

Fila

Programação II – Engenharia de Telecomunicações

Prof. Emerson Ribeiro de Mello

mello@ifsc.edu.br

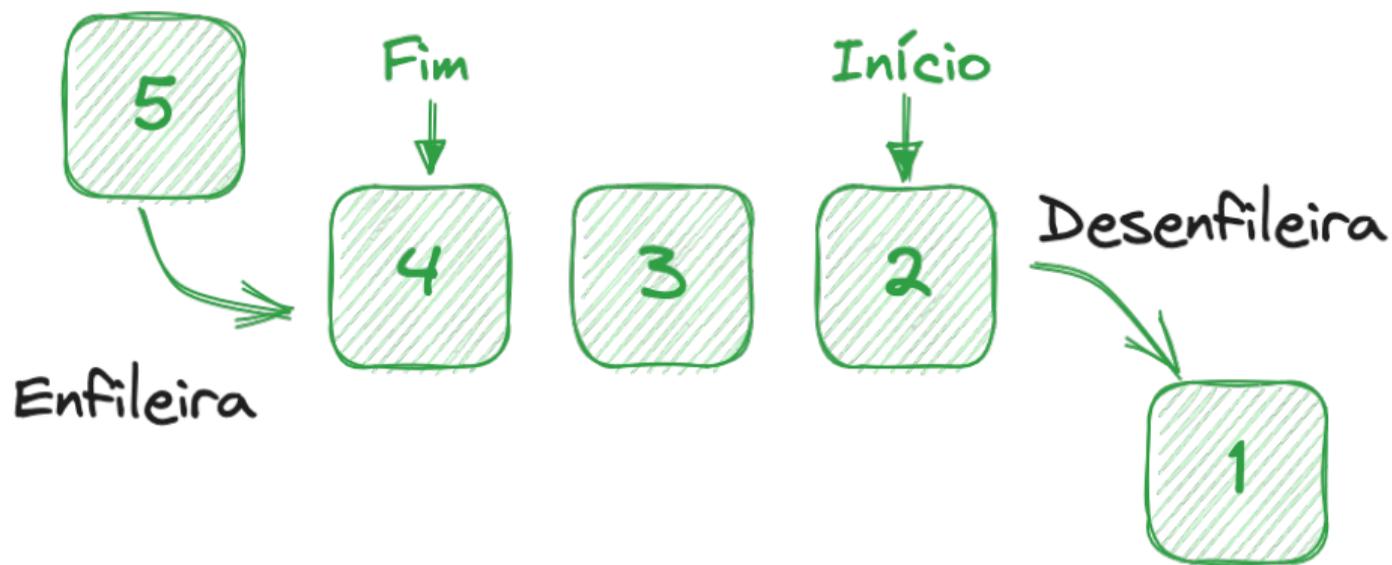
Licenciamento



Slides licenciados sob [Creative Commons "Atribuição 4.0 Internacional"](https://creativecommons.org/licenses/by/4.0/)

Fila

Queue



Fila

Tipo abstrato de dados

Sequência de elementos sendo que a **posição de um elemento** na sequência **não depende de seu valor, mas sim do momento em que foi inserido** na coleção

