



Algoritmos e Estrutura de Dados II

Aula 07
Filas

Prof. Dr. Dilermando Piva Jr
2º Semestre - CDN

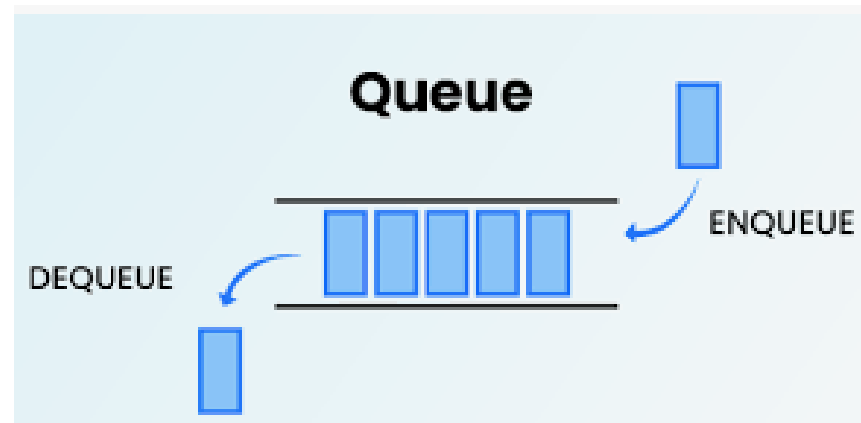


Você já passou por essa situação?

