

# Filas e Pilhas

# V e t o r e s

**Prof. Vinicius Ramos**

Adaptado de Prof. Cristian Cechinel





V  
i  
s  
ã  
o  
G  
e  
r  
a  
l

# Conteúdo

1

Filas

2

Pilhas





01

# Pilha x Fila

Quais são as suas características?



# PILHA

01

## LIFO

- Last-in-First-Out
- O **último** a entrar é o **primeiro** que sai

02

## Incluir

SEMPRE no **TOPO**

03

## Retirar

SEMPRE no **TOPO**



# FILA

01

## FIFO

- First-in-First-Out
- O **primeiro** que entra é o **primeiro** que sai

02

## Incluir

SEMPRE no **FINAL**

03

## Retirar

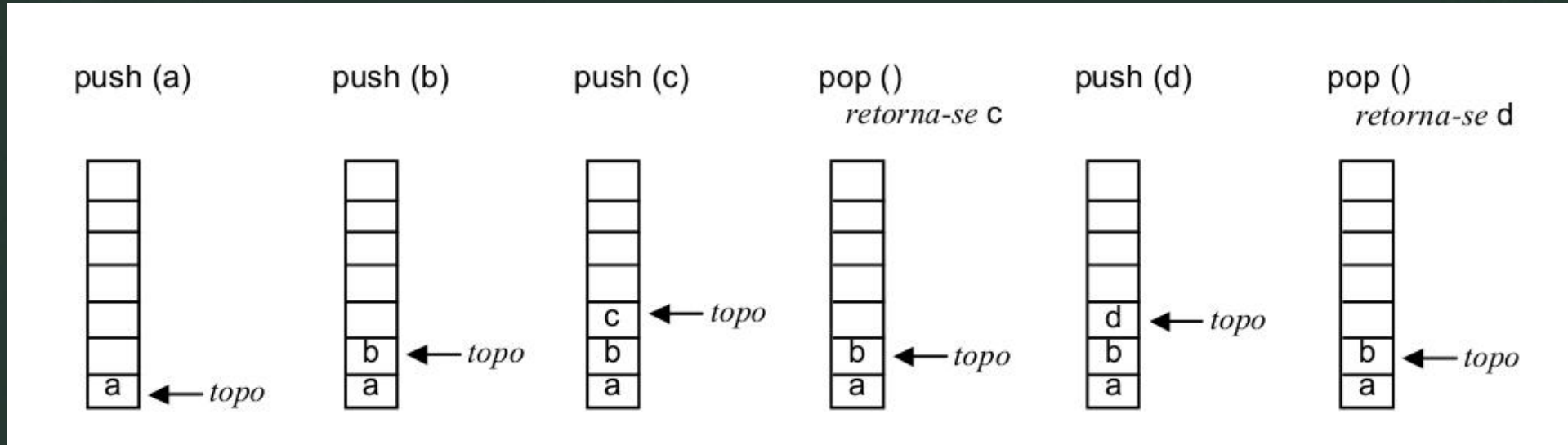
SEMPRE no **INÍCIO**



# PILHA

## Operações:

- **push()** - Empilha
- **pop()** - Desempilha



Extraído de (CELES, CERQUEIRA, e RANGEL, 2004)



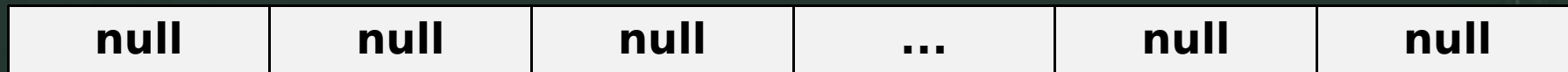
# FILA

Operações:

- `insert()` - Inserir
- `remove()` - Retirar

Início

Fim



Início

`insert(29);`



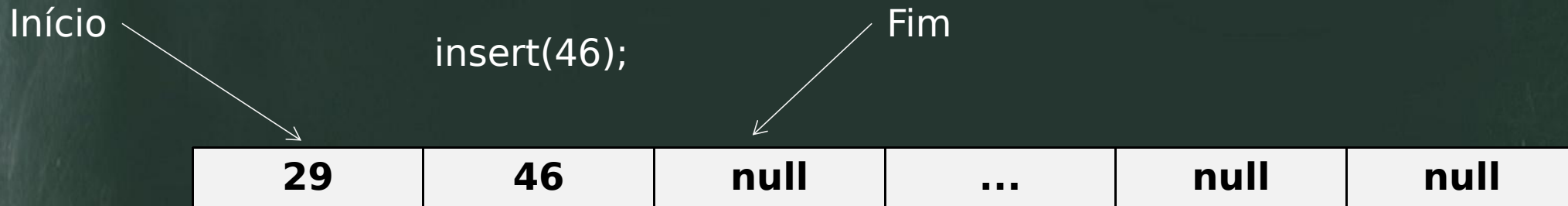
Fim



# FILA

Operações:

- `insert()` - Inserir
- `remove()` - Retirar



# Referências

- CELES, W., R. CERQUEIRA, and JL RANGEL. **Introdução a Estruturas de Dados**. Rio de Janeiro, RJ, Brasil: Campus, 2004. 294 p. ISBN 85-352-1228-0.
- [https://inf.ufes.br/~pdcosta/ensino/2012-2-estruturas-de-dados/slides/Aula9\(listas\).pdf](https://inf.ufes.br/~pdcosta/ensino/2012-2-estruturas-de-dados/slides/Aula9(listas).pdf)
- Sedgwick, Robert, and Kevin Wayne. **Algorithms**. Addison-Wesley Professional, 2011.



OBRIGADO!



# P i l h a s V e t o r e s

**Prof. Vinicius Ramos**

Adaptado de Prof. Cristian Cechinel



# Pilha

## Atributos de uma Classe do tipo Pilha

- Quantidade de elementos na pilha
- Vetor para armazenar os elementos na pilha

```
13     public class PilhaInteiros {  
14         private Integer[] dados;  
15         private int qt;  
16     }  
17 }
```



# Pilha

- Criação de uma pilha vazia
- Inserir um elemento no topo (push, empilha)
- Remover um elemento do topo (pop, desempilha)
- Verificar se a pilha está vazia



# Pilha

1. Desenvolva os métodos da classe **Pilha** e uma outra classe para utilizar a pilha.
2. Os elementos da pilha serão armazenados em uma lista representada por um vetor.



# Pilha

- Criação de uma pilha vazia

```
16  
17 public PilhaInteiros(int tamanho){  
18     dados = new Integer[tamanho];  
19     qt = 0;  
20 }
```



# Pilha

- Inserir um elemento no topo (push, empilha)

```
24  □ void empilha(int valor){  
25      if (qt<dados.length){  
26          dados[qt] = valor;  
27          qt++;  
28      }  
29  }
```



# Pilha

- **Remover um elemento do topo (pop, desempilha)**

```
30 Integer desempilha(){
31     if (qt != 0){
32         qt--;
33         return dados[qt];
34     }
35     return null;
36 }
```



# Pilha

- Verificar se a pilha está vazia

```
21  □    boolean ehVazia(){  
22  |        return qt == 0;  
23  |    }  
    |
```



# Referências

- CELES, W., R. CERQUEIRA, and JL RANGEL. **Introdução a Estruturas de Dados**. Rio de Janeiro, RJ, Brasil: Campus, 2004. 294 p. ISBN 85-352-1228-0.
- [https://inf.ufes.br/~pdcosta/ensino/2012-2-estruturas-de-dados/slides/Aula9\(listas\).pdf](https://inf.ufes.br/~pdcosta/ensino/2012-2-estruturas-de-dados/slides/Aula9(listas).pdf)
- Sedgewick, Robert, and Kevin Wayne. **Algorithms**. Addison-Wesley Professional, 2011.



OBRIGADO!



# F i l a s V e t o r e s

**Prof. Vinicius Ramos**

Adaptado de Prof. Cristian Cechinel



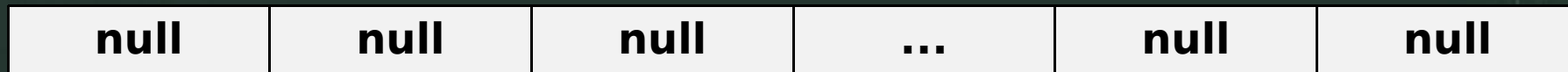
# FILA

Operações:

- `insert()` - Inserir
- `remove()` - Retirar

Início

Fim



Início

`insert(29);`



Fim



# FILA

Operações:

- `insert()` - Inserir
- `remove()` - Retirar



# FILA

## Atributos de uma Classe do tipo Fila

- Quantidade de elementos na fila
- Vetor para armazenar os elementos na fila
- Índice do primeiro elemento da Fila



# Referências

- CELES, W., R. CERQUEIRA, and JL RANGEL. **Introdução a Estruturas de Dados**. Rio de Janeiro, RJ, Brasil: Campus, 2004. 294 p. ISBN 85-352-1228-0.
- [https://inf.ufes.br/~pdcosta/ensino/2012-2-estruturas-de-dados/slides/Aula9\(listas\).pdf](https://inf.ufes.br/~pdcosta/ensino/2012-2-estruturas-de-dados/slides/Aula9(listas).pdf)
- Sedgwick, Robert, and Kevin Wayne. **Algorithms**. Addison-Wesley Professional, 2011.



OBRIGADO!