- Primeiro a chegar, primeiro a sair
 - *First-In First-Out (FIFO)*



Crédito: Image by macrovector on Freepik.com

Fila

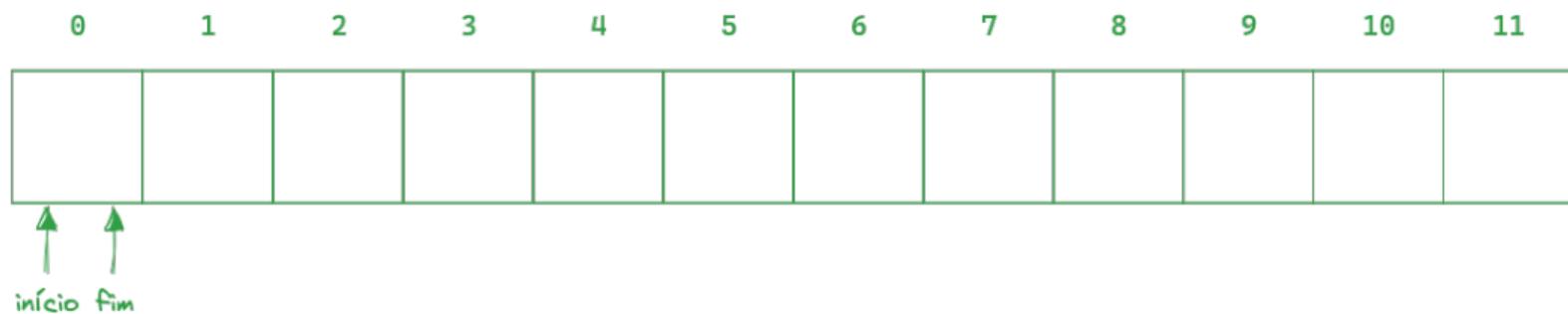
Uso em sistemas computacionais

- **Eventos gerados pelo usuário no ambiente gráfico**
 - teclas pressionadas, posição do cursor do mouse a cada clique
- **Armazenamento em área temporária (*buffering*)**
 - Pacotes de um fluxo de vídeo sob demanda
- **Roteadores de pacotes IP**
 - Pacotes ingressantes são armazenados em uma fila (ou mais)
- **Sistema de navegação por satélite**
 - Definição de rotas, paradas intermediárias
- **Servidor web**
 - Para atender pedidos concorrentes de clientes