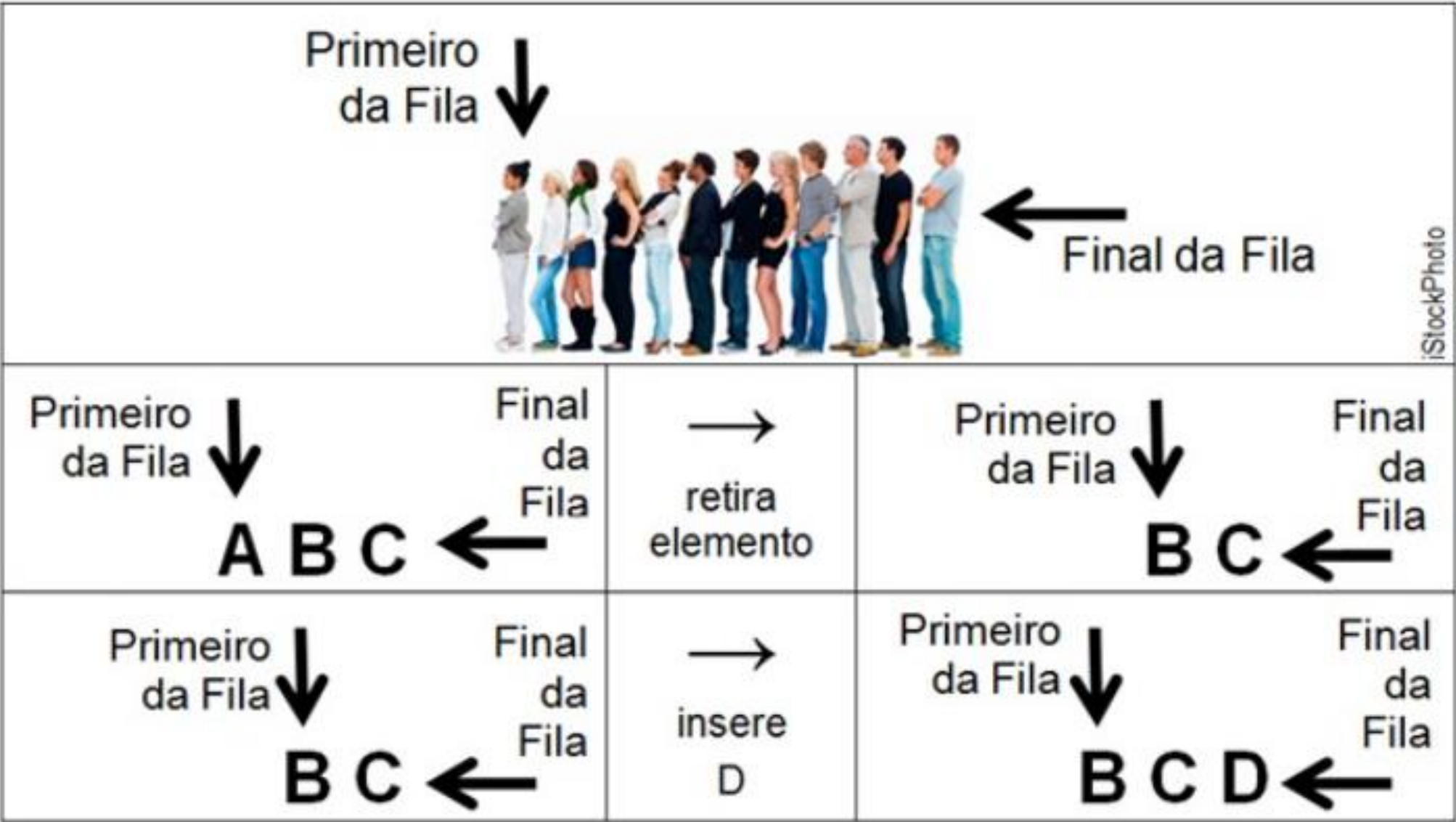


FILAS

- Uma fila é uma estrutura de dados linear em que a inserção de elementos é realizada em uma extremidade (final) e a remoção é realizada na outra extremidade (início).
- Assim como em uma fila de pessoas no mundo real, o primeiro a entrar será o primeiro a sair, o que define o princípio de ordenação FIFO, ou do inglês, First In First Out.
- Além disso, filas devem ser restritivas no sentido de que um elemento não pode passar na frente de seu antecessor.



FILAS



iStockPhoto

FILAS

`x = Q.dequeue()`



`Q.enqueue(21)`



`Q.enqueue(74)`



FILAS

Operações básicas

Operação	Conteúdo	Retorno	Descrição
<code>q.is_empty()</code>	<code>[]</code>	<code>True</code>	Verifica se fila está vazia
<code>q.enqueue(4)</code>	<code>[4]</code>		Inserir elemento no final
<code>q.enqueue('dog')</code>	<code>['dog', 4]</code>		Inserir elemento no final
<code>q.enqueue(True)</code>	<code>[True, 'dog', 4]</code>		Inserir elemento no final
<code>q.size()</code>	<code>[True, 'dog', 4]</code>	<code>3</code>	Retorna número de elementos da fila
<code>q.is_empty()</code>	<code>[True, 'dog', 4]</code>	<code>False</code>	Verifica se fila está vazia
<code>q.enqueue(8.4)</code>	<code>[8.4, True, 'dog', 4]</code>		Inserir elemento no final
<code>q.dequeue()</code>	<code>[8.4, True, 'dog']</code>	<code>4</code>	Remove elemento do início
<code>q.dequeue()</code>	<code>[8.4, True]</code>	<code>'dog'</code>	Remove elemento do início
<code>q.size()</code>	<code>[8.4, True]</code>	<code>2</code>	Retorna número de elementos da fila

FILAS

Operações básicas

Link: <https://www.cs.usfca.edu/~galles/visualization/QueueArray.html>

Queue (Array Implementaion)

Enqueue Dequeue Clear Queue

Head 0 Tail 8



`x = Q.dequeue()`



`Q.enqueue(21)`



`Q.enqueue(74)`



Filas

Implementação da classe Fila

```
class Queue:
```

```
    # Inicia com uma fila vazia
```

```
    def __init__(self):
```

```
        self.itens = []
```

```
    # Verifica se fila está vazia
```

```
    def is_empty(self):
```

```
        return self.itens == []
```

```
    # Adiciona elemento no início da fila
```

```
    def enqueue(self, item):
```

```
        self.itens.insert(0, item)
```

```
        print('ENQUEUE %s' %item)
```

```
    # Remove elemento do final da fila
```

```
    def dequeue(self):
```

```
        print('DEQUEUE')
```

```
        return self.itens.pop()
```

```
    # Retorna o número de elementos da fila
```

```
    def size(self):
```

```
        return len(self.itens)
```

```
    # Imprime a fila na tela
```

```
    def print_queue(self):
```

```
        print(self.itens)
```

```
# TESTANDO
```

```
Q = Queue()
```

```
Q.print_queue()
```

```
Q.enqueue(1)
```

```
Q.enqueue(2)
```

```
Q.enqueue(3)
```

```
Q.print_queue()
```

```
Q.dequeue()
```

```
Q.dequeue()
```

```
Q.print_queue()
```

```
Q.enqueue(7)
```

```
Q.enqueue(8)
```

```
Q.enqueue(9)
```

```
Q.print_queue()
```

```
print(Q.is_empty())
```


VAMOS PARA A PRÁTICA ?!!!



