

# FILA

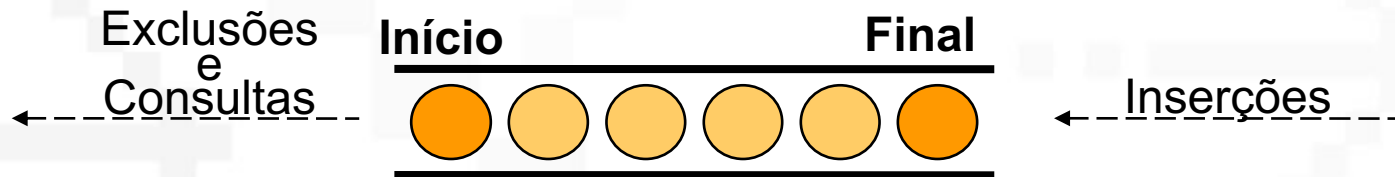
Prof. José Luiz de Freitas Júnior

Listas lineares especiais mais usuais: **Fila**

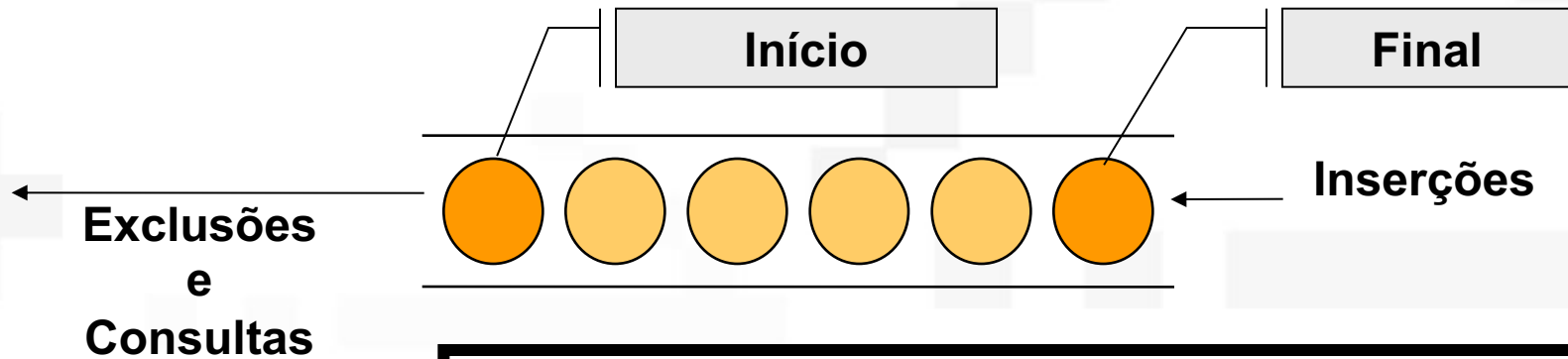
**FIFO** *First In First Out*

**o primeiro componente inserido  
é também o primeiro a ser retirado**

# FILA



# Filas



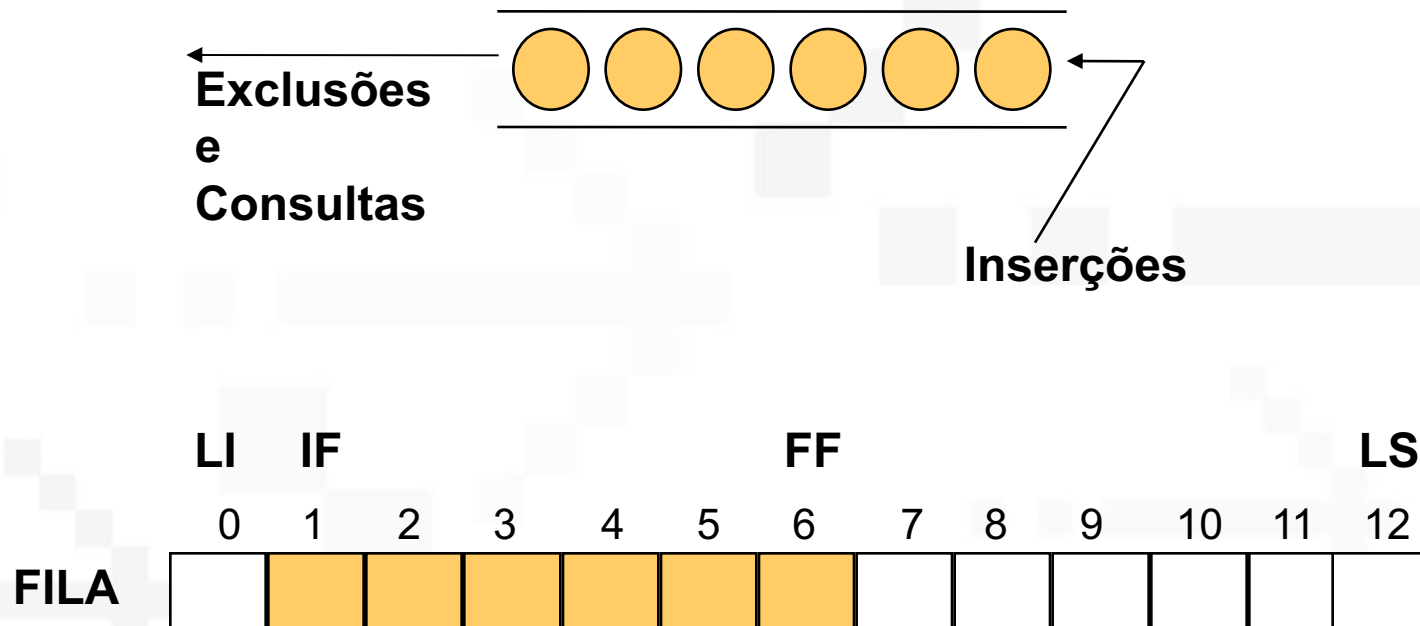
## Operações válidas:

- Criar uma fila vazia
- Inserir um elemento no final da fila