Fila

Representada em um vetor

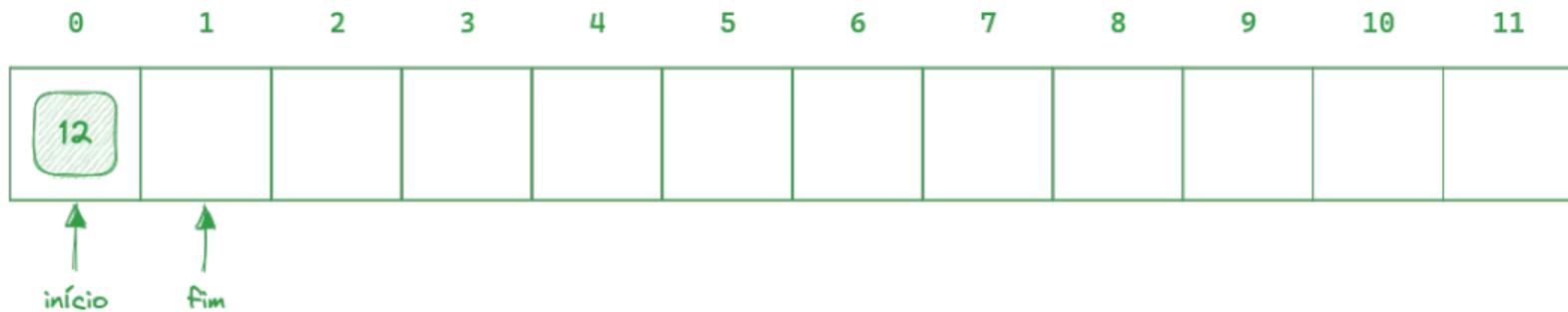
Fila vazia



Fila

Representada em um vetor

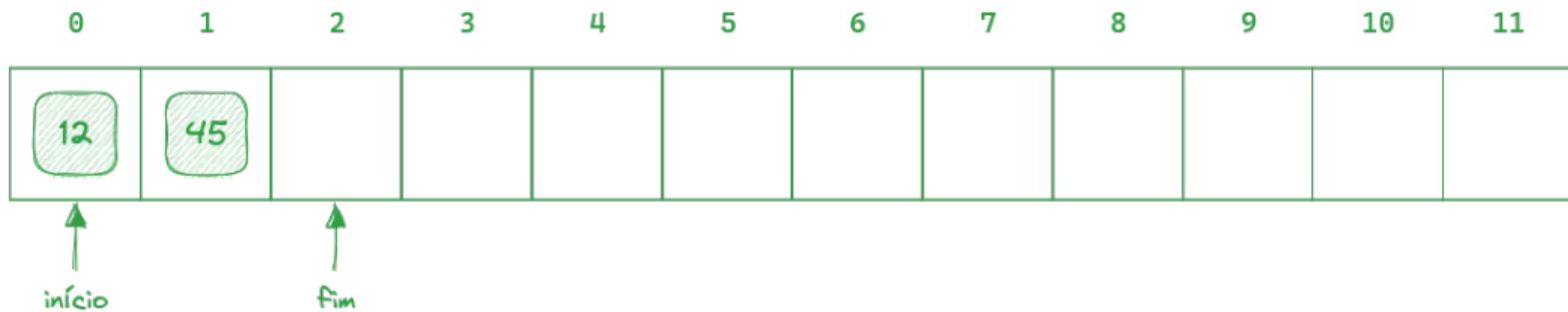
Elemento é inserido onde está o ponteiro *Fim*



Fila

Representada em um vetor

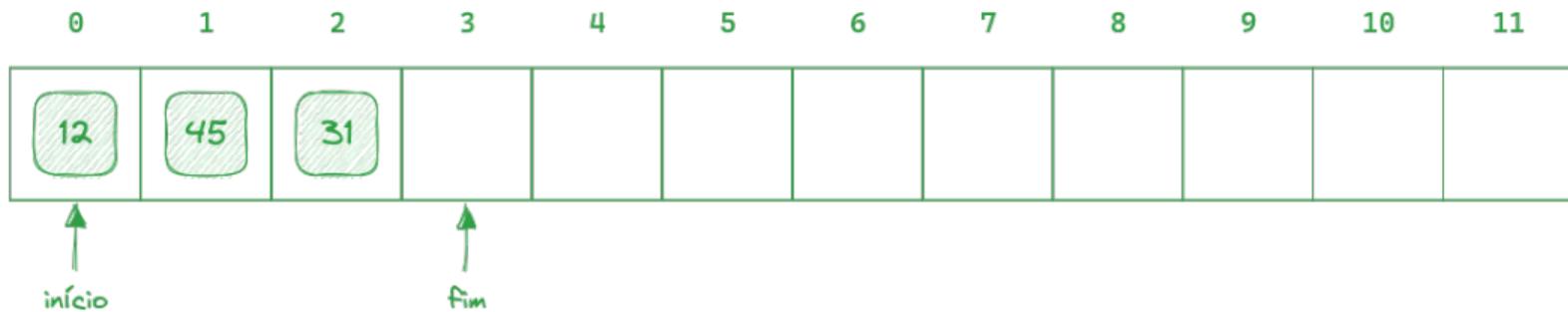
Elemento é inserido onde está o ponteiro *Fim*



Fila

Representada em um vetor

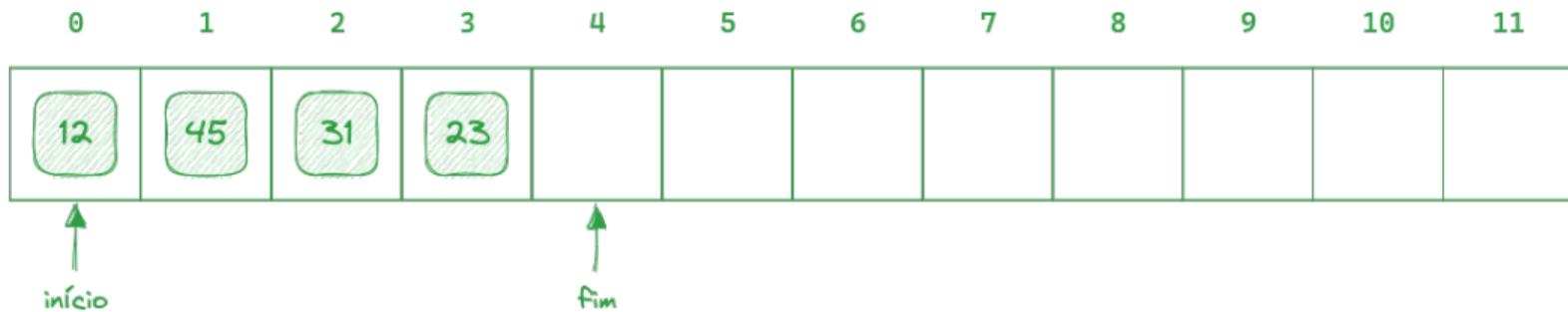
Elemento é inserido onde está o ponteiro *Fim*



