas

Prof. dr. Samuel Martins (Samuka)
@xavecoding @hisamuka