- Remover um elemento do início da fila
- Acessar o elemento do início da fila
- Verificar se a fila esta vazia
- Obter o número de elementos da fila
- Destruir a fila

# Filas

## Filas implementadas por contiguidade física - Vetor

# Fila implementada sobre arranjo



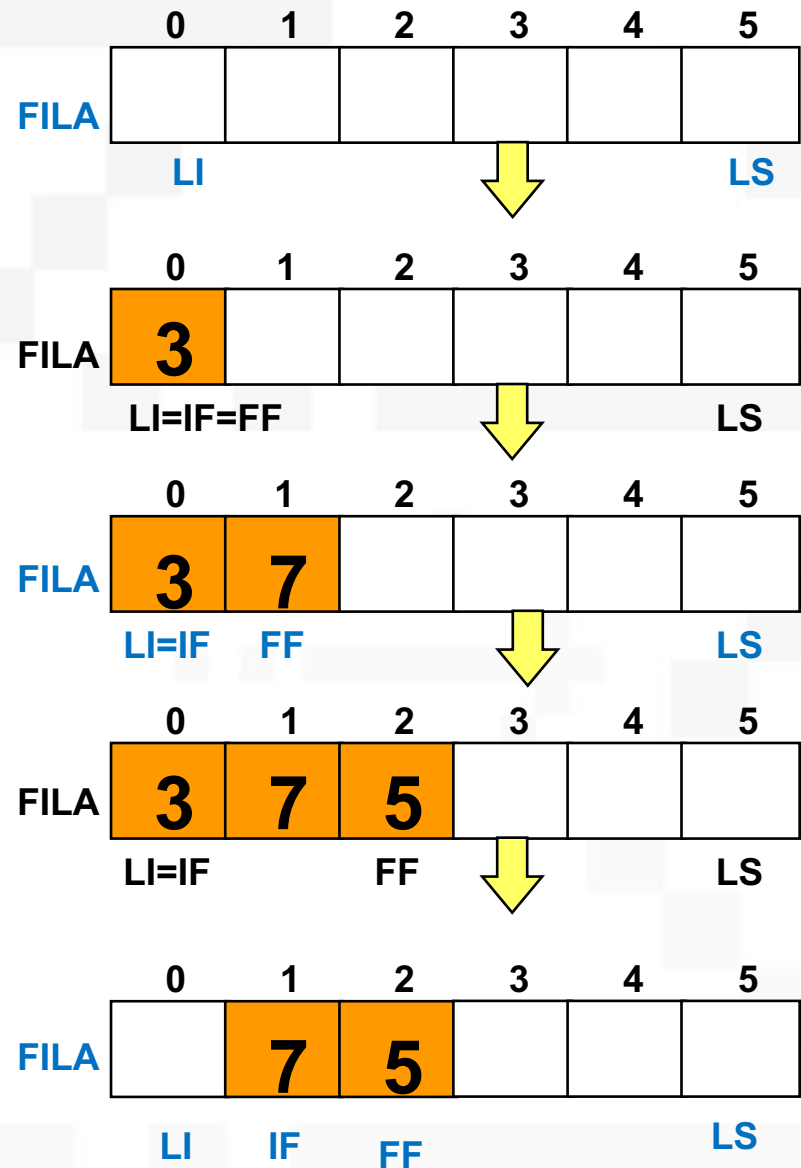
LI : limite inferior da área  
IF : início da fila  
FF : final da fila  
LS : limite superior da área

