

Unidade V:

Estruturas de Dados Básicas

com Alocação Flexível - Lista Simples

Prof. Max do Val Machado



PUC Minas

Instituto de Ciências Exatas e Informática
Curso de Ciência da Computação

Código Fonte

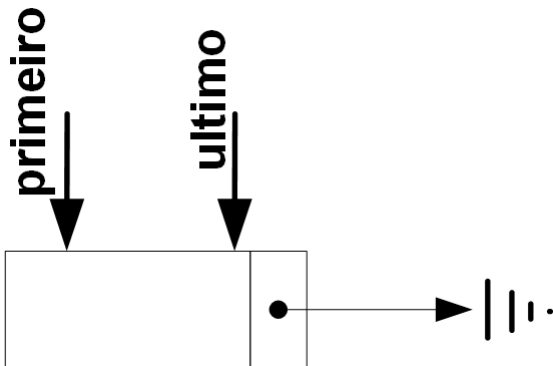
- [PrincipalLista.java](#), igual ao da estrutura sequencial
- [Lista.java](#), tem os atributos primeiro e início e os métodos abaixo:
 - Inserir no início
 - Inserir no fim
 - Inserir
 - Remover no início
 - Remover no fim
 - Remover

Classe Lista Simples

```
class Lista {  
    private Celula primeiro, ultimo;  
    public Lista () {  
        primeiro = new Celula();  
        ultimo = primeiro;  
    }  
    public void inserirInicio(int x) { ... }  
    public void inserirFim(int x) { ... }  
    public int removerInicio() { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
    public void mostrar() { ... }  
}
```

Classe Lista Simples

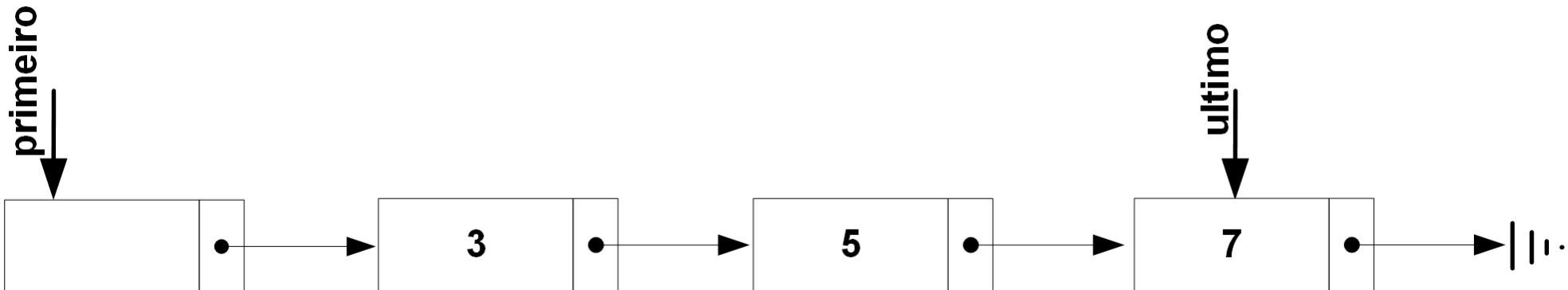
```
class Lista {  
    private Celula primeiro, ultimo;  
    public Lista () {  
        primeiro = new Celula();  
        ultimo = primeiro;  
    }  
    public void inserirInicio(int x) { ... }  
    public void inserirFim(int x) { ... }  
    public int removerInicio() { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
    public void mostrar() { ... }  
}
```



Classe Lista Simples

```
class Lista {  
    private Celula primeiro, ultimo;  
    public Lista () {  
        primeiro = new Celula();  
        ultimo = primeiro;  
    }  
    public void inserirInicio(int x) { ... }  
    public void inserirFim(int x) { ... }  
    public int removerInicio() { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
    public void mostrar() { ... }  
}
```

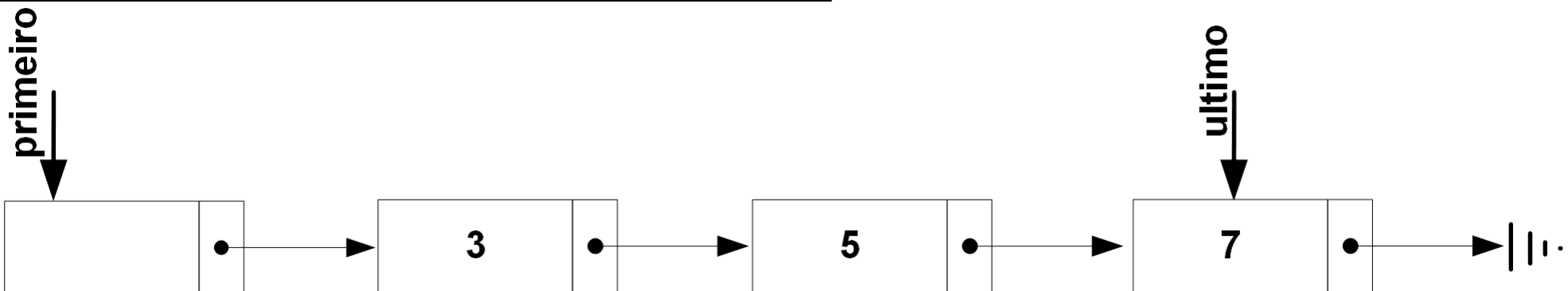
Iguais aos métodos da fila



Classe Lista Simples

```
class Lista {  
    private Celula primeiro, ultimo;  
    public Lista () {  
        primeiro = new Celula();  
        ultimo = primeiro;  
    }  
    public void inserirInicio(int x) { ... }  
    public void inserirFim(int x) { ... }  
    public int removerInicio() { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
    public void mostrar() { ... }  
}
```

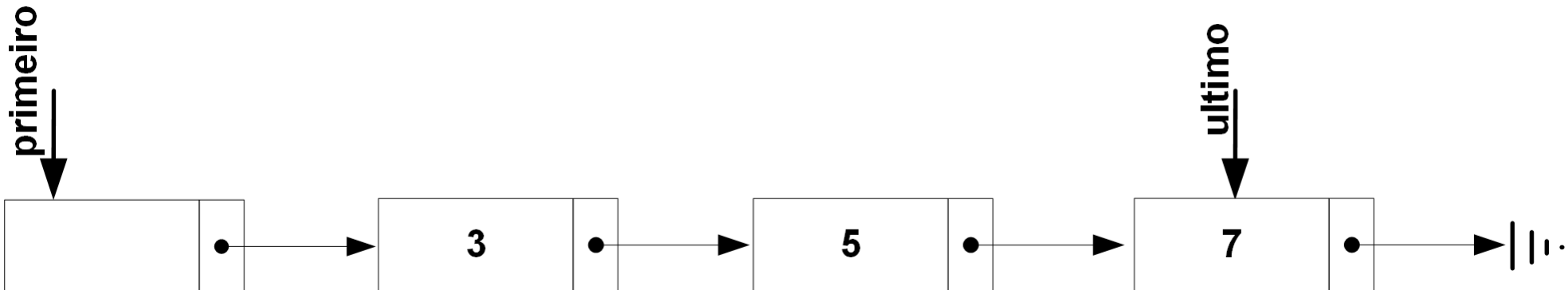
Igual aos da fila/pilha



Classe Lista Simples

```
class Lista {  
    private Celula primeiro, ultimo;  
    public Lista () {  
        primeiro = new Celula();  
        ultimo = primeiro;  
    }  
    public void inserirInicio(int x) { ... }  
    public void inserirFim(int x) { ... }  
    public int removerInicio() { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
    public void mostrar() { ... }  
}
```

Assim, ...



Classe Lista Simples

```
class Lista {
```

```
    public void inserirInicio(int x) { ... }
```

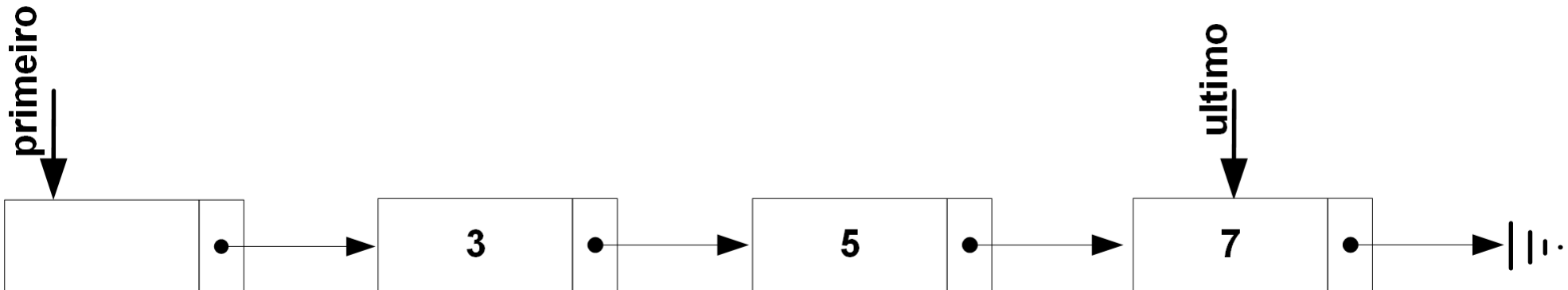
```
    public int removerFim() { ... }
```

```
    public void inserir(int x, int pos) { ... }
```

```
    public int remover(int pos) { ... }
```

```
}
```

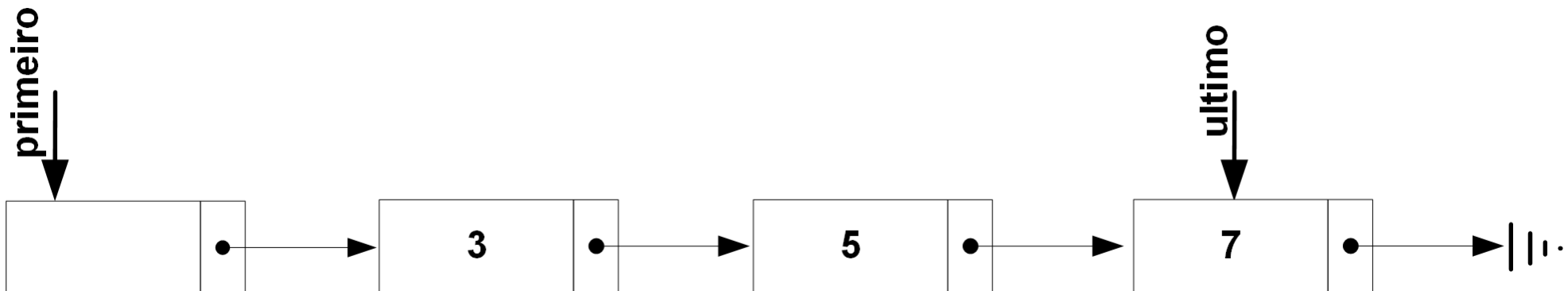
Assim, ...



Classe Lista Simples

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

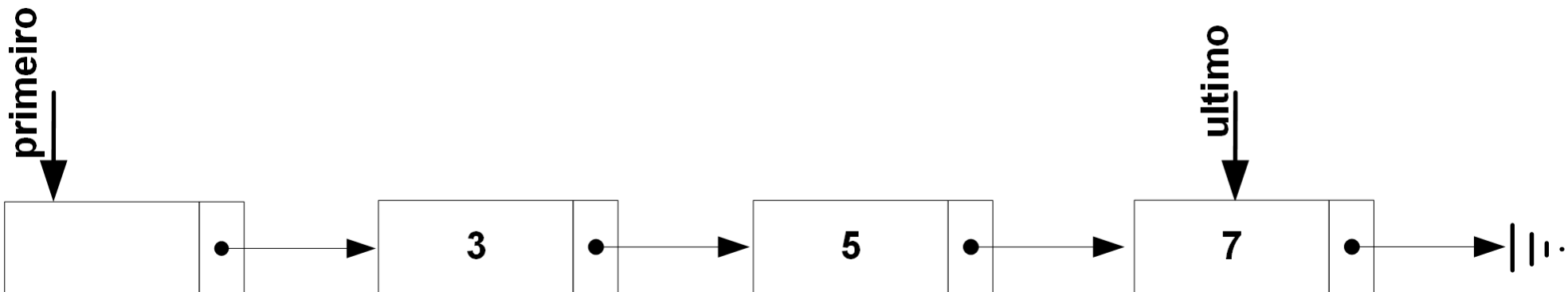
Assim, ...