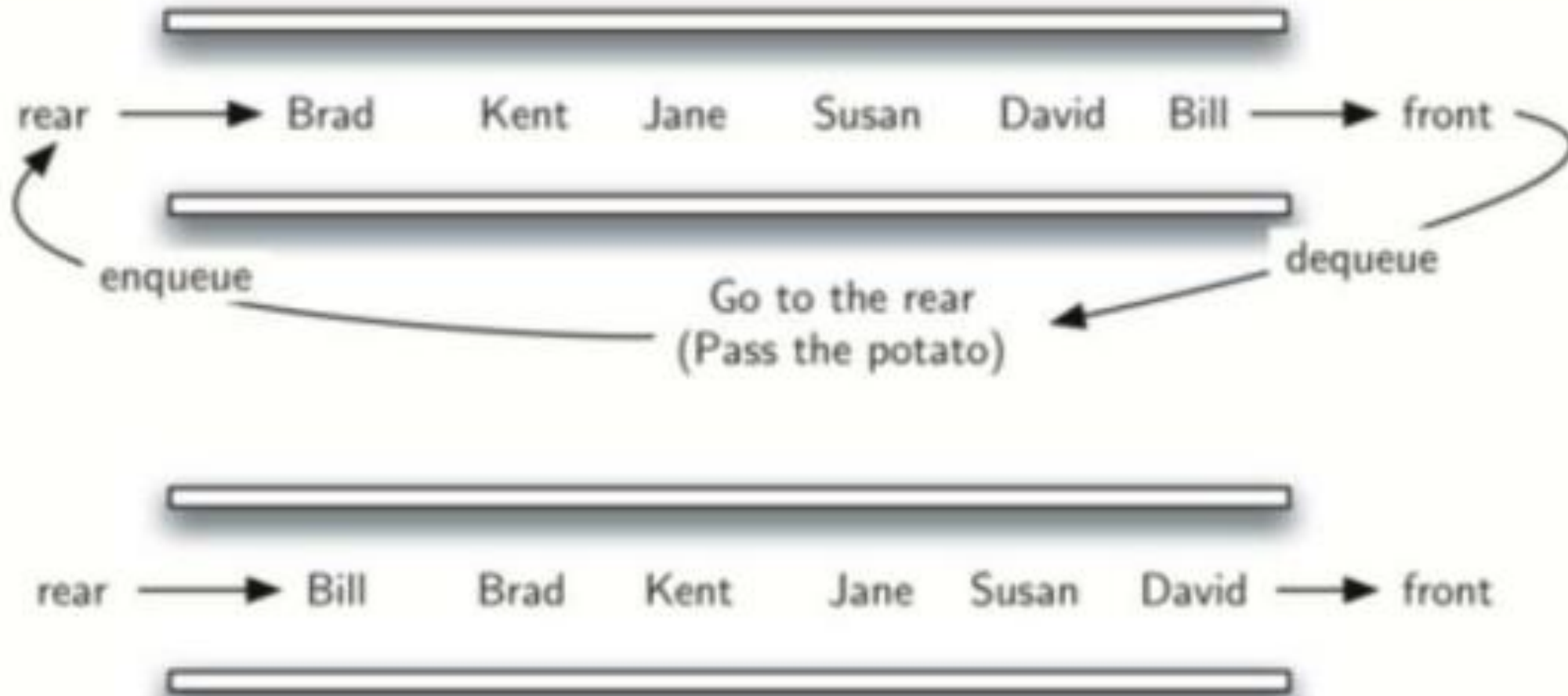
APLICAÇÃO DE FILAS

JOGO: Batata Quente

- Neste jogo, as crianças formam um círculo e passam um item qualquer (batata) cada um para o seu vizinho da frente o mais rápido possível.
- Em um certo momento do jogo, essa ação é interrompida (queimou) e a criança que estiver com o item (batata) na mão é excluída da roda.
- O jogo então prossegue até que reste apenas uma única criança, que é a vencedora.
- Para simular um círculo (roda), utilizaremos uma fila da seguinte maneira: a criança que está com a batata na mão será sempre a aquela que estiver no início da fila. Após passar a batata, a simulação deve instantaneamente remover e inserir a criança, colocando-a novamente no final da fila. Ela então vai esperar até que todas as outras assumam o início da fila, antes de assumir essa posição novamente.
- Após um número pré estabelecido MAX de operações enqueue/dequeue, a criança que ocupar o início da fila será removida e outro ciclo da brincadeira é realizado.
- O processo continua até que a fila tenha possui tamanho um.

APLICAÇÃO DE FILAS

JOGO: Batata Quente



Batata Quente

```
# Utilização da classe QUEUE.
# Simula o jogo batata_quente
def batata_quente(nomes, MAX):
    # Cria fila para simular roda
    fila = Queue()

    # Coloca os N nomes em cada posição
    for nome in nomes:
        fila.enqueue(nome)
```

```
# Inicia a lógica do jogo
    while fila.size() > 1:
        # Para simular MAX passagens da batata
        for i in range(MAX):
            # Remove o primeiro e coloca no final
            fila.enqueue(fila.dequeue())

            # Quem parar no início da fila, está com batata
            # Deve ser eliminado da fila
            fila.dequeue()

    # Após N-1 rodadas, retorna a fila com o vencedor
    vencedor = fila.dequeue()
```

```
# TESTANDO
v = batata_quente(['Alex', 'Julia', 'Carlos', 'Maria', 'Ana', 'Caio'], 7)
print(f'0 vencedor é {v}')
```