Fila vazia

quantidade = 0

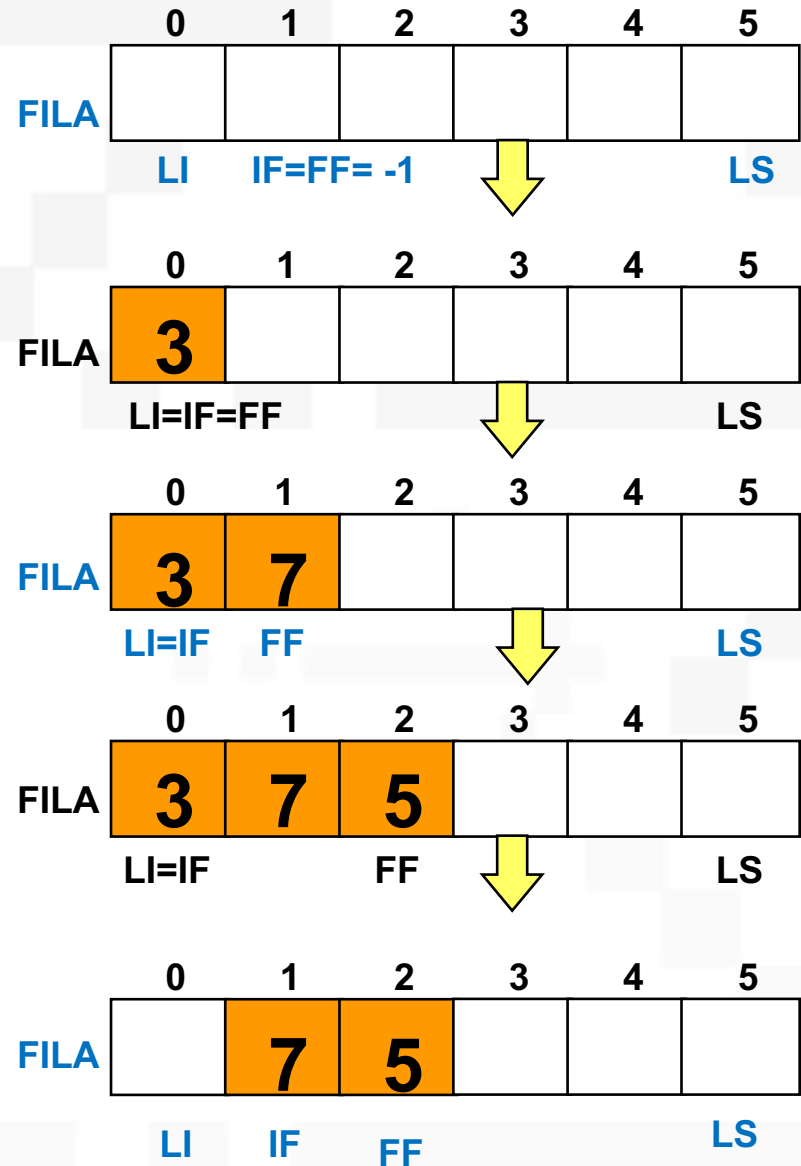
# Evolução da Fila

1. Inicializar a fila
2. Inserir um novo nodo com valor 3
3. Inserir um novo nodo com valor 7
4. Inserir um novo nodo com valor 5
5. Remover um nodo