Inserir no Início

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

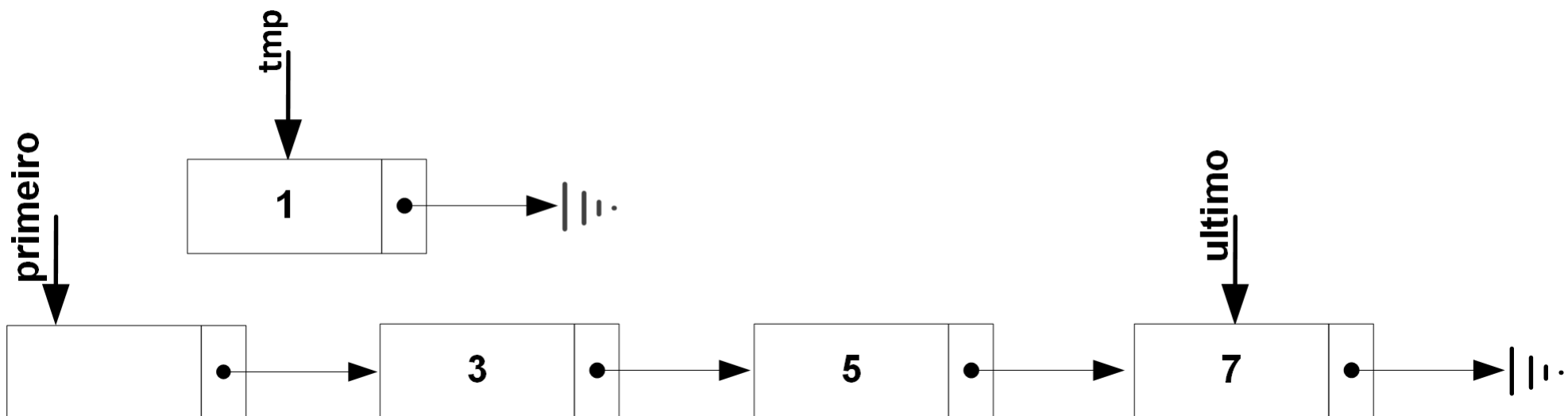
```
//inserirInicio(1)  
public void inserirInicio(int x) {  
    Celula tmp = new Celula(x);  
    tmp.prox = primeiro.prox;  
    primeiro.prox = tmp;  
    if (primeiro == ultimo) {  
        ultimo = tmp;  
    }  
    tmp = null;  
}
```



Inserir no Início

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

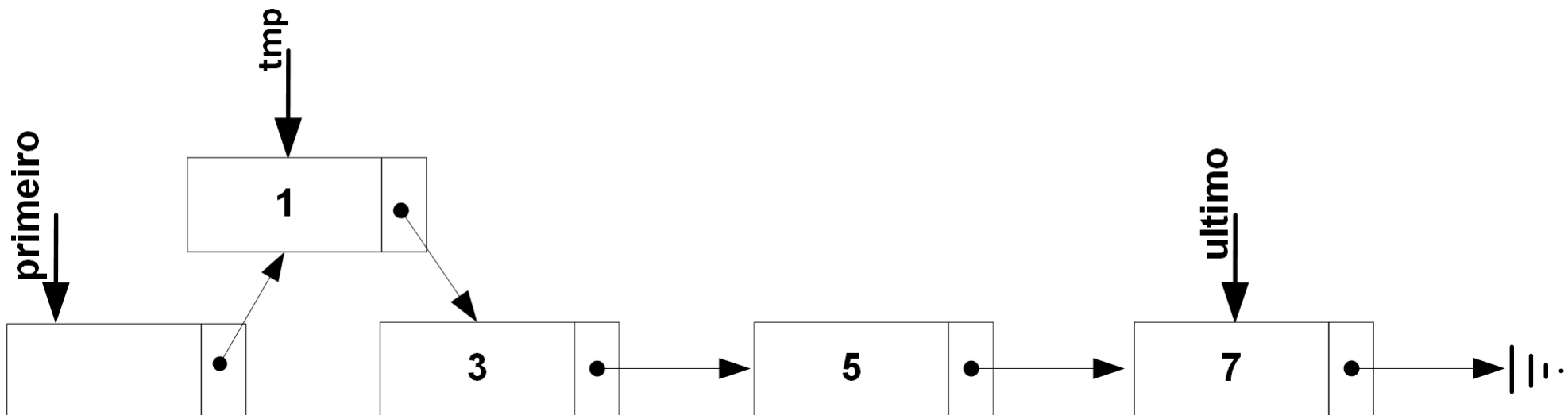
```
//inserirInicio(1)  
public void inserirInicio(int x) {  
    Celula tmp = new Celula(x);  
    tmp.prox = primeiro.prox;  
    primeiro.prox = tmp;  
    if (primeiro == ultimo) {  
        ultimo = tmp;  
    }  
    tmp = null;  
}
```



Inserir no Início

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

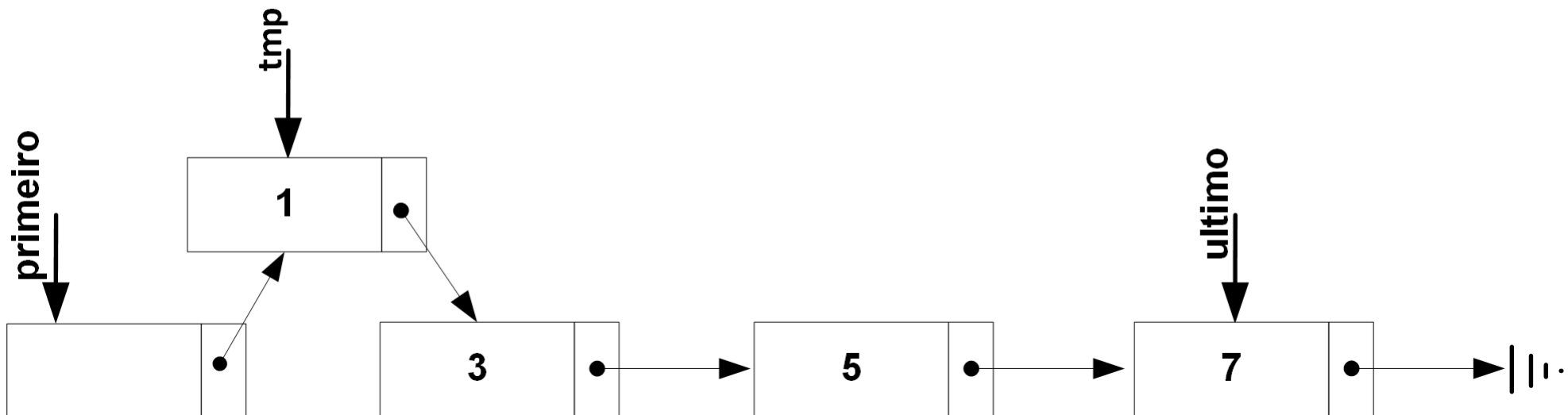
```
//inserirInicio(1)  
public void inserirInicio(int x) {  
    Celula tmp = new Celula(x);  
    tmp.prox = primeiro.prox;  
    primeiro.prox = tmp;  
    if (primeiro == ultimo) {  
        ultimo = tmp;  
    }  
    tmp = null;  
}
```



Inserir no Início

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

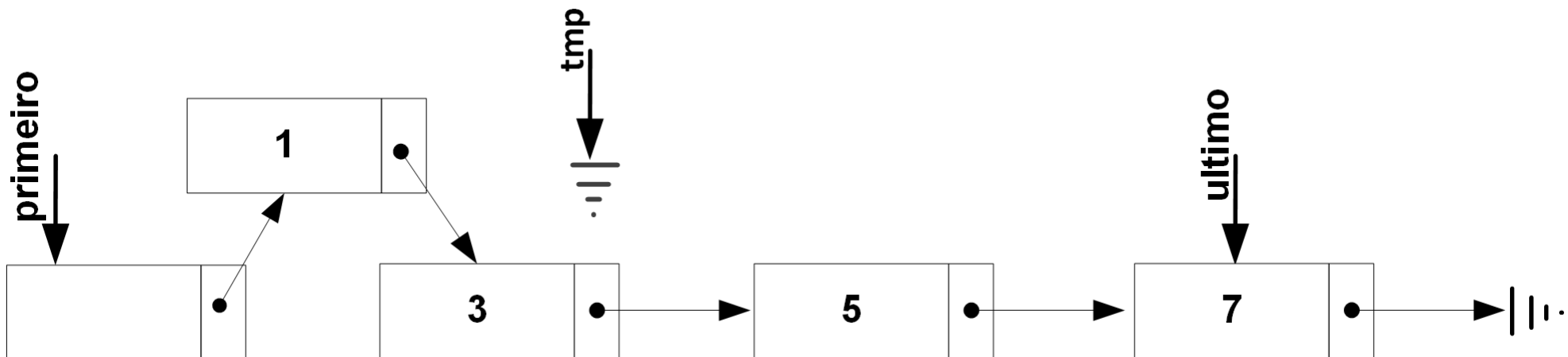
```
//inserirInicio(1)  
public void inserirInicio(int x) {  
    Celula tmp = new Celula(x);  
    tmp.prox = primeiro.prox;  
    primeiro.prox = tmp;  
    if (primeiro == ultimo) {  
        ultimo = tmp;  
    }  
    tmp = null;  
}
```



Inserir no Início

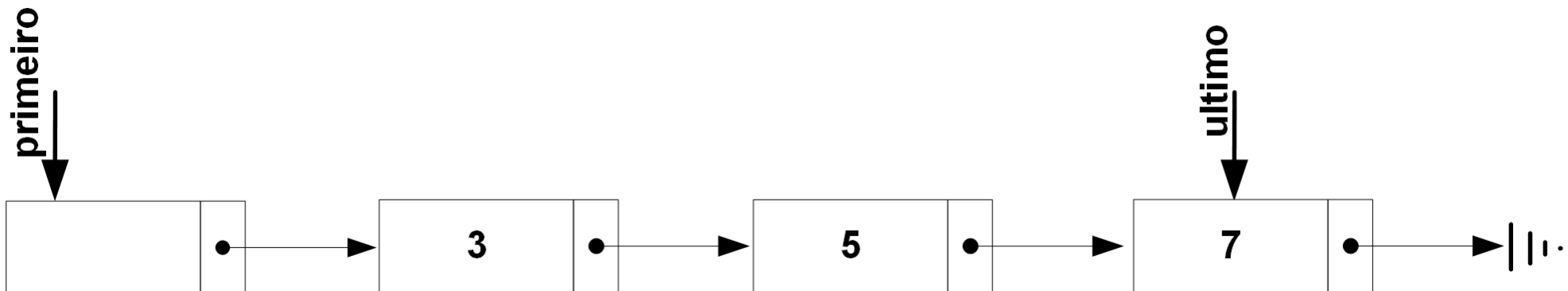
```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