Fila

Representada em um vetor

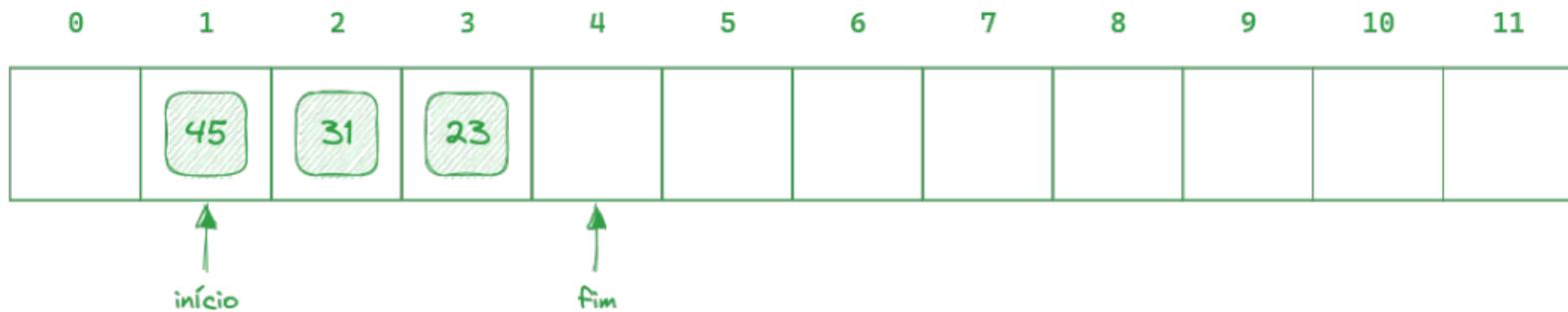
Elemento é inserido onde está o ponteiro *Fim*



Fila

Representada em um vetor

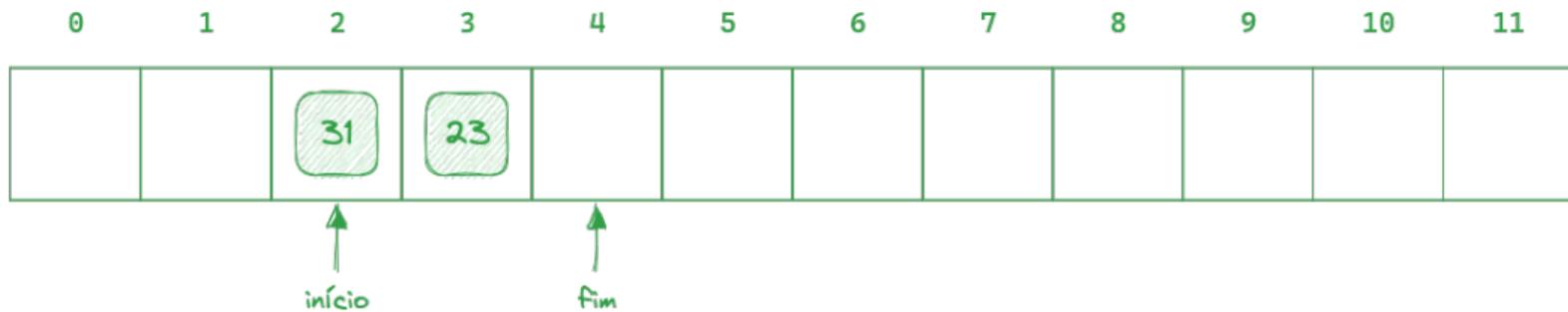
Elemento onde está o ponteiro *Início* é removido