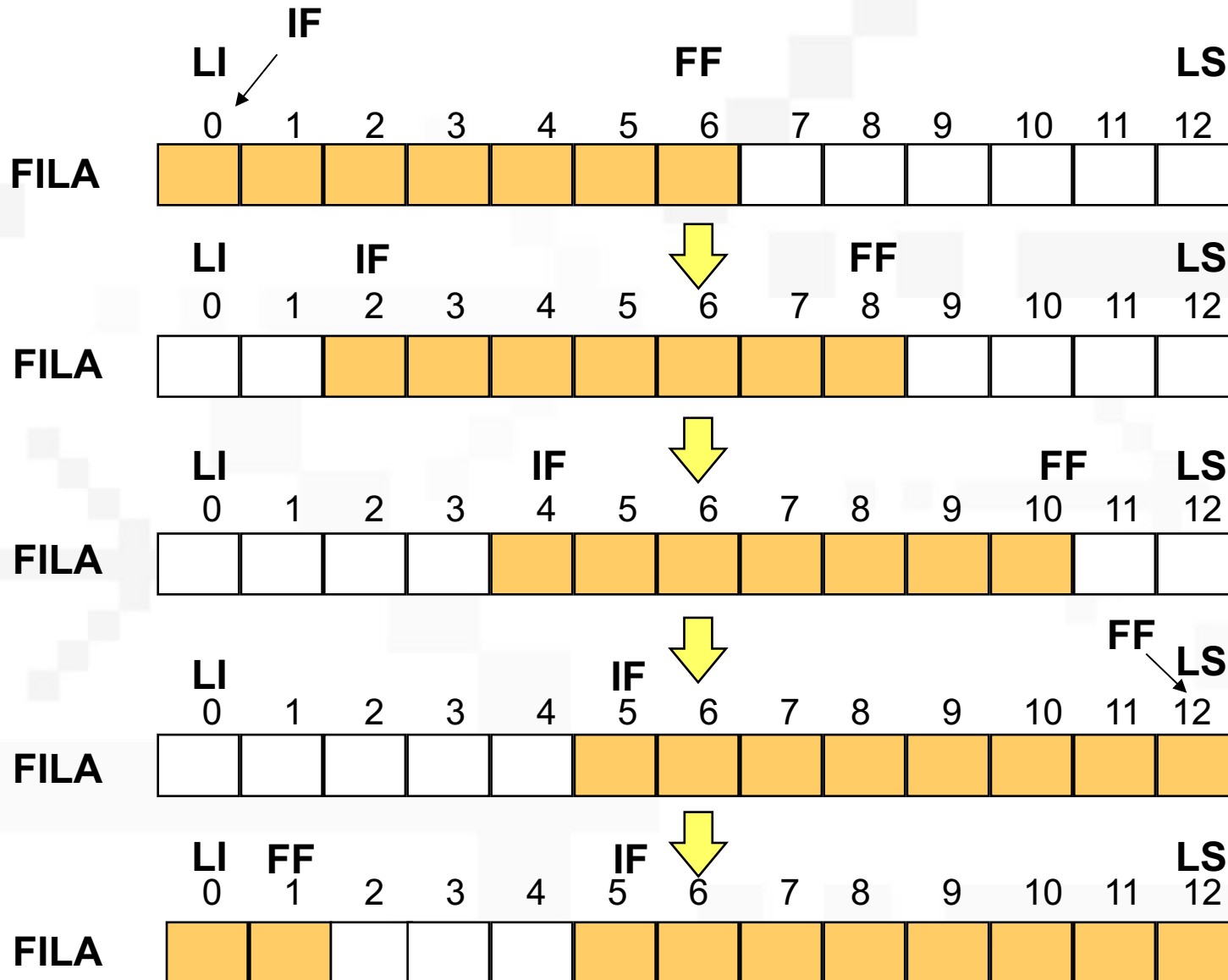
Qual problema deste modelo de solução?

Como resolver?