```
//inserirInicio(1)  
public void inserirInicio(int x) {  
    Celula tmp = new Celula(x);  
    tmp.prox = primeiro.prox;  
    primeiro.prox = tmp;  
    if (primeiro == ultimo) {  
        ultimo = tmp;  
    }  
    tmp = null;  
}
```



Remover no Fim

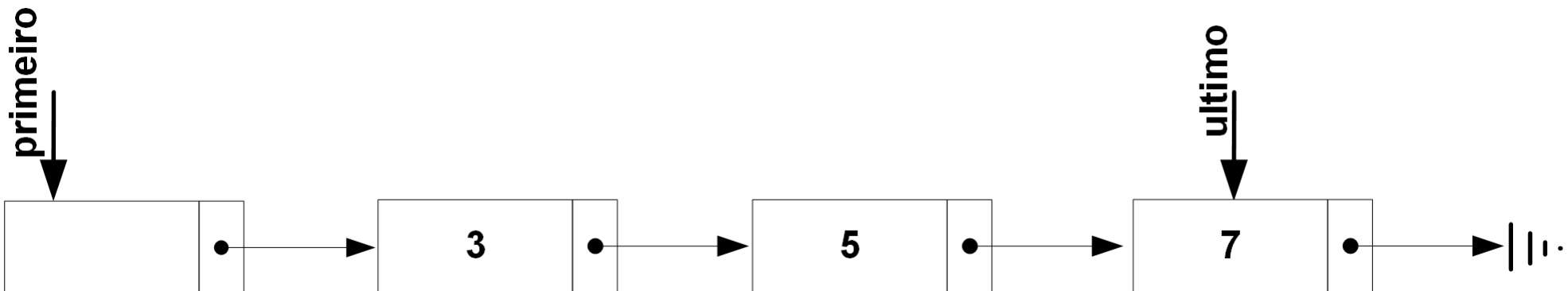
```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

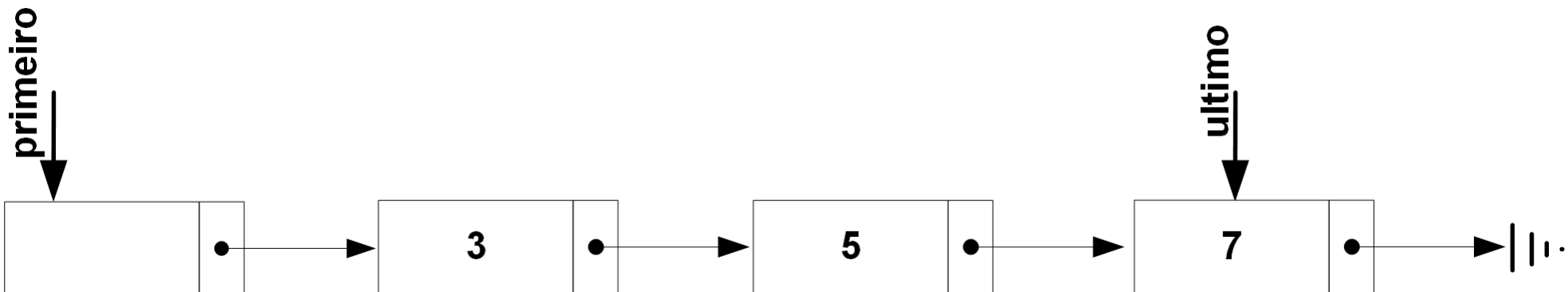
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

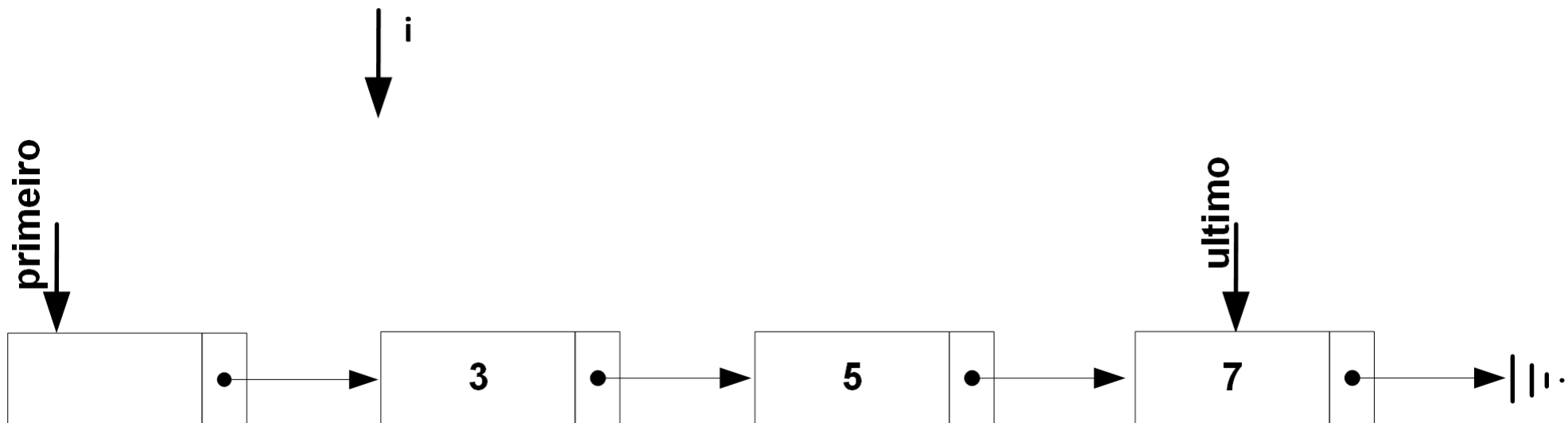
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

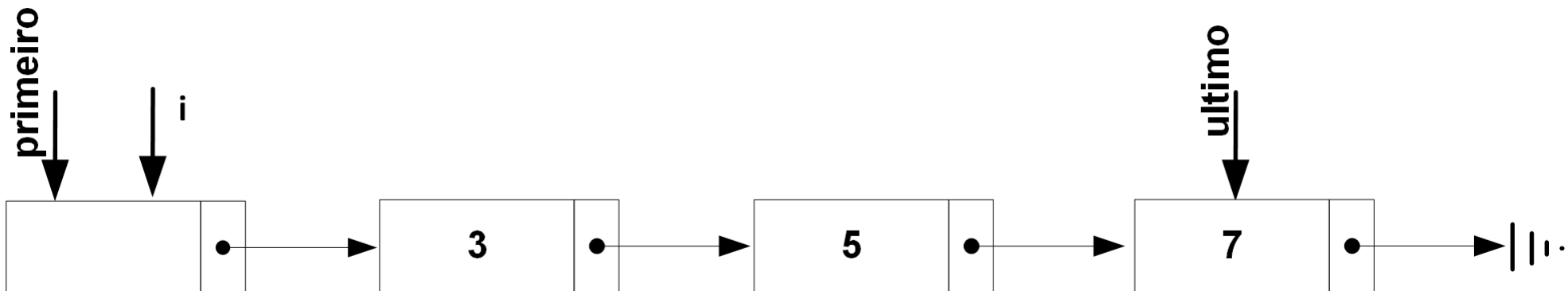
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

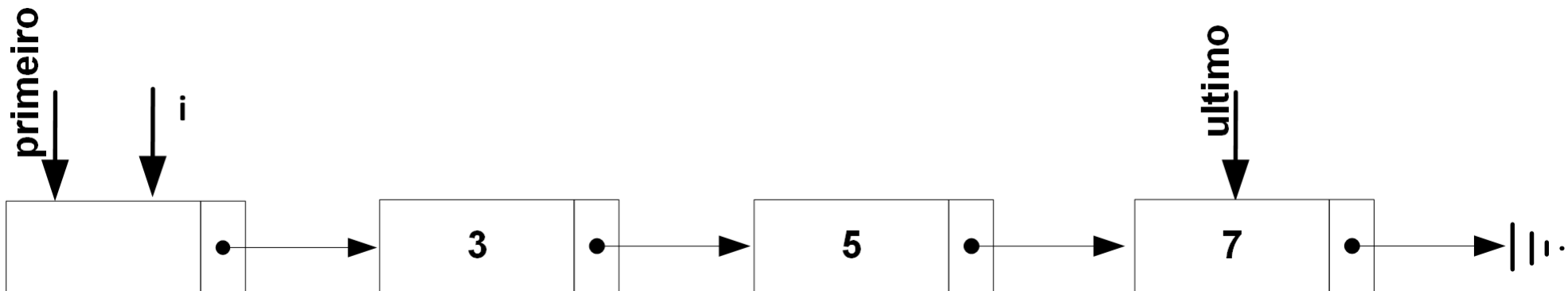
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for (i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

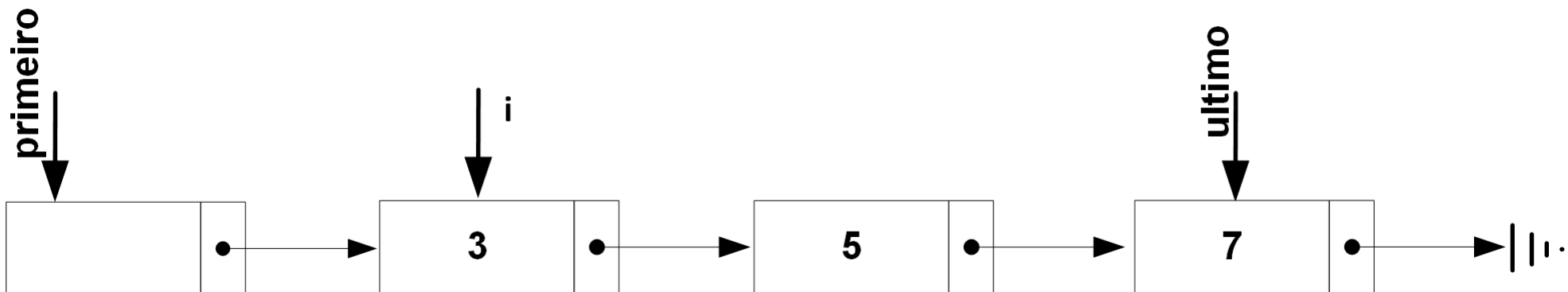
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null; true  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

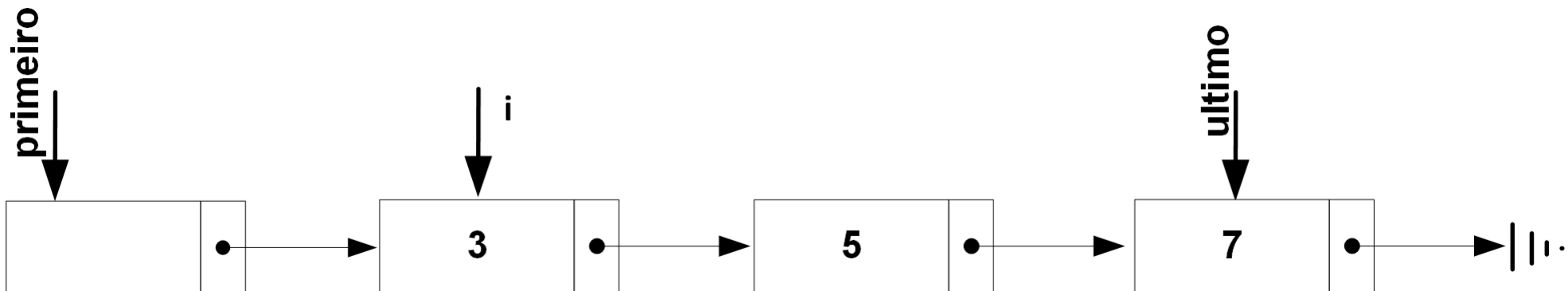
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