Fila

Representada em um vetor

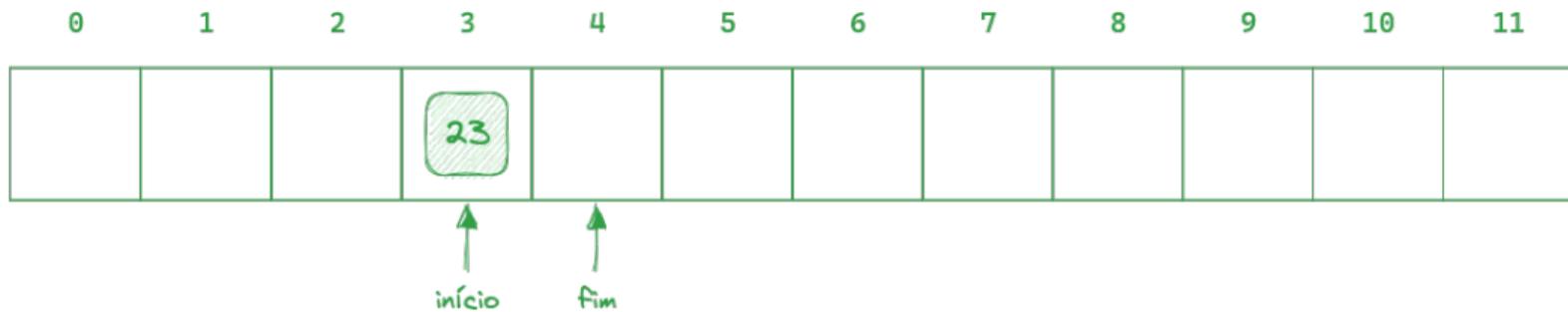
Elemento onde está o ponteiro **Início** é removido



Fila

Representada em um vetor

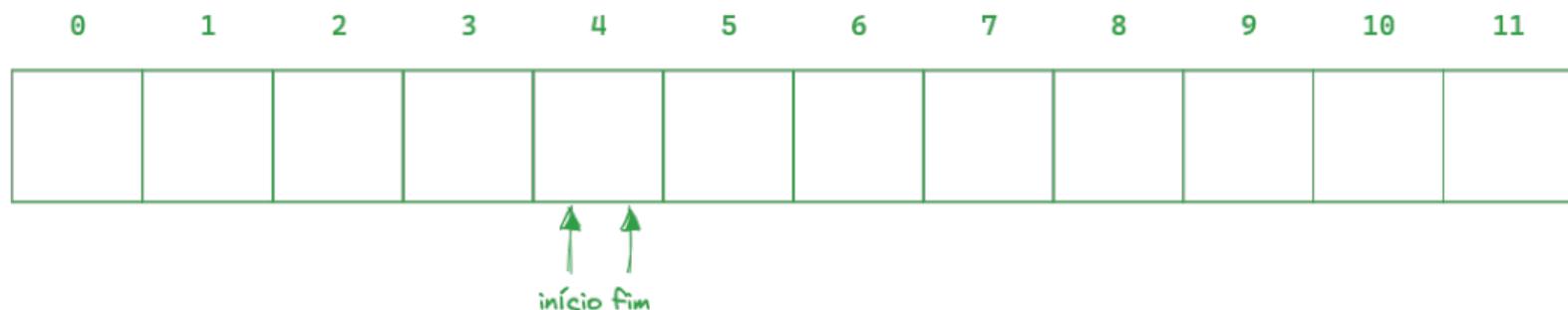
Elemento onde está o ponteiro Início é removido