# Ocupação circular do arranjo



# Operações sobre Filas implementadas por contiguidade - Vetor

- Criar uma fila vazia
- Inserir um elemento no final da fila
- Remover um elemento do início da fila
- Acessar o elemento do início da fila
- Verificar se a fila esta vazia
- Obter o número de elementos da fila
- Destruir a fila

**Tipo de dados** utilizado nos algoritmos para fila implementada por contiguidade física:

```
TipoFila = arranjo [1..N] de TipoNo
```

# Operações sobre Filas

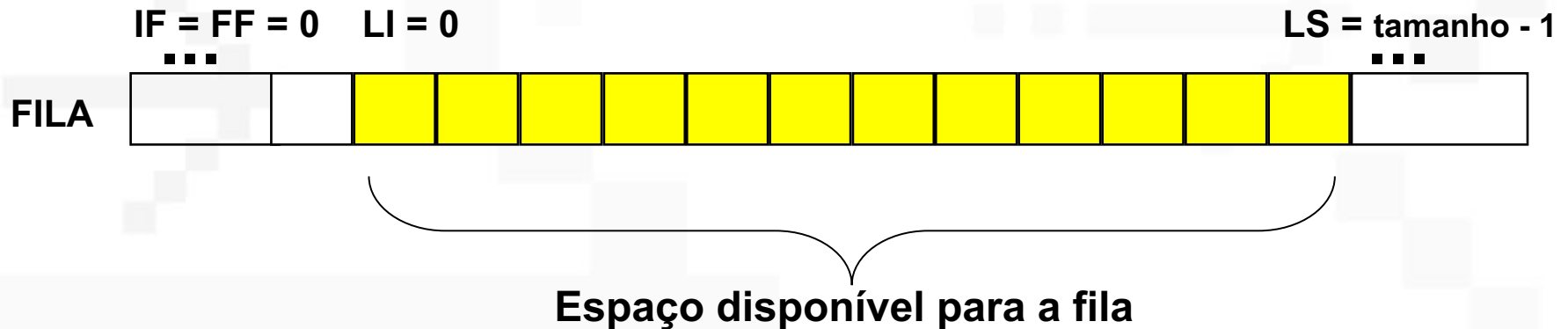
## implementadas por contiguidade - Vetor

### Fila

- tamanho : int = 0 {valor > 0}
  - array [tamanho] : int
  - inicio : int = 0
  - fim : int = 0
  - quantidadeDeElementos : int = 0
- 
- + Fila( in tamanho : int)
  - + ~Fila()
  - + enqueue( in elemento : int) : void {insere um elemento na fila}
  - + dequeue() : void {remover o elemento da fila}
  - + peek() : int {acessar o elemento do início da fila}
  - + size() : int {retorna o tamanho da fila}
  - + empty() : boolean {retorna se a fila esta vazia}
  - + destroy() : void

# Criação de uma fila

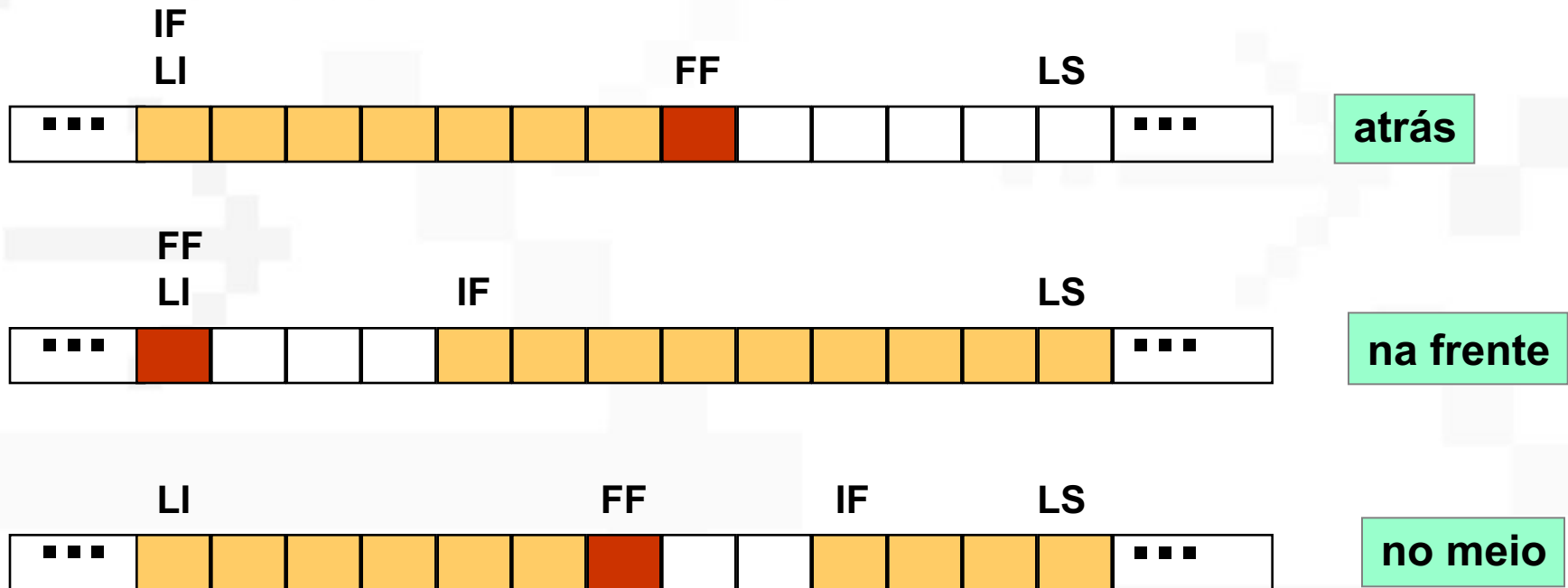
- Inicializar variáveis que controlam início e final da fila e início e final da área disponível para a fila