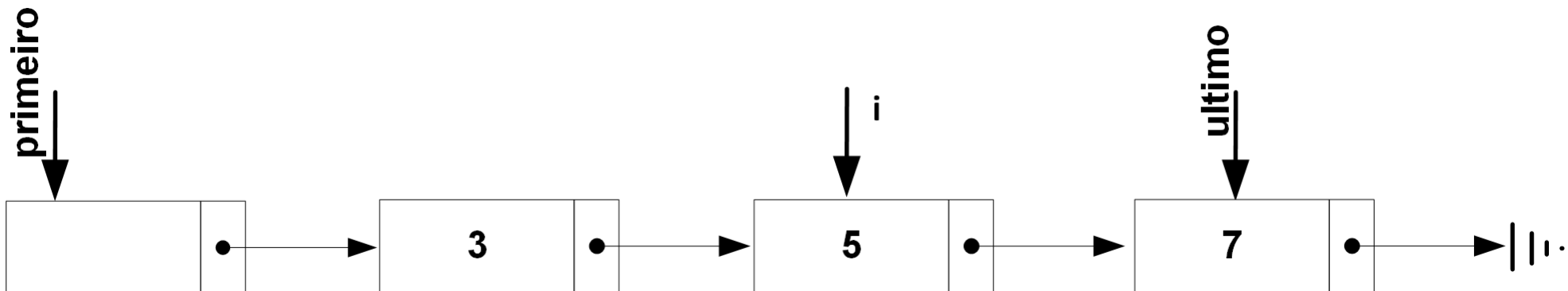
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

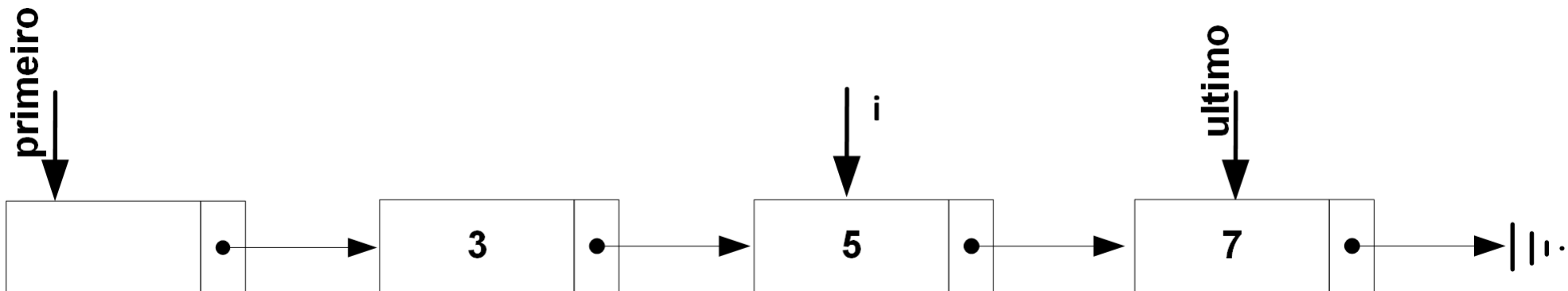
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

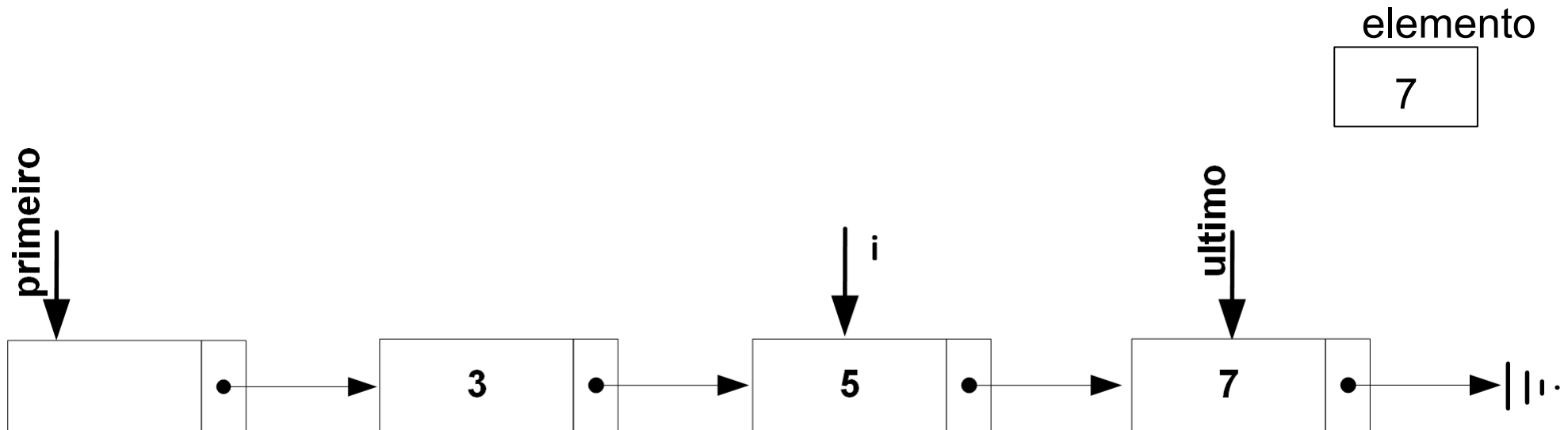
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

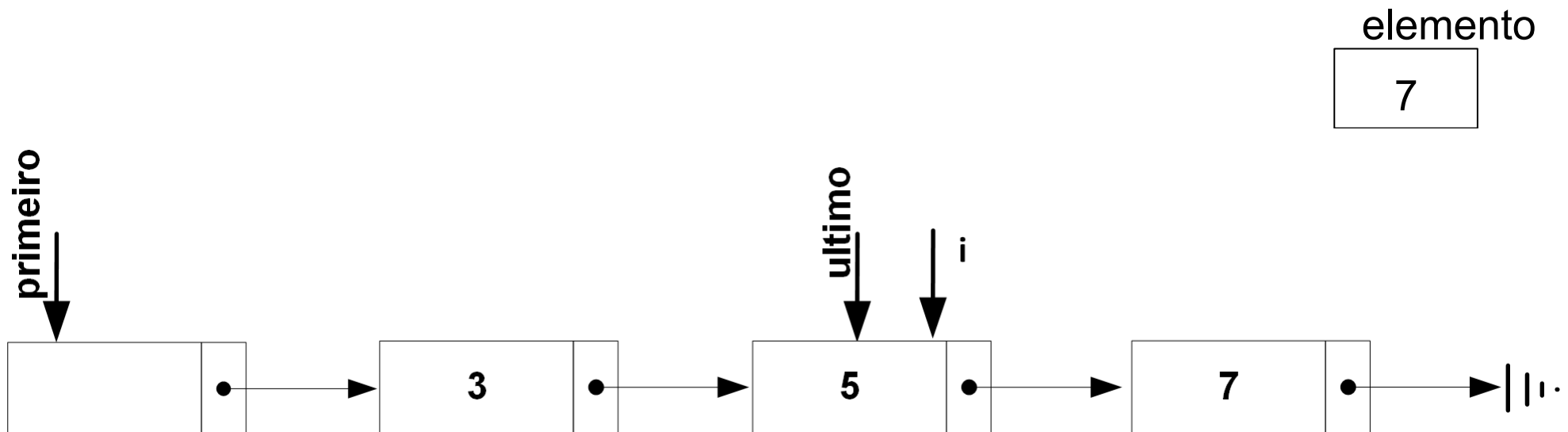
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

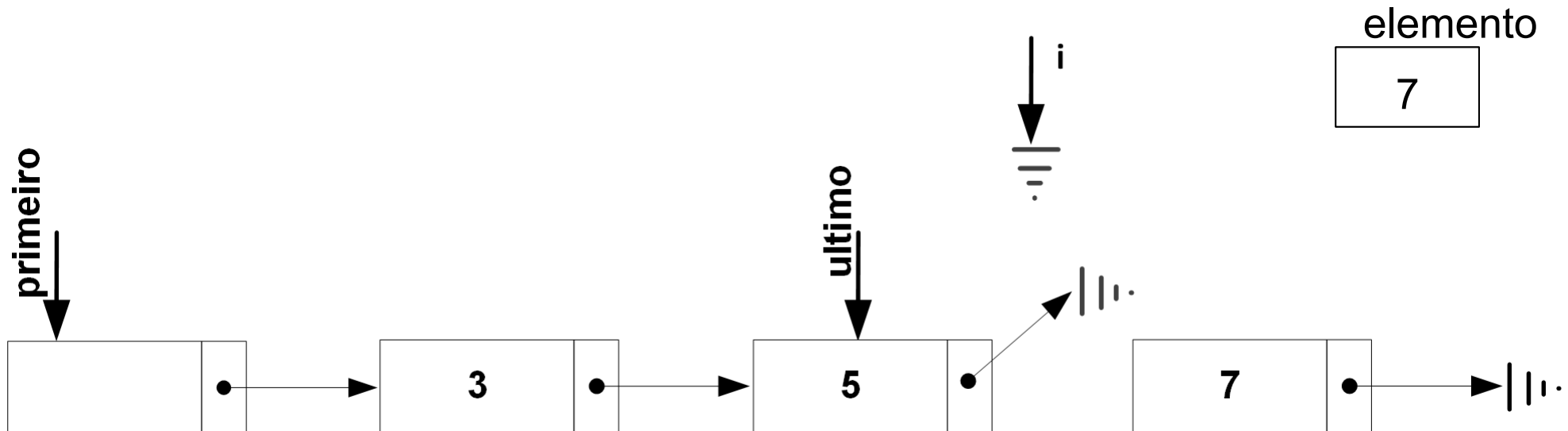
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

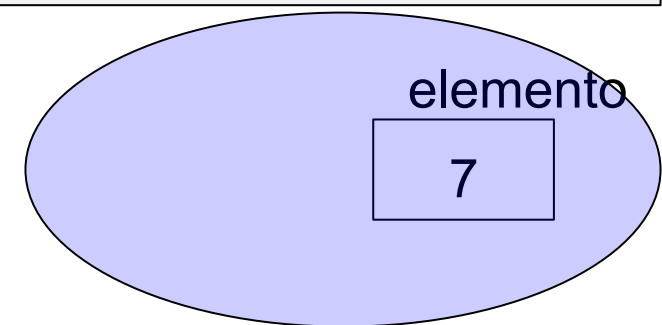
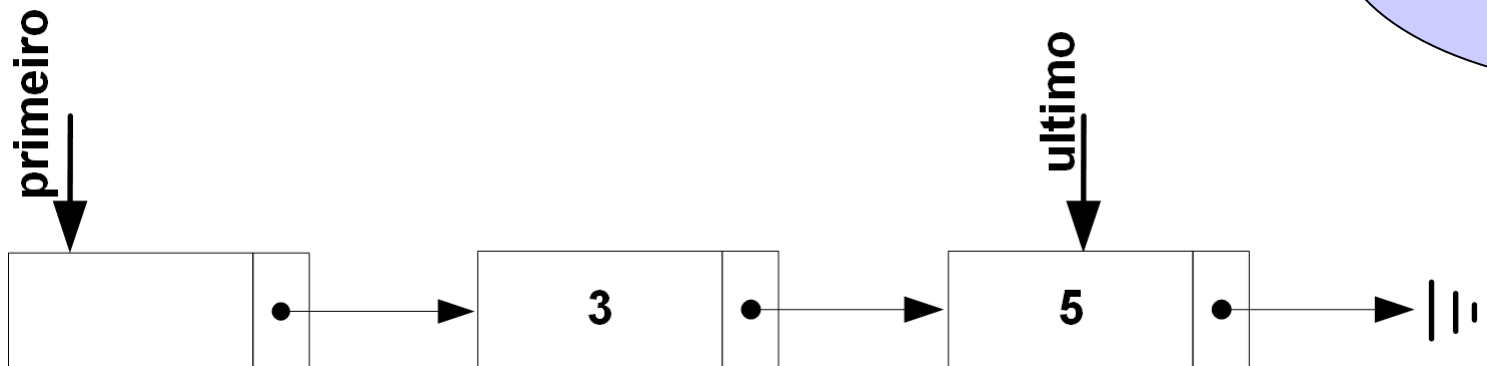
```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```



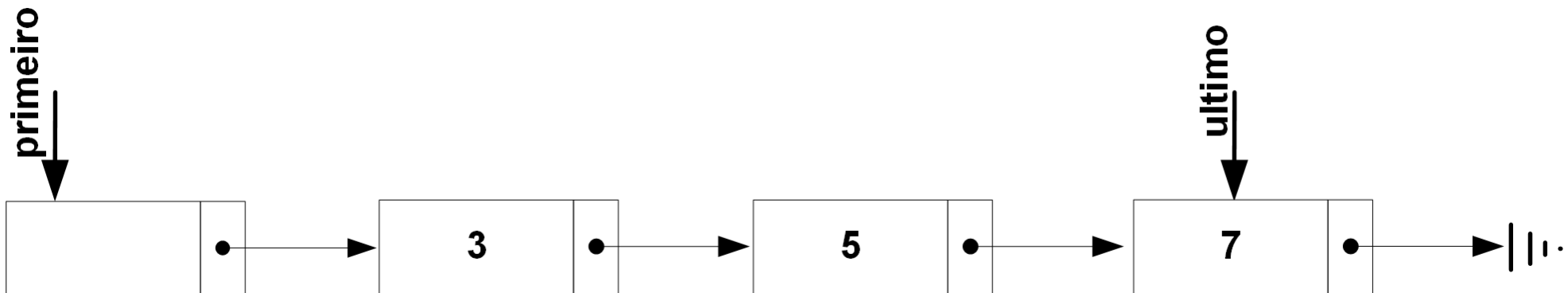
Remover no Fim

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

