

Movimento FIFO em Python

Mário Leite

A disciplina Estruturas de Dados sempre apresenta desafios para quem está iniciando em cursos de Processamento de Dados, Ciência da Computação e Engenharia de Software; e entre os conceitos que mais geram dúvidas está o movimento **FIFO**; um dos padrões fundamentais no estudo de filas (*queues*). O termo FIFO significa **F**irst **I**n, **F**irst **O**ut, que em português quer dizer “**o primeiro que entra é o primeiro a sair**”. Esse tipo de movimento aparece em diversos contextos: filas de banco, caixas de supermercado, ordens de atendimento, buffers de impressão e em muitas aplicações de *software*.

Para entender intuitivamente, basta imaginar uma fila de pessoas: quem chega primeiro é atendido prioritariamente; quem chega depois só é atendido quando sua vez chegar. A ordem de saída portanto, é a mesma ordem de entrada - diferente da pilha LIFO, onde a ordem é invertida.

Aqui está o básico do funcionamento básico de uma fila FIFO:

A fila é normalmente representada horizontalmente, com dois lados:

- Início da fila → onde os elementos saem (remoção - Dequeue)
- Fim da fila → onde os elementos entram (inserção - Enqueue)

O esquema geral é o seguinte:

INÍCIO → [Item 1] [Item 2] [Item 3] [Item 4] ← **FIM**

↑ Sai primeiro ↑ Entra por aqui

- Item 1, por ter entrado primeiro, será o primeiro a sair.
- Novos itens sempre entram no final da fila.

Um exemplo clássico: fila para servir almoço

Imagine uma fila de alunos se servindo no refeitório da escola.

Os alunos chegam nesta ordem:

1. Ana
2. Bruno
3. Carla
4. Diego

A fila é representada assim:

INÍCIO → [Ana] [Bruno] [Carla] [Diego] ← **FIM**

Entrada de pessoas na fila (enqueue): Sempre que alguém chega, é colocado no final da fila.

Saída da fila (dequeue)

Quem sai é sempre a pessoa no início.

A ordem de saída será:

1. Ana (1ª a entrar → 1ª a sair)
2. Bruno
3. Carla
4. Diego

A tabela abaixo mostra a ordem comparativa de entrada e saída

| Ordem de Entrada | Ordem de Saída |
|------------------|----------------|
| 1 - Ana | 1 - Ana |
| 2 - Bruno | 2 - Bruno |
| 3 - Carla | 3 - Carla |
| 4 - Diego | 4 - Diego |

Observe que a ordem é preservada, exatamente como ocorre em uma fila real.

Conclusões importantes

- ♦ FIFO = First In, First Out
- ♦ O primeiro a entrar é sempre o primeiro a sair
- ♦ Usado em filas, buffers, sistemas de atendimento, implementações de redes, etc.
- ♦ É uma estrutura organizada “em linha reta”, com entradas em um lado e saídas no outro.

A estrutura **FIFO**, também conhecida como **fila**, é amplamente utilizada em diversos cenários, como processamento de tarefas, sistemas de fila de impressão e *buffers* de dados.

O programa “**MovimentoFIFO**”, codificado em “Python”, cria uma classe para mostrar o movimento do tipo **FIFO**, usando estruturas híbridas, como deque (fila dupla), que permite o comportamento de pilha ou fila.

A **figura 1** mostra a saída do programa, considerando as quatro pessoas citadas acima.

```
'''
MovimentoFIFO.py
-----
Cria uma classe para mostrar o movimento do tipo FIFO (First In First Out) usando
estruturas híbridas, como deque (fila dupla), que permite o comportamento de
pilha ou fila.
-----
'''
from collections import deque

class ClsFilaDePessoas:
    def __init__(self):
        self.Deqfila = deque()  #utiliza deque para eficiência

    def EntrarNaFila(self, pessoa):

        self.Deqfila.append(pessoa)
        print(f"{pessoa} entrou na fila.")

    def AtenderPessoa(self):
        if self.Deqfila:
            pessoa = self.Deqfila.popleft()
            print(f"Atendendo: {pessoa}")
        else:
            print("Nenhuma pessoa na fila para atender!")

    def ExibirFila(self):
        print(f"Fila atual: {list(self.Deqfila)}")

#-----
```

```

def main():
    print()
    Fila = ClsFilaDePessoas()

    #Adiciona pessoas na ordem especificada
    Fila.EntrarNaFila("Ana")
    Fila.EntrarNaFila("Bruno")
    Fila.EntrarNaFila("Carla")
    Fila.EntrarNaFila("Diego")

    #Exibe e processa a fila
    Fila.ExibirFila()
    Fila.AtenderPessoa()
    Fila.AtenderPessoa()
    Fila.ExibirFila()
    Fila.AtenderPessoa()
    Fila.AtenderPessoa()
    Fila.AtenderPessoa() #tentativa com a fila vazia

#=====
#Programa principal
if(__name__ == "__main__"):
    main()
#Fim do programa "MovimentoFIFO" -----

```

```

IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/MovimentoFIFO.py =====
Ana entrou na fila.
Bruno entrou na fila.
Carla entrou na fila.
Diego entrou na fila.
Fila atual: ['Ana', 'Bruno', 'Carla', 'Diego']
Atendendo: Ana
Atendendo: Bruno
Fila atual: ['Carla', 'Diego']
Atendendo: Carla
Atendendo: Diego
Nenhuma pessoa na fila para atender!
>>>
Ln: 17 Col: 0

```

Figura 1 - Saída do programa “MovimentoFIFO”

Nota1: Postagem baseada no livro: “*1001 Programas em Python Para Você Aprender Praticando - Volume 3: Nível Avançado*”. Publicado pelo autor na “Amazon” e no “Clube de Autores”

<https://www.amazon.com.br/Curso-B%C3%A1sico-Programa%C3%A7%C3%A3o-Teoria-Pr%C3%A1tica/dp/8539908700>

Nota2: Acesse o *link* abaixo para ver meus mais recentes livros de Python publicado pelo “Clube de Autores”, no formato impresso, da coleção “*1001 Programas em Python Para Você Aprender Praticando*”:

Volume1: Nível Básico (500 programas)

Volume2: Nível Intermediário (300 programas)

Volume3: Nível Avançado (201 programas)

<https://clubedeautores.com.br/livros/autores/mario-leite>

Para adquirir PDF dos livros: marleite@gamil.com