# Inserção de um elemento em uma fila

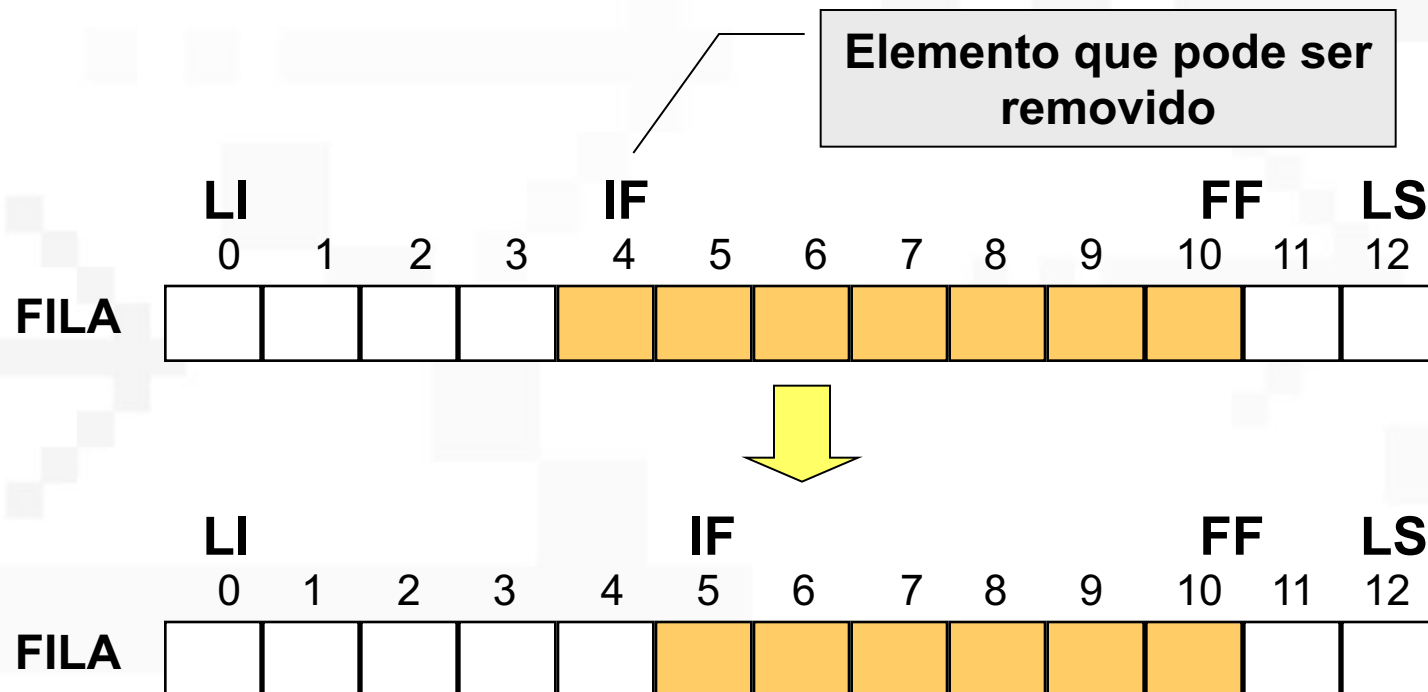
- Elemento inserido sempre no final da fila