Fila

Representada em um vetor

Fila vazia

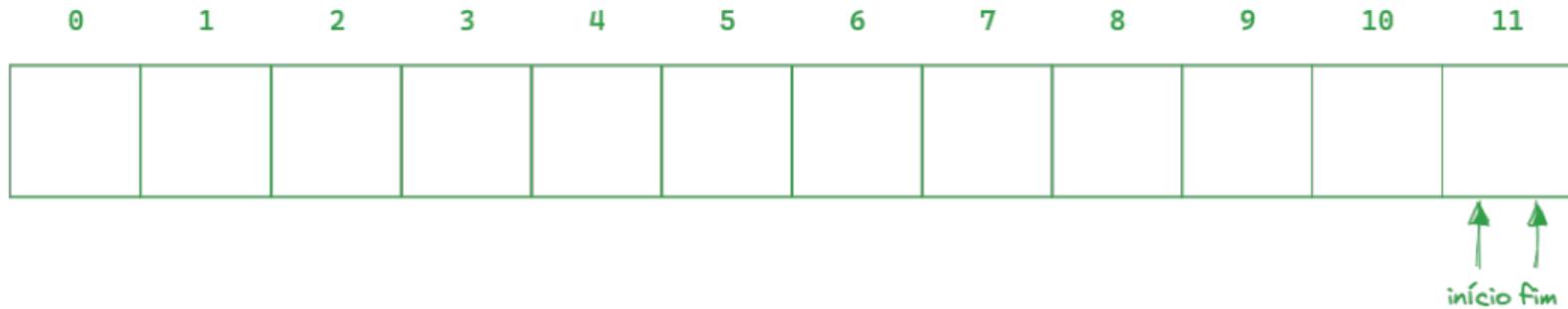


- Ao tentar remover elemento acontece um erro chamado **estouro negativo** (*underflow*)