```
public int removerFim() throws Exception {  
    if (primeiro == ultimo)  
        throw new Exception("Erro!");  
    Celula i;  
    for(i = primeiro; i.prox != ultimo; i = i.prox);  
    int elemento = ultimo.elemento;  
    ultimo = i;  
    i = ultimo.prox = null;  
    return elemento;  
}
```

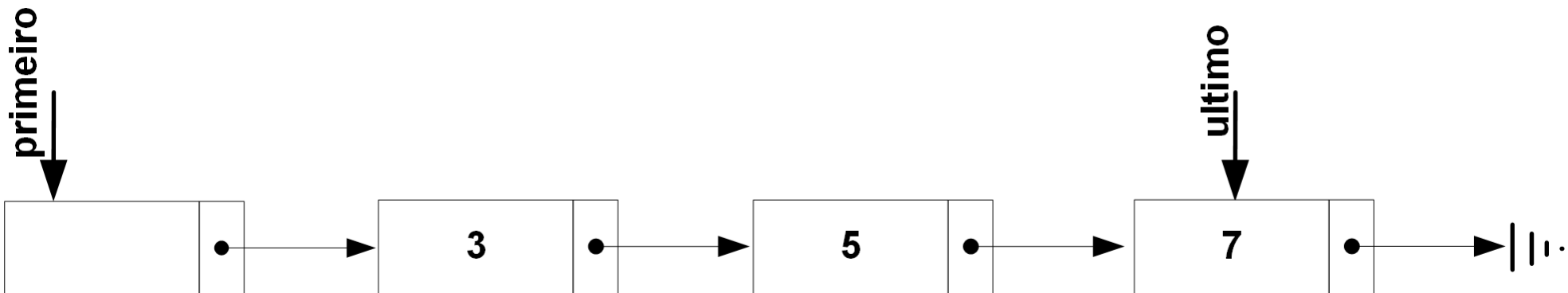


```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```



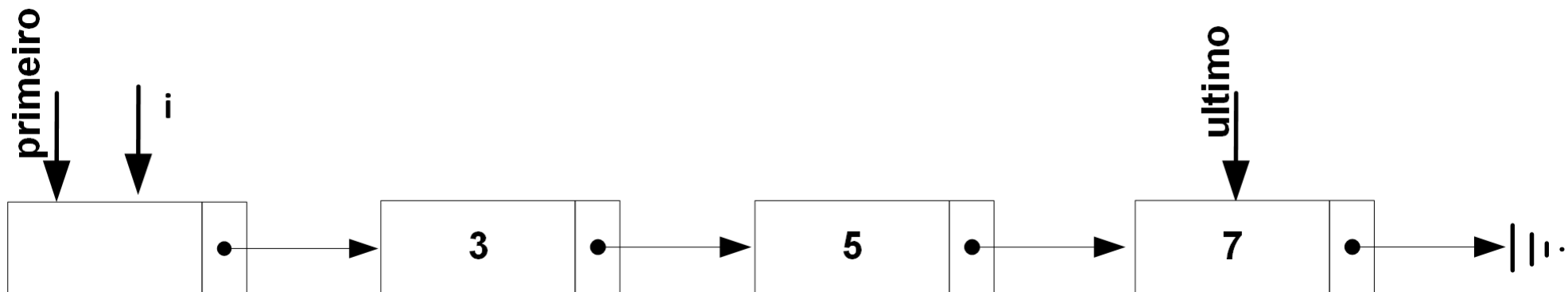
Inserir

```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    }
}
```



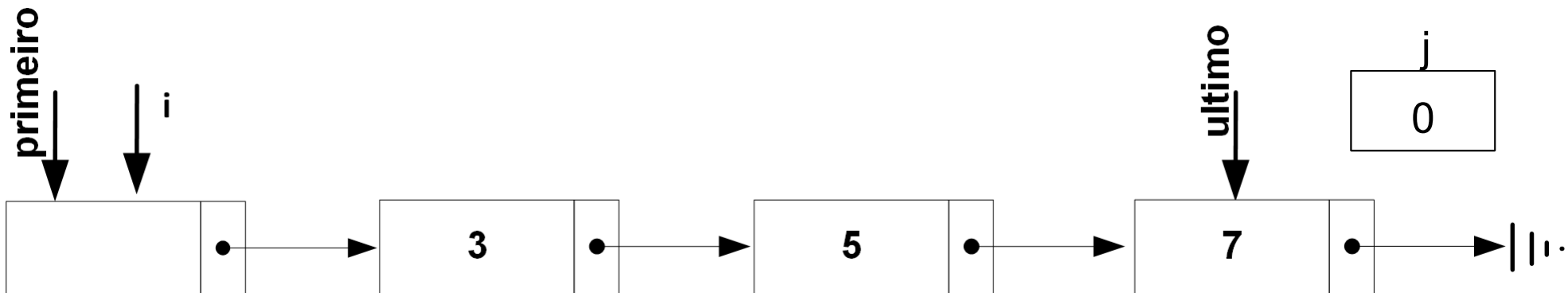
Inserir

```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    } }
}
```



Inserir

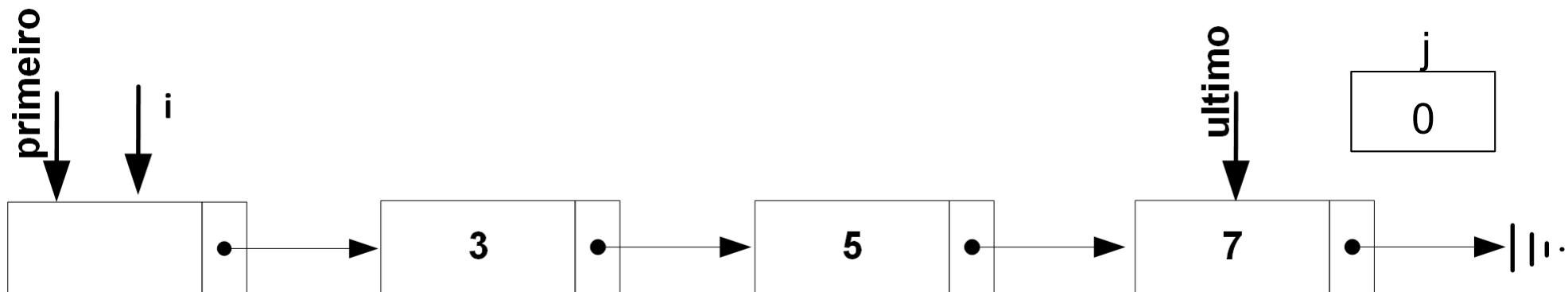
```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    }
}
```



Inserir

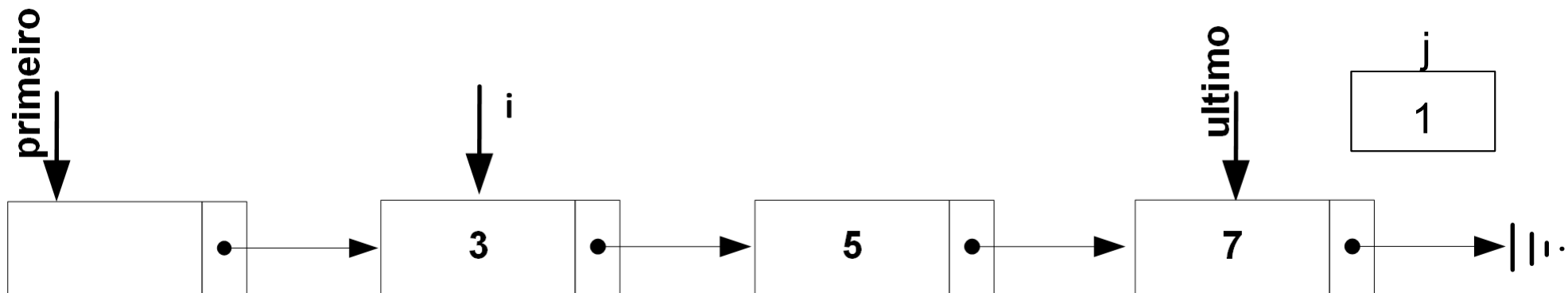
```
public void inserir(int x, int pos) throws Exception { //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho) { throw new Exception("Erro!");
    } else if (pos == 0) { inserirInicio(x);
    } else if (pos == tamanho) { inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    }
}
```

true



Inserir

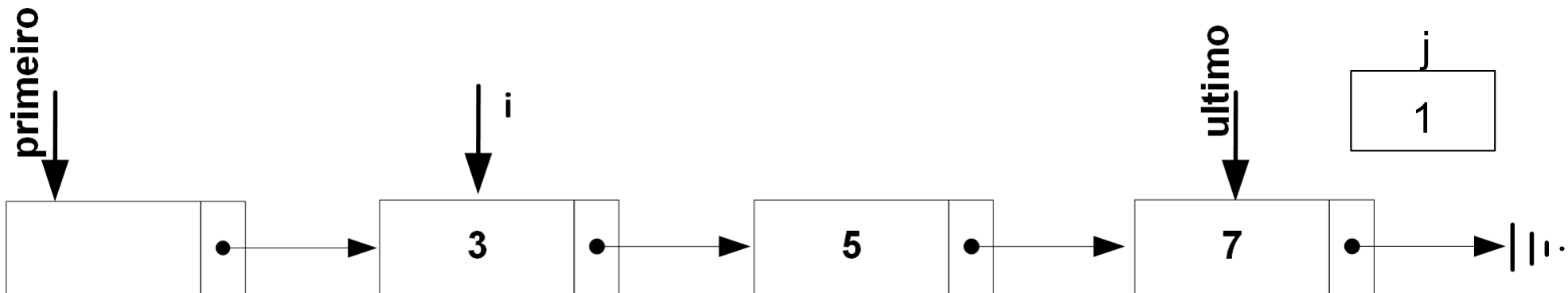
```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    }
}
```