Testar se tem espaço livre para inserir:



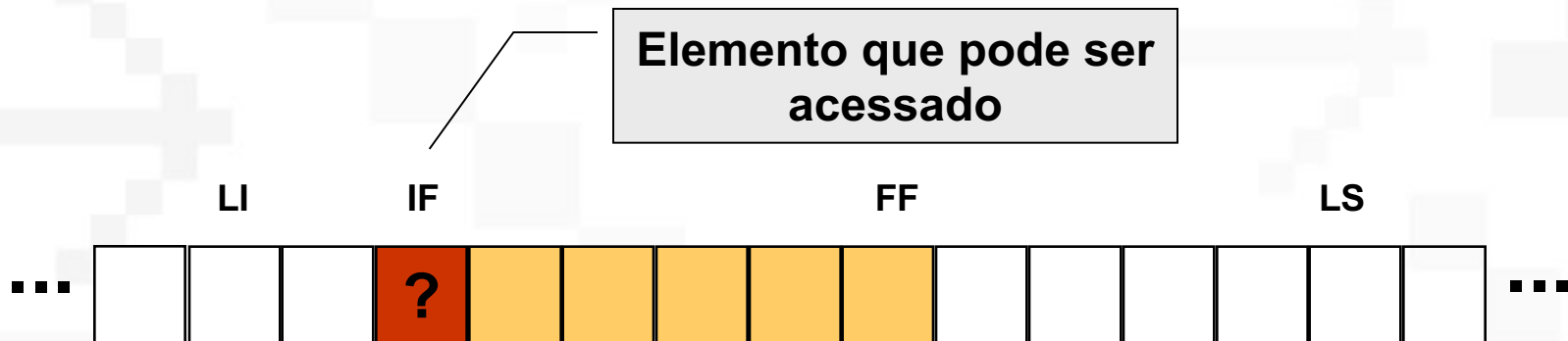
# Remoção de um elemento de uma fila

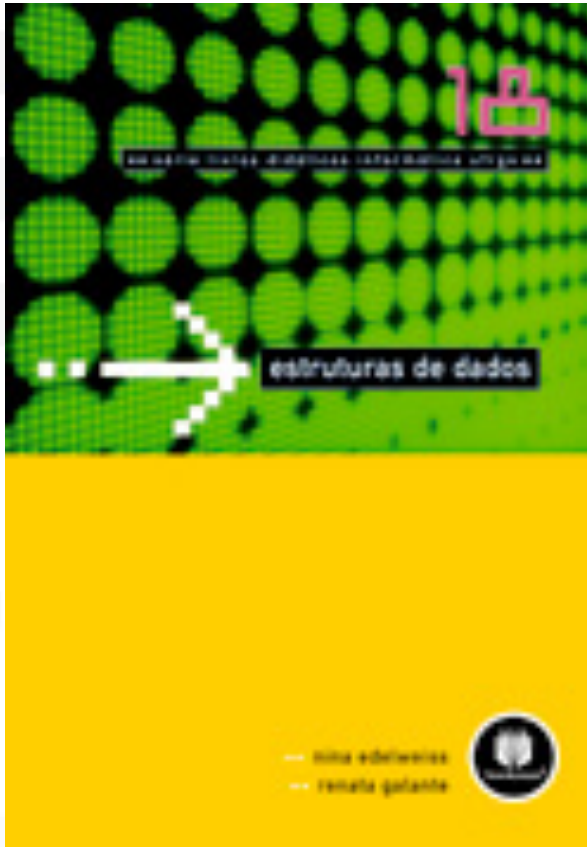
- Elemento removido é sempre o do início da fila



## Acesso à fila

- Só o elemento do início da fila pode ser acessado
- Acesso para consulta





EDELWEISS, Nina, GALANTE, Renata.  
**Estruturas de Dados.**  
Porto Alegre : Bookman, 2009.

Transparências do Livro