Fila

Representada em um vetor

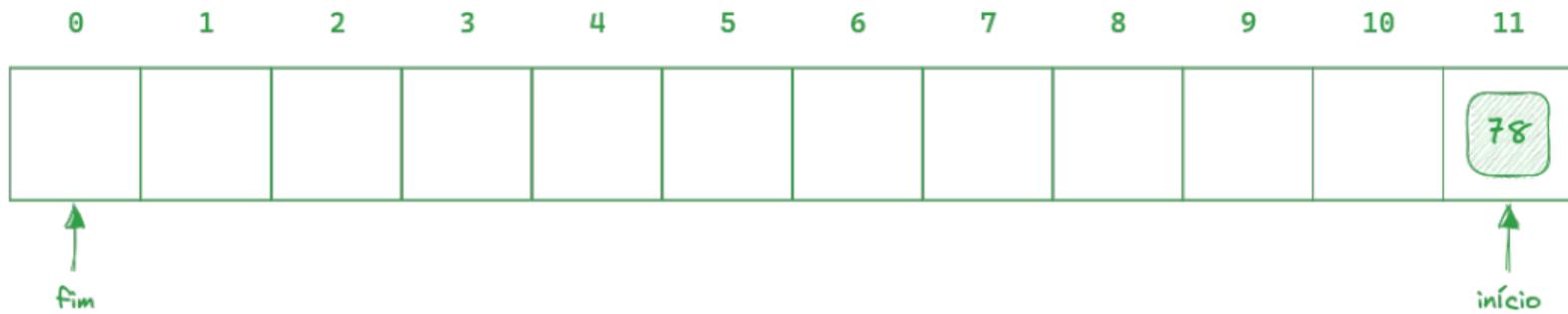
Fila vazia



Fila

Representada em um vetor

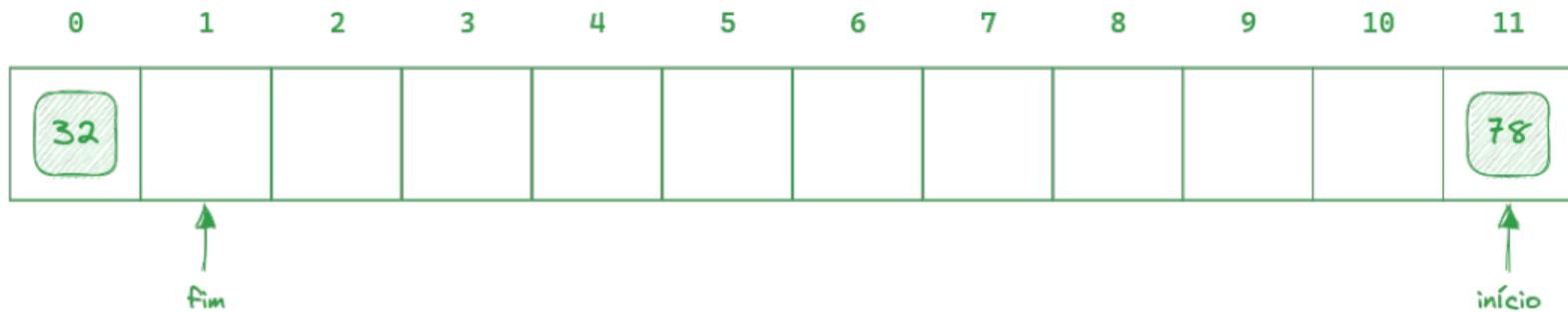
Elemento é inserido onde está o ponteiro *Fim*



Fila

Representada em um vetor

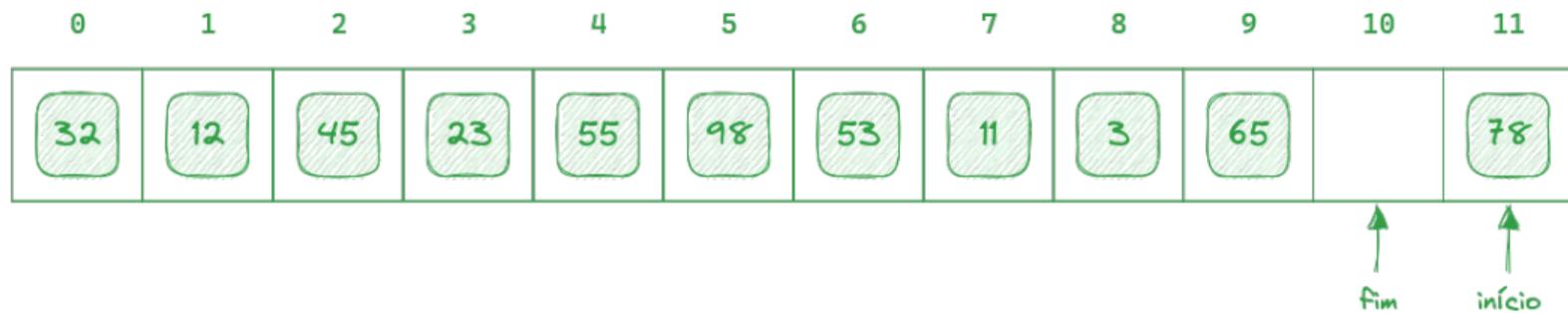
Elemento é inserido onde está o ponteiro *Fim*



