

Estrutura de Dados I

Pilhas

Implementação de Pilhas por meio de
Estruturas Auto-Referenciadas

- Fila é uma lista linear em que todas as inserções são realizadas em um extremo da lista, e todas as retiradas e, geralmente, os acessos são realizados no outro extremo da lista.
- O modelo intuitivo de uma fila é o de uma fila de espera em que as pessoas no início da fila são servidas primeiro e as pessoas que chegam entram no fim da fila.
- São chamadas listas fifo (“first-in”, “first-out”).

- Existe uma ordem linear para filas que é a “ordem de chegada”.
- São utilizadas quando desejamos processar itens de acordo com a ordem “primeiro-que-chega, primeiro-atendido”.
- Sistemas operacionais utilizam filas para regular a ordem na qual tarefas devem receber processamento e recursos devem ser alocados a processos.

- TAD Filas

1. Conjunto de operações:

2. Criar uma fila vazia.

3. Enfileirar o item x no final da fila.

4. Desenfileirar.

- Essa função retorna o item x no início da fila e o retira da fila.

5. Verificar se a fila está vazia.

- Essa função retorna true se a fila está vazia; do contrário, retorna false.

- A fila é implementada por meio de nós (células).
- Cada nó (célula) contém um item da fila (que contém os dados armazenados) e uma referência para outro nó (outra célula).
- A classe Fila contém uma referência para a frente da fila e uma referência para a parte de trás da fila.

Classe Pessoaa

- nome: String
- matricula: String
- telefone: String
- + get....
- + set ...

Classe Fila

Classe No

- dados: Livro
- ant: No

- inicio: No
- fim: No
- tam: int
- + enfileirar(Pessoa): void
- + desenfileirar(): Pessoa
- + vazia(): lógico
- + getTamanho(): int

Classe Main

- leia: Scanner, static
- + main(): void, static
- + menu(): int, static
- + obtemLPessoa(): Pessoa, static