Inserir

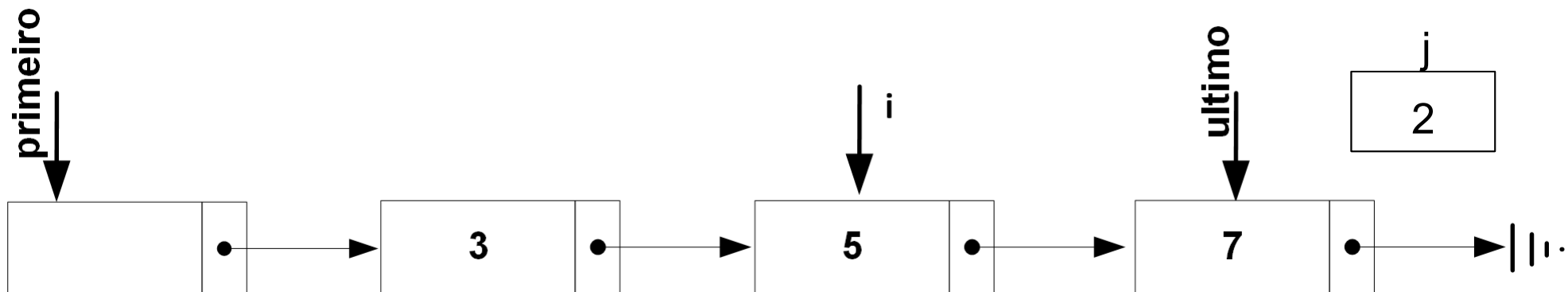
```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){   inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    }
}
```

true



Inserir

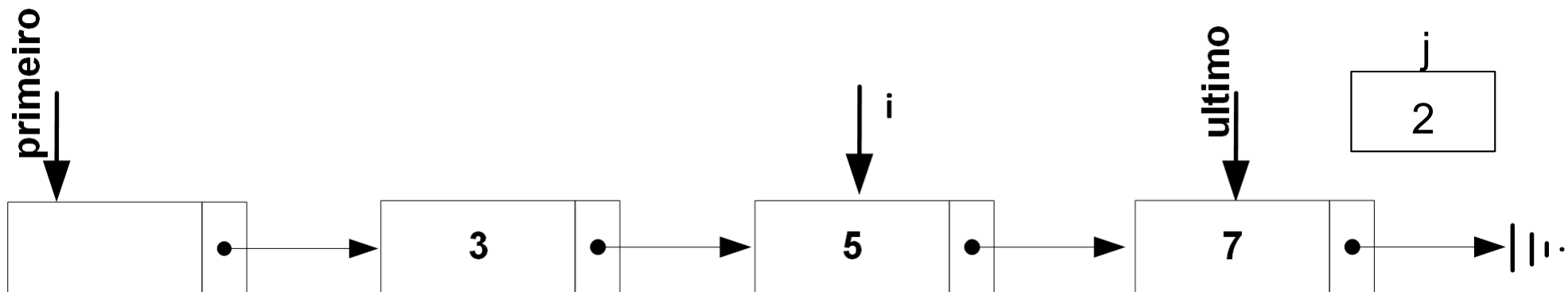
```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    }
}
```



Inserir

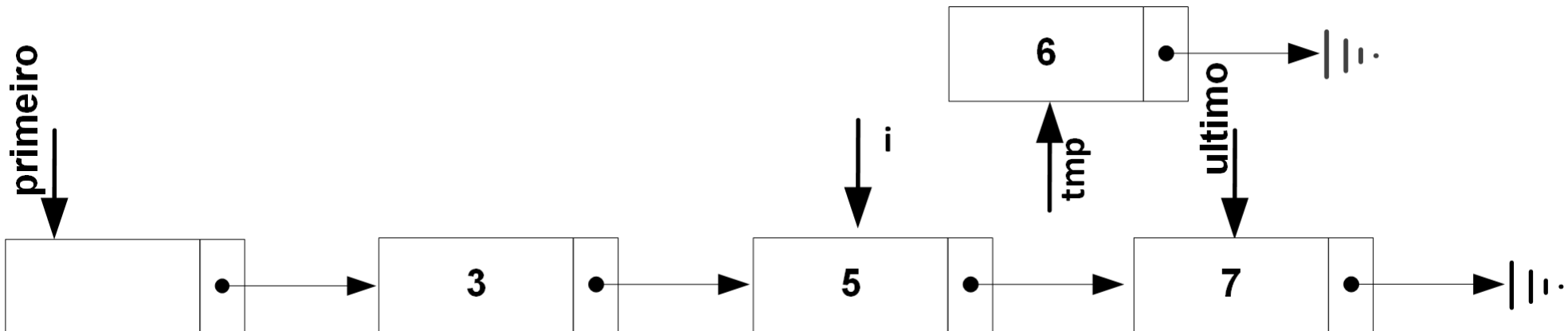
```
public void inserir(int x, int pos) throws Exception { //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho) { throw new Exception("Erro!");
    } else if (pos == 0) { inserirInicio(x);
    } else if (pos == tamanho) { inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    }
}
```

false



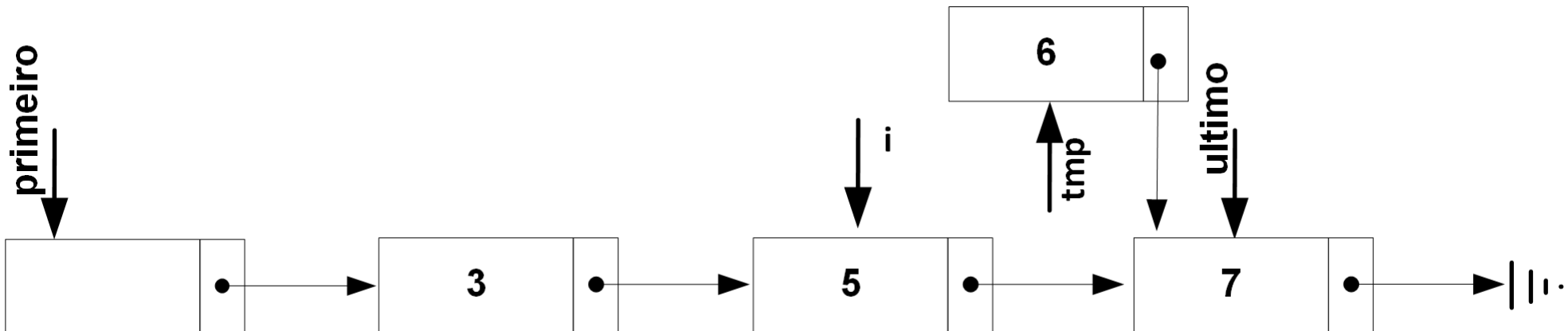
Inserir

```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    } }
}
```



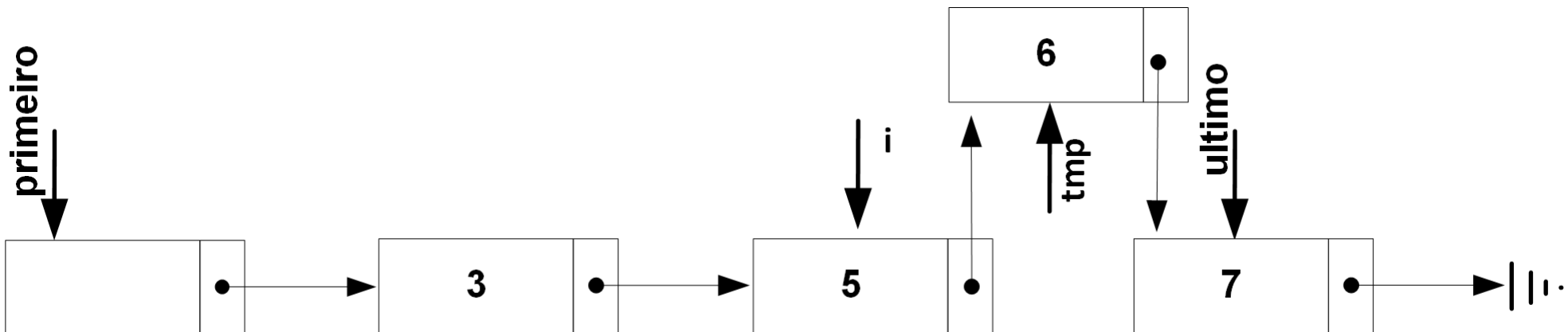
Inserir

```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    } }
}
```



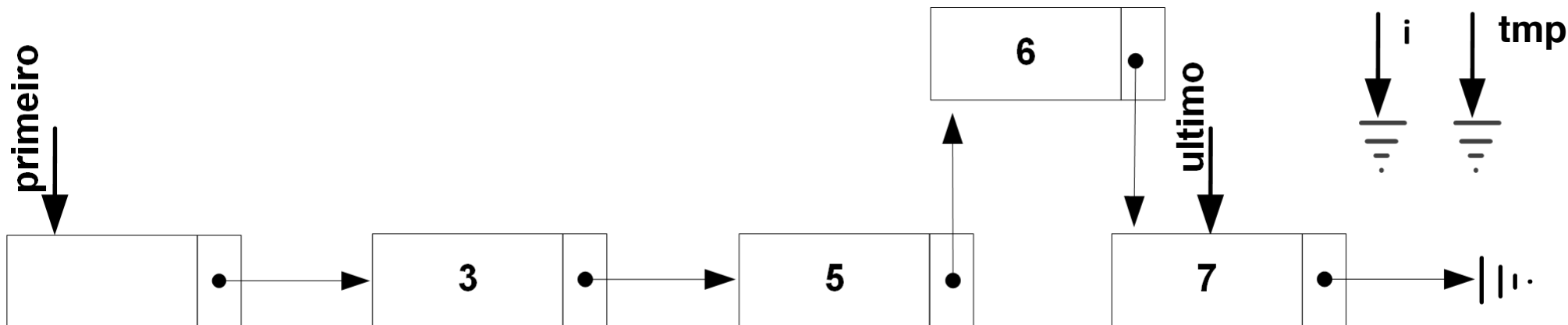
Inserir

```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    } }
}
```



Inserir

```
public void inserir(int x, int pos) throws Exception {    //Inserir(6, 2)
    int tamanho = tamanho();
    if (pos < 0 || pos > tamanho){ throw new Exception("Erro!");
    } else if (pos == 0){          inserirInicio(x);
    } else if (pos == tamanho){    inserirFim(x);
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = new Celula(x);
        tmp.prox = i.prox;
        i.prox = tmp;
        tmp = i = null;
    }
}
```

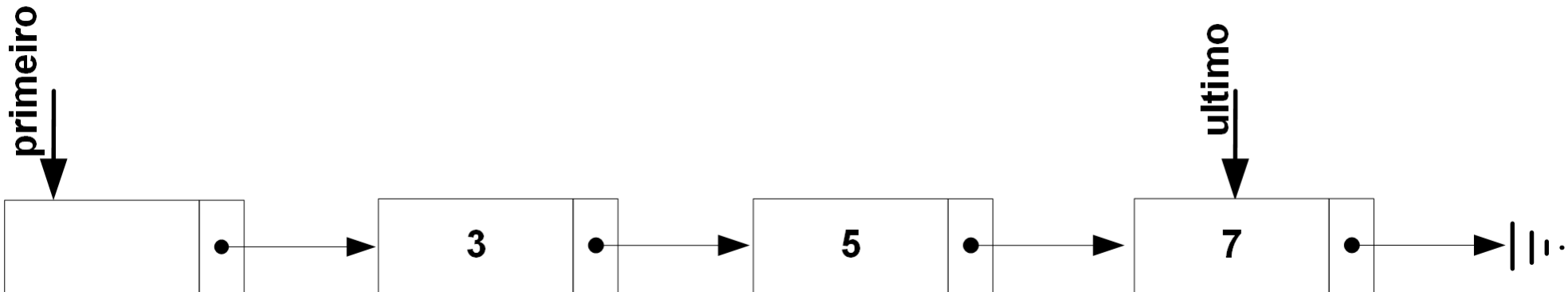


Remover

```
class Lista {  
    ...  
    public void inserirInicio(int x) { ... }  
    public int removerFim() { ... }  
    public void inserir(int x, int pos) { ... }  
    public int remover(int pos) { ... }  
}
```