Fila

Representada em um vetor

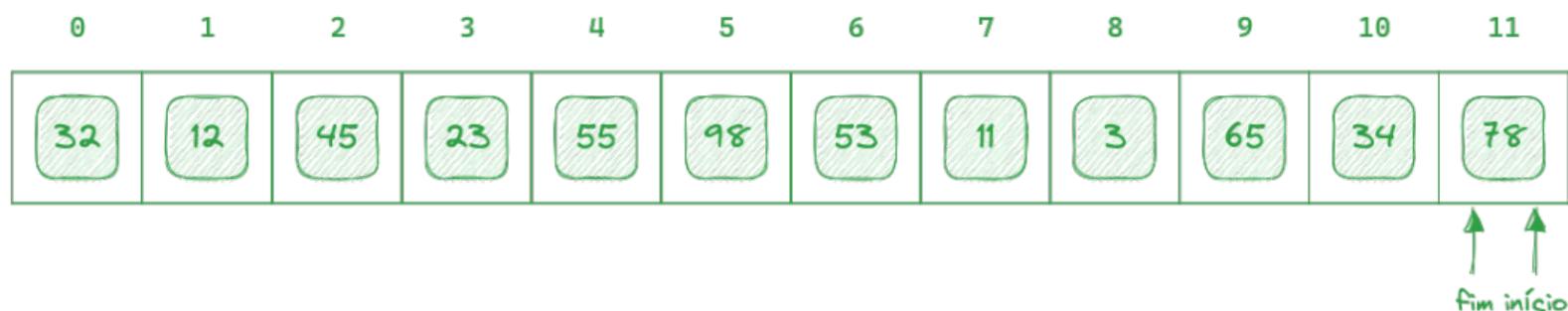
Elemento é inserido onde está o ponteiro *Fim*



Fila

Representada em um vetor

Fila cheia



- Ao tentar inserir um novo elemento acontece um erro chamado **estouro positivo** (*overflow*)

Fila

Operações básicas

Operação		Descrição
enfileirar	<code>enqueue(e)</code>	Adiciona um elemento no final da fila
desenfileirar	<code>dequeue()</code>	Remove o elemento que está no início da fila
início	<code>head()</code>	Retorna o elemento que está no início da fila
fim	<code>tail()</code>	Retorna o elemento que está no final da fila
tamanho	<code>size()</code>	Retorna o total de elementos na fila
vazia	<code>empty()</code>	Indica se a fila está vazia ou não
cheia	<code>full()</code>	Indica se a fila está cheia ou não

Complexidade das operações de enfileirar e desenfileirar

São constantes, logo, $O(1)$

Outros tipos de fila

■ Fila circular

- Início da fila conectado ao seu fim
- Exemplos de uso: escalonamento de processos na CPU, gerenciamento de memória

■ Deque – *double ended queue*

- Elementos podem ser adicionados ou removidos de qualquer lado da fila
- Exemplos de uso: escalonamento de tarefas e algoritmos de busca

■ Fila com prioridades

- A cada elemento é associada um nível de prioridade o qual determinará sua ordem para saída da fila
- Exemplos de uso: escalonamento de tarefas, roteamento de pacotes IP

Exercício 1

- Evolua a biblioteca `libprg`, criada por você na aula de lista de sequencial, para ofertar uma **estrutura de dados do tipo fila circular** representada em um vetor, para armazenar números inteiros
 - As definições das funções devem ser feitas obrigatoriamente no arquivo de cabeçalho `libprg.h`
 - As implementações das funções obrigatoriamente no arquivo com o nome `fila.c`
 - Deve-se implementar as operações básicas sobre filas listadas no *slide 6*

Exercício 2

- Crie um projeto com CMake de uma aplicação em C que dependa da biblioteca `libprg`, criada no exercício anterior
 - Faça uso do `FetchContent` no CMake para baixar a dependência
- Faça um menu interativo que permita ao usuário executar as seguintes operações usando uma estrutura de dados fila
 - 1 Criar uma fila que permita armazenar n elementos
 - 2 Inserir um novo elemento
 - 3 Remover um elemento (e imprimir na tela o elemento removido)
- Deve-se imprimir na tela o tempo gasto (relógio de parede) para executar cada uma das operações
- Deve-se informar o usuário caso uma operação não possa ser realizada (i.e se o usuário tentar remover um elemento de uma fila vazia, indicar que a fila está vazia)

Referências

Aula baseada em



CORMEN, Thomas H. et al. **Algoritmos: teoria e prática**. LTC, 2012.
Disponível em: <<https://app.minhabiblioteca.com.br/reader/books/9788595158092>>.



SZWARCFITER, Jayme Luiz; MARKENZON, Lilian. **Estruturas de dados e seus algoritmos**. LTC, 2010. Disponível em:
<<https://app.minhabiblioteca.com.br/reader/books/978-85-216-2995-5>>.