



# Algoritmo e Programação Filas

**#EU TENHO**  
PAIXÃO POR EDUCAR

Paulo Rodrigues Milhorato  
[paulo.rodriques@pitagoras.com.br](mailto:paulo.rodriques@pitagoras.com.br)  
Algoritmo e Programação

# Indagação inicial

O que é uma fila?

# Citação

*"A fila anda".*

*(Autor desconhecido)*

# Introdução

Quem nunca se deparou com a frase anterior?

└ Principalmente após tomar um pé na bunda!

Dizemos isso para termos a ideia de que algo se foi.

└ Mas que um novo algo está chegando!

A fila anda, sim, e as informações também!

# Introdução

Praticamente todo serviço ou recurso não está disponível a todos ao mesmo tempo.

Um caixa de banco, por exemplo, não possui atendimento imediato a todos que chegarem à agência.

Esperamos que algum dos recursos seja liberado e, então, nos dirigimos ao caixa.

Mas, por que você seria o próximo a utilizar o serviço?

# Introdução

Por que não seria a vez de uma outra pessoa?

Porque existe uma ordem pré-definida.

Esta ordem é conhecida por todos e é determinada pela ordem de chegada.

E, por qual motivo temos essa regra?

# Introdução

Primeiro, por questões de organização.

└ Imagina o caos se não houvesse uma regra!

Segundo, por questões de justiça.

└ Afinal, quem chegou primeiro, deve ser atendido primeiro.

Nas filas, o primeiro a chegar é o primeiro a ser atendido.

No mundo virtual, não existem portas ou "frentes".

└ Não será que você implemente isto.

# Introdução

Até porque, assim como na vida real, podem existir filas com atendimento prioritário.

- └ Idosos, gestantes...

Onde as filas são implementadas na Computação?

- └ Praticamente em todos os recursos computacionais, tais como, processador, impressora, teclado, requisições...

Vamos ver então como isto fica na programação!



# Filas

Na programação não inserimos pessoas na fila, mas sim informações.

Qual tipo de informações?

- └ Tipos primitivos, tipos abstratos, estruturas...

Um dos aspectos mais importantes é controlar o início e o final da fila.

Por quê?

- └ Porque as extremidades são os locais de manipulação.

# Filas

Assim, temos o princípio:

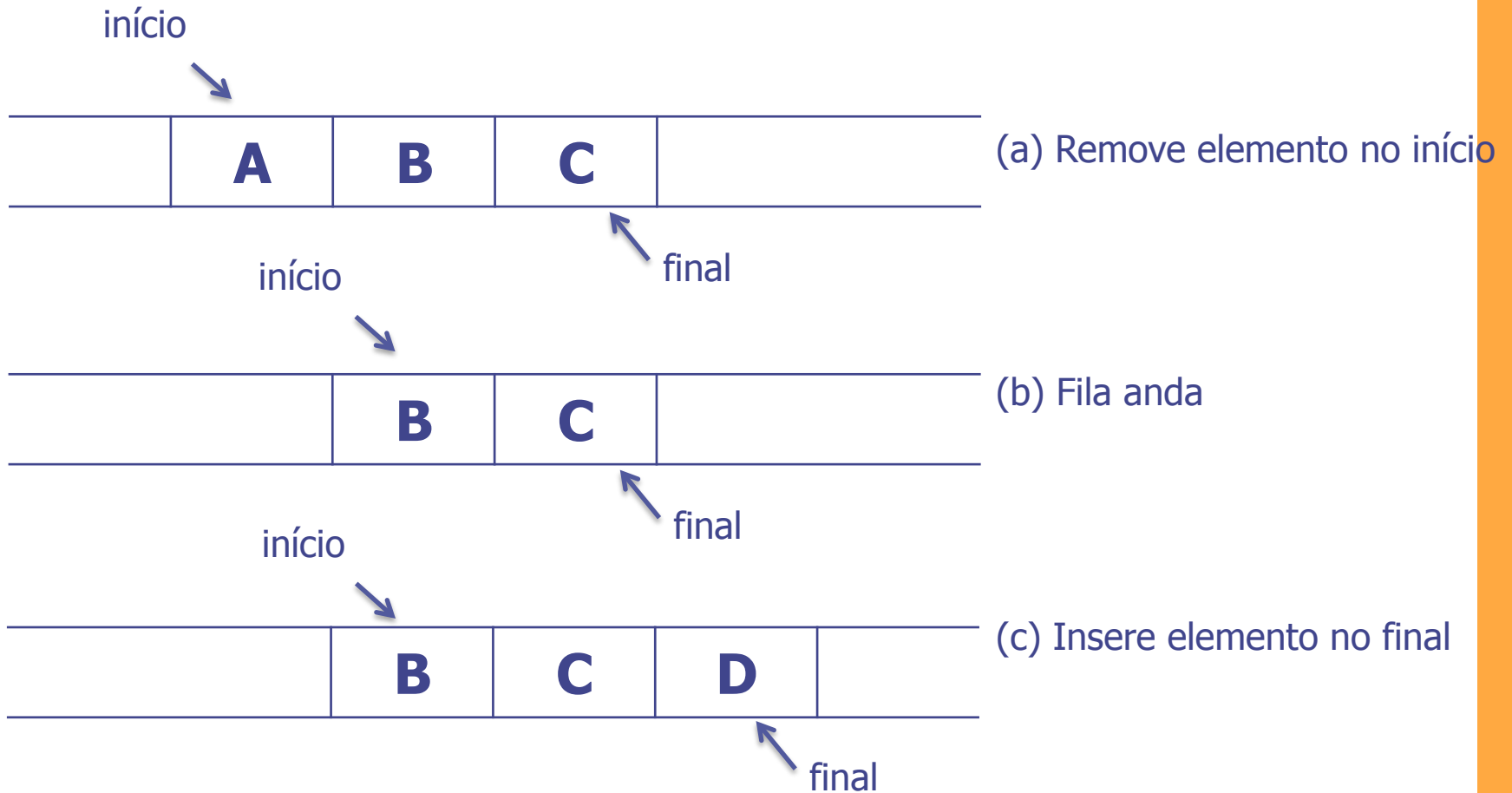
- ⇒ Toda informação que chega à fila é adicionada ao fim;
- ⇒ Toda informação que é consumida da fila é retirada do início.

Por isso, as filas possuem o comportamento FIFO (*first-in, first-out*).

- └ O primeiro a entrar é o primeiro a sair.

Vamos ver um esquema de um funcionamento de fila.

# Filas



Comportamento de uma fila.

# Filas

E como implementaríamos uma fila?

Se você pensou na lista ligada, está correto!

└ Podemos usar os nós da lista ligada!

Se você pensou em vetor, também está correto.

└ Mas não será por esta forma que implementaremos.

Quais são as duas funcionalidades de manipulação de uma fila?

R: Entrar na fila e sair da fila.

# Filas

Para entrar na fila:

- ⇒ Deve posicionar o elemento imediatamente após o último elemento que está nela.
- ⇒ Este elemento, então, se tornará o último da fila.
- ⇒ A última posição sempre será a procurada para um elemento se inserir na fila.

Para sair da fila:

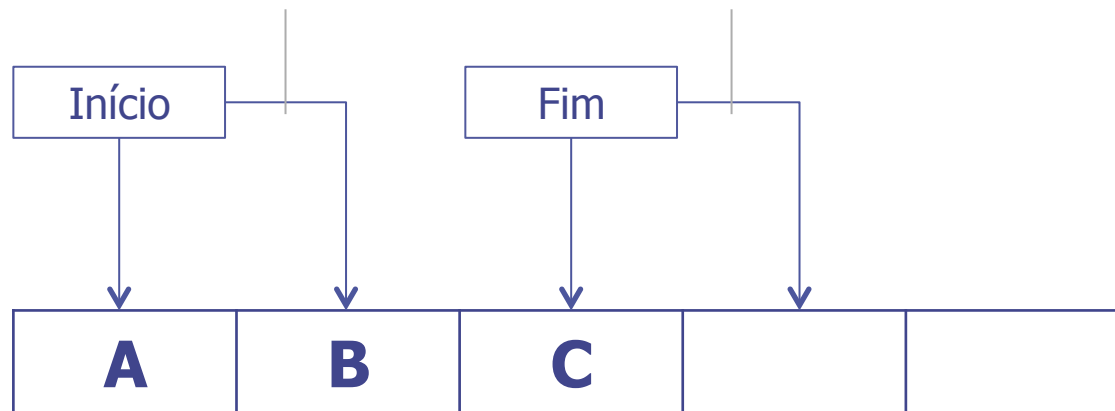
- ⇒ Apenas o primeiro elemento sai.
- ⇒ Quando este for liberado, o segundo se torna o primeiro, o terceiro se torna o segundo e assim por diante.

E como faremos isso em programação?

# Filas

Deslocamento do ponteiro início **após o atendimento** do cliente.

Deslocamento do ponteiro fim **ao chegar um novo cliente** cliente.



Quando o caixa chamar o próximo cliente, **ele sairá desta posição** por causa do ponteiro de início.

Quando um **novo cliente chega na fila**, obrigatoriamente ele deve assumir esta posição.

# Filas

Levando em consideração o diagrama anterior, fica claro que precisa-se de duas funções:

⇒ #1 Uma para inserir nova informação

⇒ #2 Uma para consumir informação

Quando quisermos adicionar uma nova informação, chamaremos a função inserir.

Quando quisermos remover uma informação, chamaremos de consumir.

# Filas – Inicializar

## Inicializar uma fila

⇒ Uma fila vazia significa uma fila sem informações inseridas.

⇒ Os ponteiros inicial e final devem apontar para NULL.

⇒ O tamanho da fila é iniciado com 0.



# Filas – Inserir

## Inserir em uma fila

- ⇒ Verificar se a fila está vazia.
- ⇒ Se estiver, início e final apontarão para o novo elemento inserido.
- ⇒ Se não estiver, deve ser inserido ao final, sendo que o antigo último será o penúltimo e o novo elemento o último.

# Filas – Imprimir

Exibir elementos de uma fila

⇒ Verificar se a fila está vazia.

⇒ Se estiver, não há elementos a serem impressos.

⇒ Se não estiver, devem ser exibidos os elementos e suas posições.

# Filas – Remove

Consumir elementos de uma fila

⇒ Verificar se a fila está vazia.

⇒ Se estiver, não há como consumir.

⇒ Se não estiver, verificar se há apenas um, para voltar com a fila para o estado vazia.

⇒ Se possuir 2 ou mais elementos, consumir o primeiro, o segundo será o primeiro, o terceiro será o segundo e assim por diante.

# Filas - Exercício

## 2. Faça um algoritmo que:

- a. Tenha uma estrutura que contenha:
  - i. Nome:
  - ii. ValorPagamento:
- b. O algoritmo deve enfileirar os clientes na fila do caixa e ir consumindo, ou seja, removendo-os da fila. A cada remoção, imprima a fila novamente. Crie um menu para operar a fila.

# Referências

(Cap. 3-4) EDELWEISS, Nina; GALANTE, Eduarda. **Estrutura de dados**. Porto Alegre: Bookman, 2009.

(Cap. 5-6) KERNIGHAN, B. W; RITCHIE, D. M. **The C programming language**. 2. ed. New Jersey: Prentice Hall, 1988.

(Cap. 9) PIVA JUNIOR, D. *et al.* **Estrutura de dados e técnicas de programação**. Rio de Janeiro: Elsevier, 2014.

(Cap. 20) SCHILDT, Herbert. **C completo e total**. 3. ed. Tradução de Roberto Carlos Mayer. São Paulo: Makron Books, 1996. 810 p.



---

[www.faculdadepitagoras.com.br](http://www.faculdadepitagoras.com.br)