Remover

```
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                       elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
```



Remover

```
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                        elemento = removerFim();
    } else {
```

```
    Celula i = primeiro;
```

```
    for(int j = 0; j < pos; j++, i = i.prox);
```

```
    Celula tmp = i.prox;
```

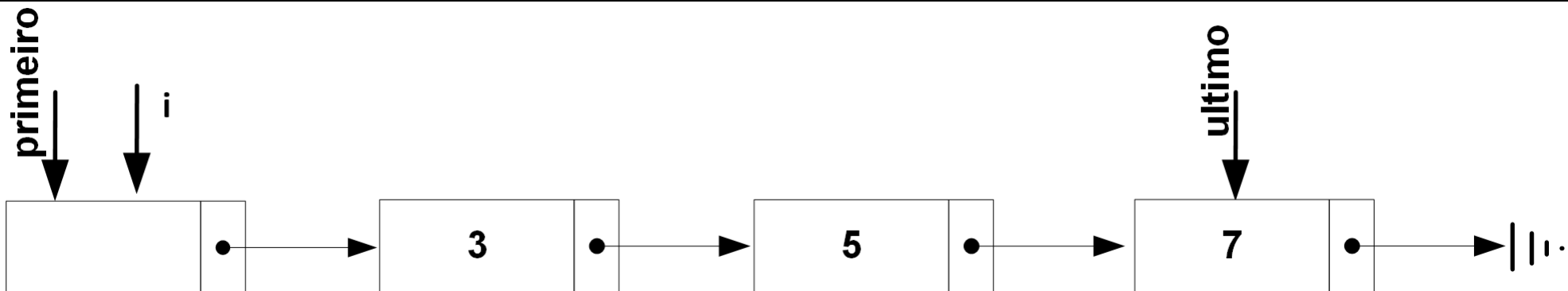
```
    elemento = tmp.elemento;    i.prox = tmp.prox;
```

```
    tmp.prox = null;           i = tmp = null;
```

```
}
```

```
return elemento;
```

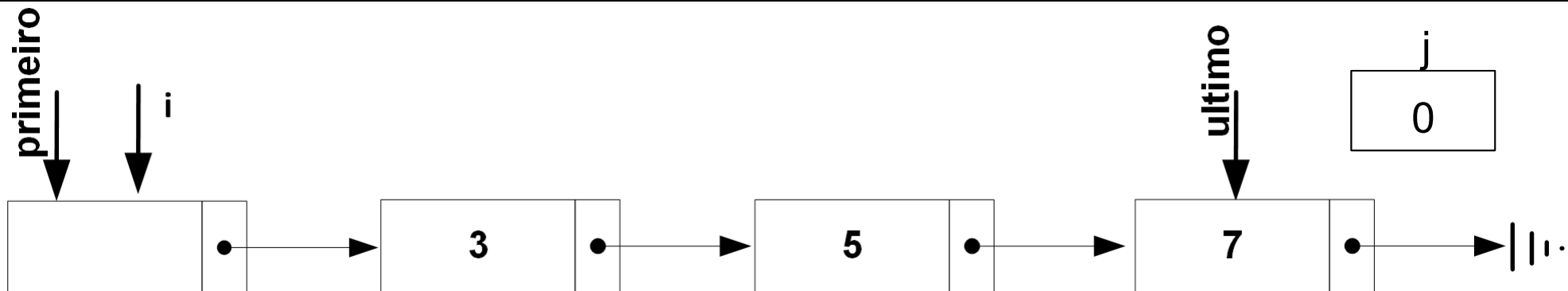
```
}
```



Remover

```

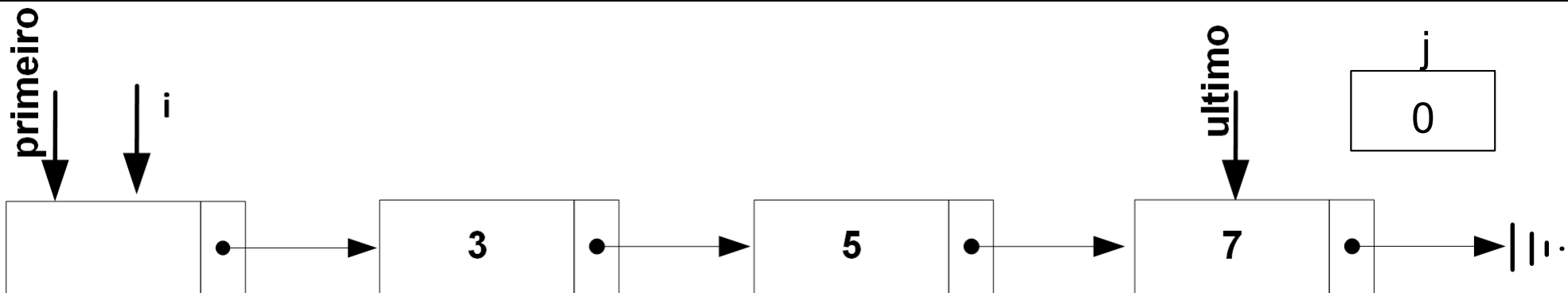
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                       elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
    
```



Remover

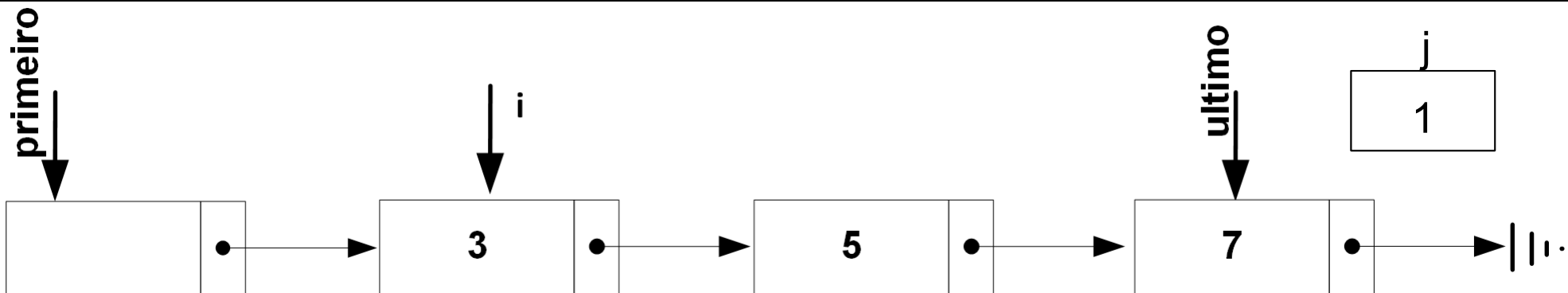
```

public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) { elemento = removerInicio();
    } else if (pos == tamanho - 1){ elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
    
```



Remover

```
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) { elemento = removerInicio();
    } else if (pos == tamanho - 1){ elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
```

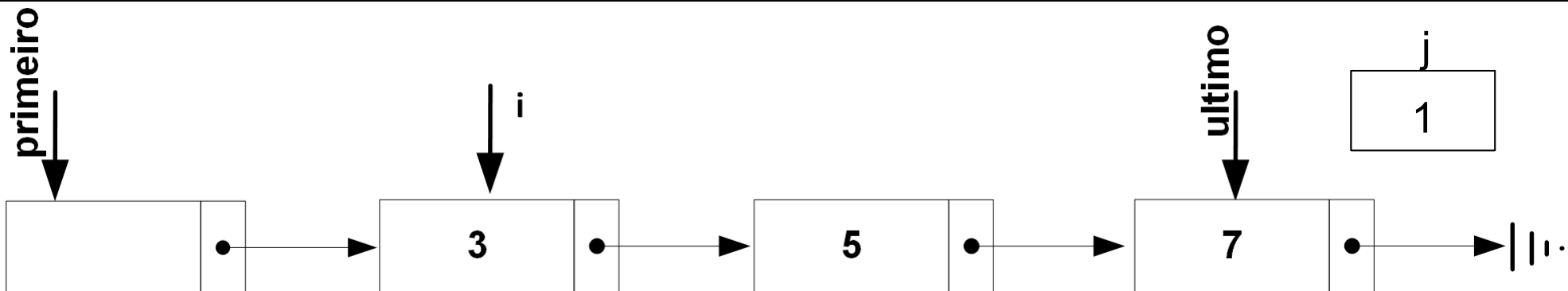


Remover

```

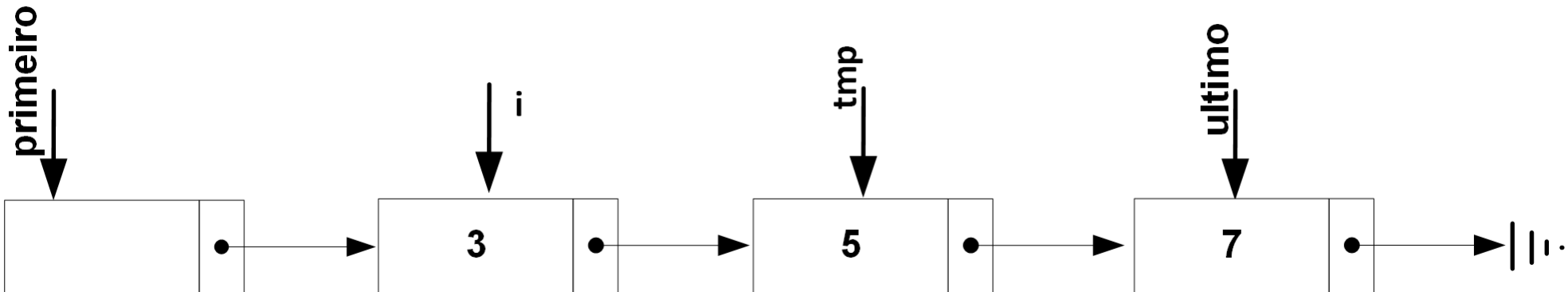
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                       elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
    
```

false



Remover

```
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) { elemento = removerInicio();
    } else if (pos == tamanho - 1){ elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
```

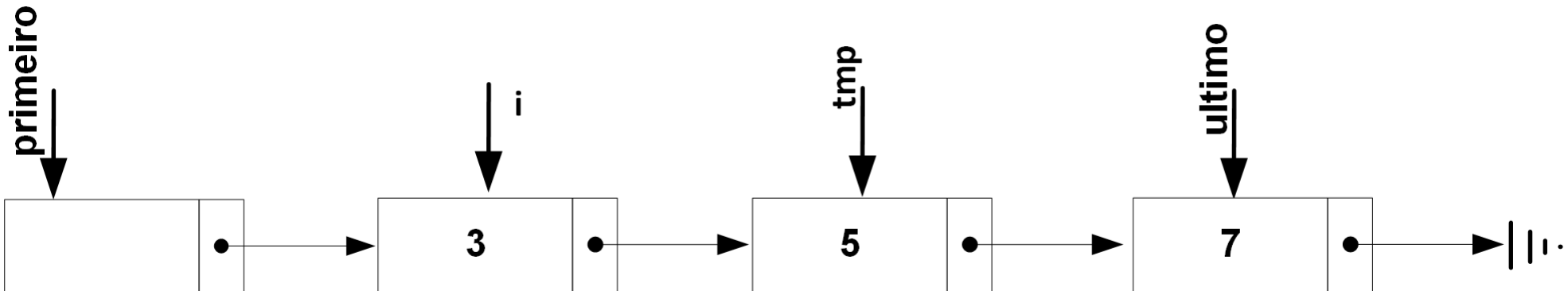


Remover

```
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                       elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
```

elemento

5

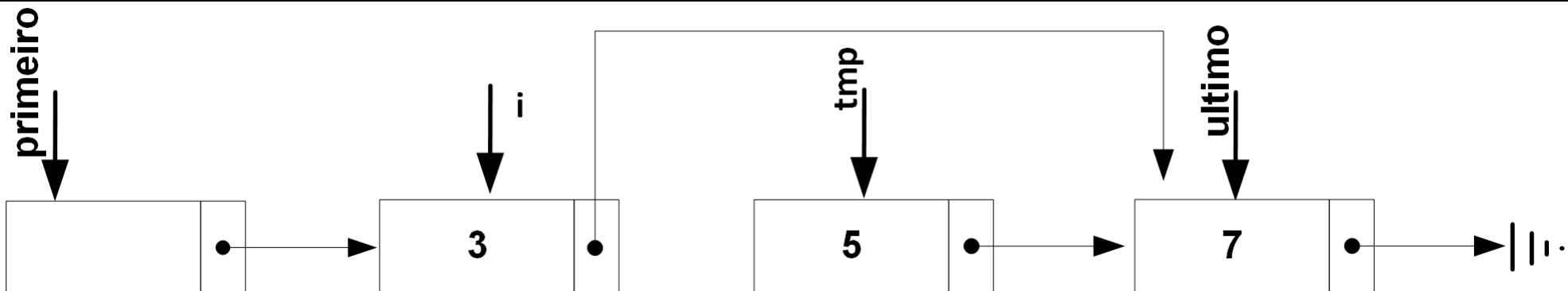


Remover

```
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                       elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento; i.prox = tmp.prox;
        tmp.prox = null; i = tmp = null;
    }
    return elemento;
}
```

elemento

5



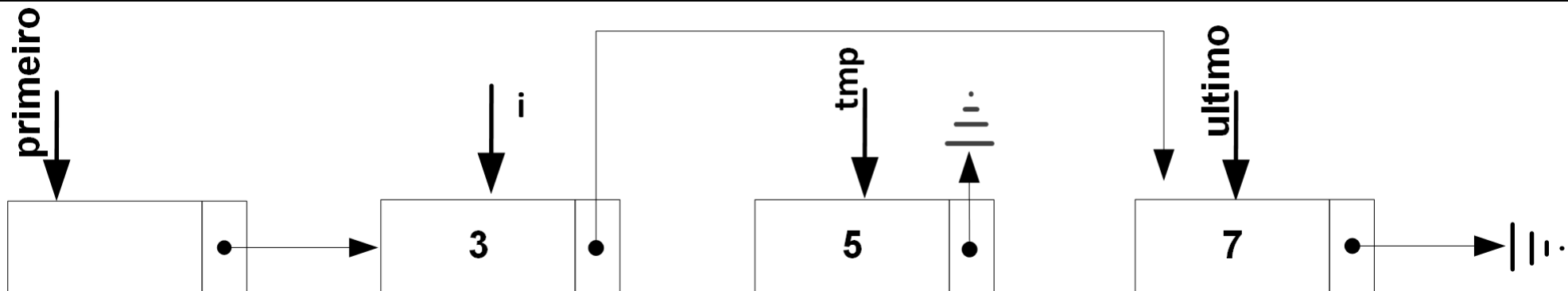
Remover

```

public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                       elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;           i = tmp = null;
    }
    return elemento;
}
    
```

elemento

5



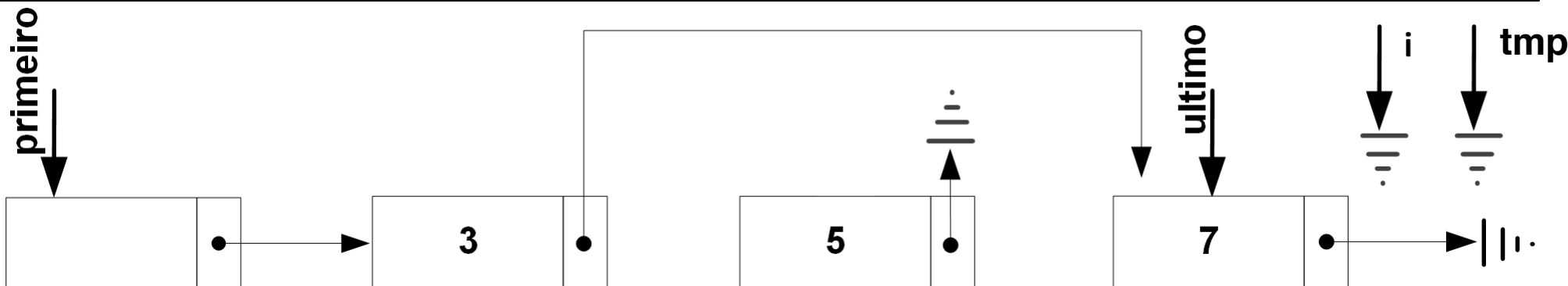
Remover

```

public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                       elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
    
```

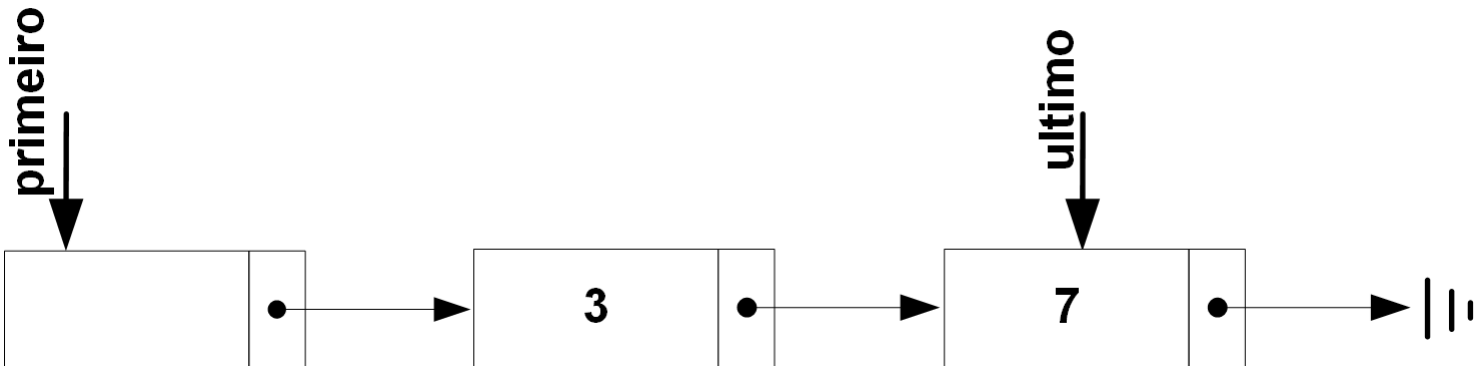
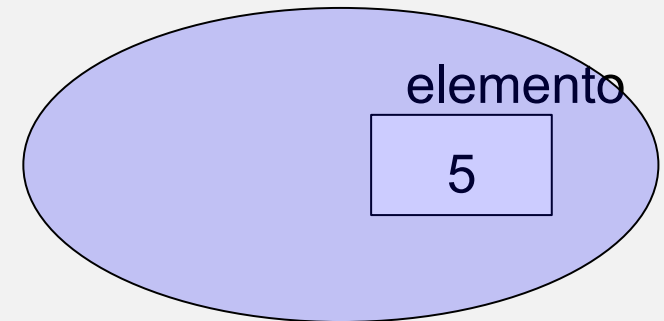
elemento

5



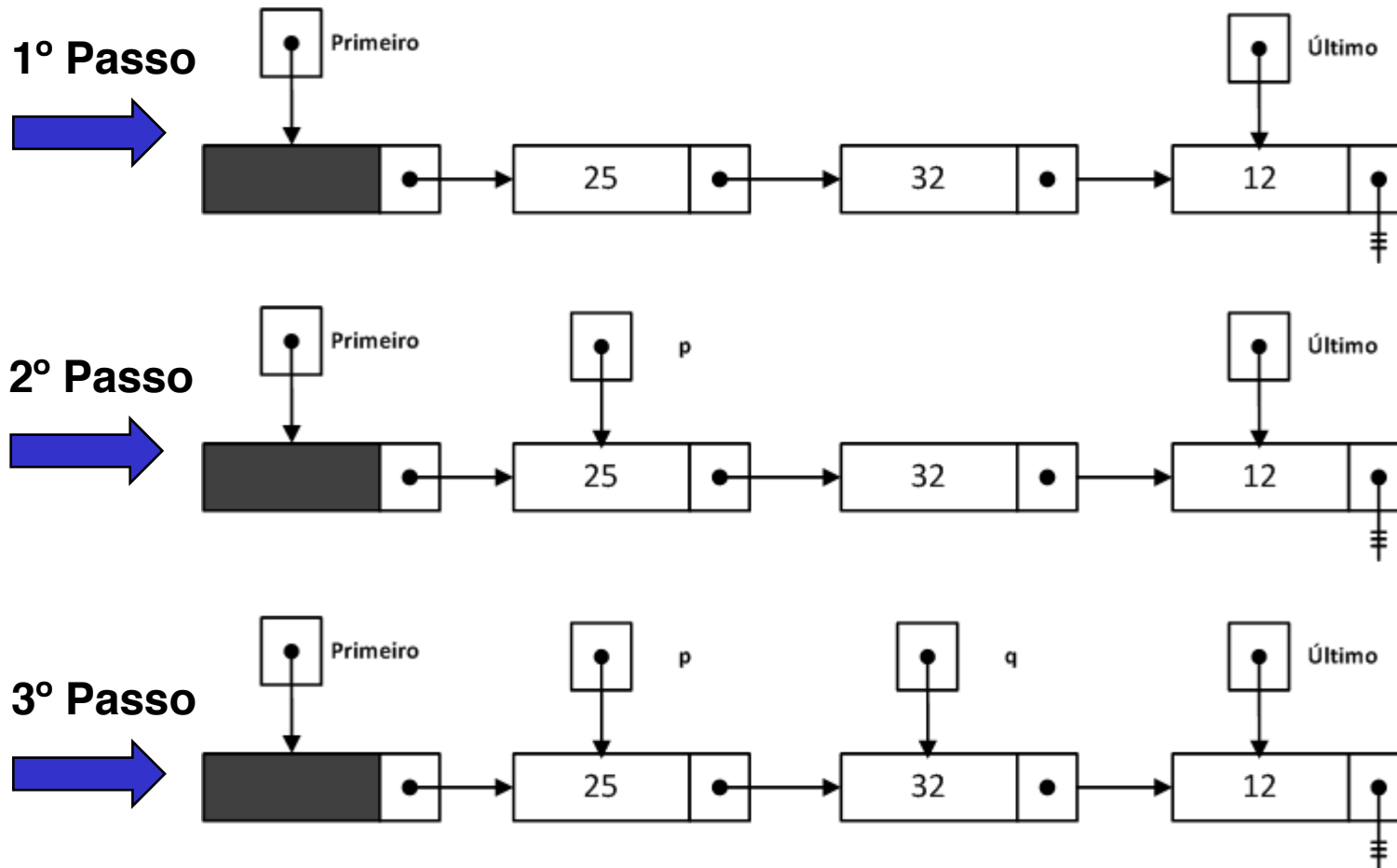
Remover

```
public int remover(int pos ) throws Exception {           //remover(1)
    int elemento, tamanho = tamanho();
    if (primeiro == ultimo || pos < 0 || pos >= tamanho){ throw new Exception("Erro!");
    } else if (pos == 0) {                                elemento = removerInicio();
    } else if (pos == tamanho - 1){                       elemento = removerFim();
    } else {
        Celula i = primeiro;
        for(int j = 0; j < pos; j++, i = i.prox);
        Celula tmp = i.prox;
        elemento = tmp.elemento;    i.prox = tmp.prox;
        tmp.prox = null;            i = tmp = null;
    }
    return elemento;
}
```



Exercício

- Seja nossa classe Lista, implemente um método que remove a segunda posição válida. Siga os passos da figura abaixo



Exercício

- Seja nossa classe Lista, implemente um método que remove a segunda posição válida. Siga os passos da figura abaixo