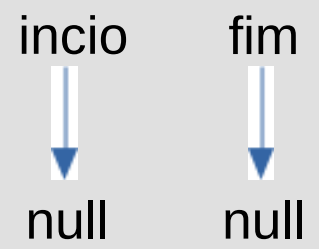


designed by freepik

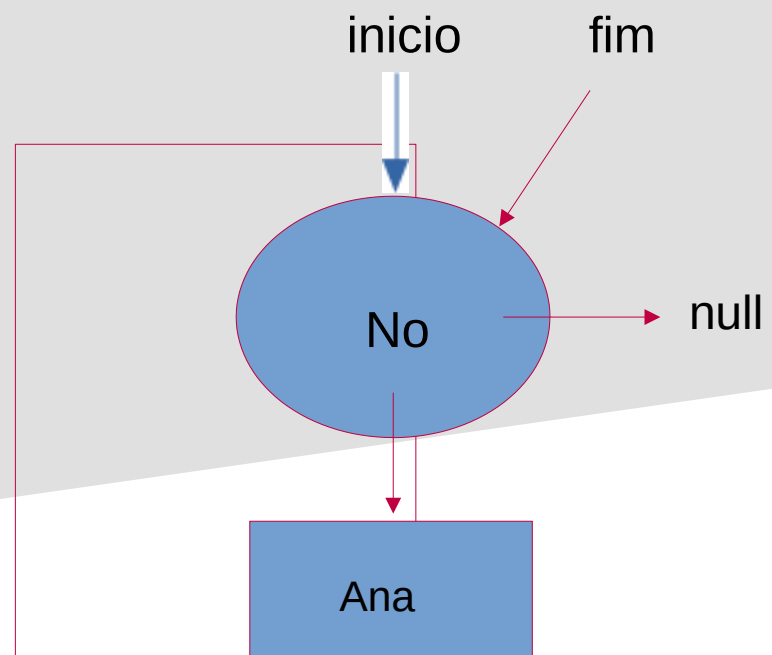




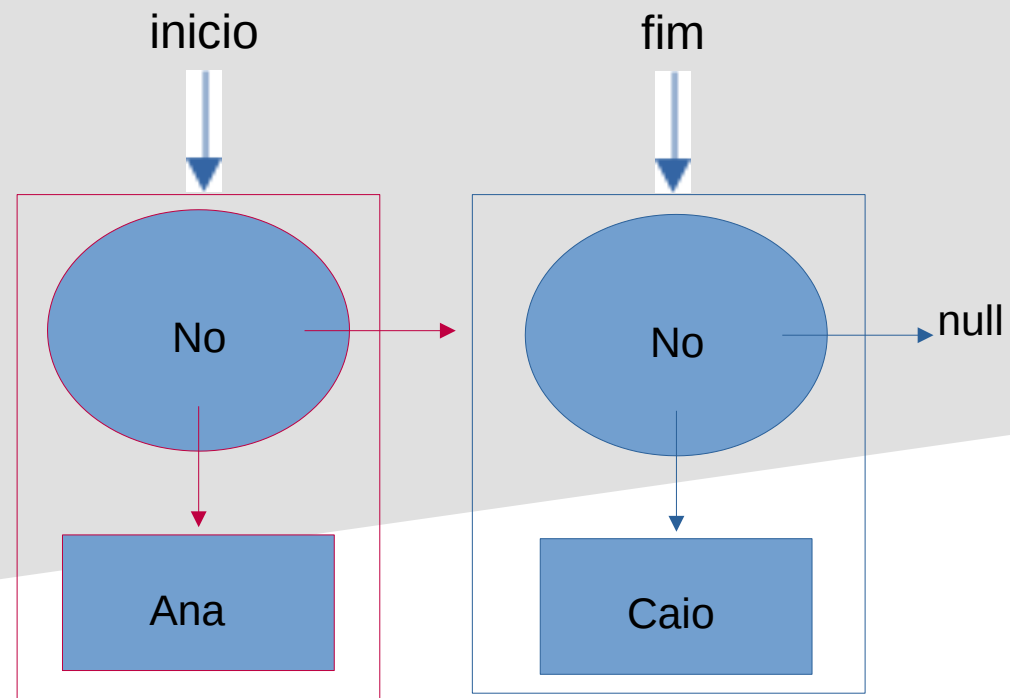
Fila vazia



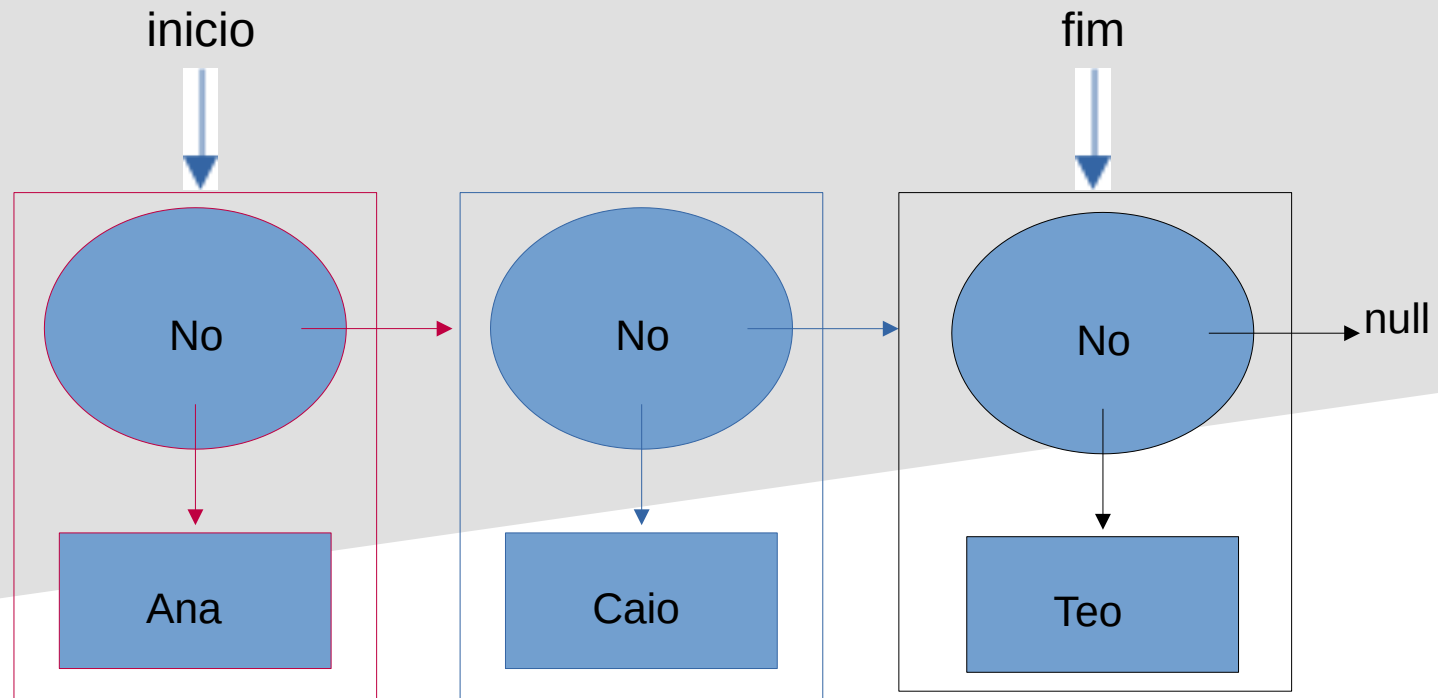
Fila com 1 Pessoa



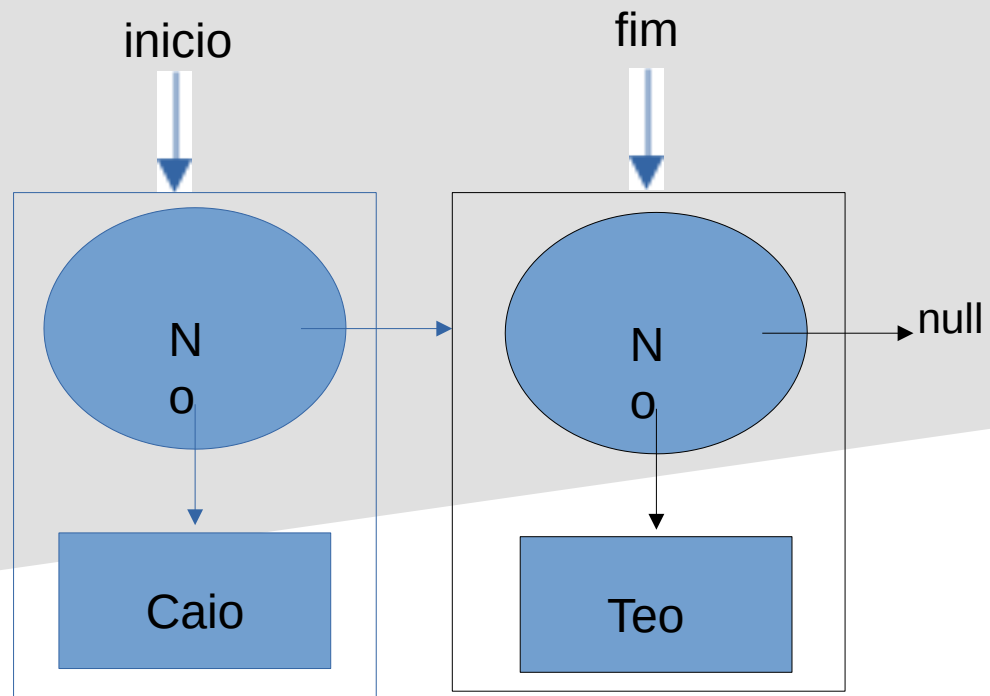
Fila com 2 Pessoas



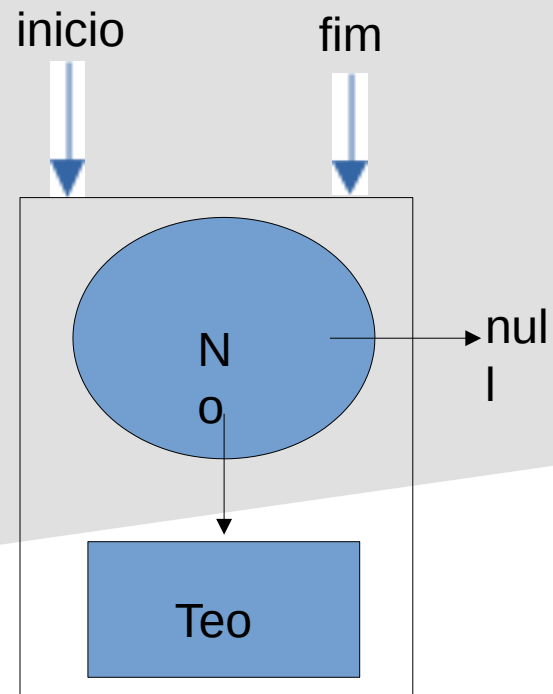
Fila com 3 Pessoas



Fila com 2 Pessoas



Fila com 1 Pessoas



Fila vazia

inicio



nul

|

fim



nul

|


```
package filaencadeada;
```

```
public class Pessoa {  
    private String nome, matricula, telefone;  
    public Pessoa(String pNom, String pMat, String pTel){  
        nome = pNom;  
        matricula = pMat;  
        telefone = pTel;  
    }  
    @Override  
    public String toString() {  
        return(nome+"\n"+matricula+"\n"+telefone+"\n");  
    }  
}
```

```
package filaencadeada;
```

```
public class Fila {  
    public class No{  
        private Pessoa dados;  
        private No ant;  
        public No(Pessoa aux) {  
            dados = aux;  
            ant = null;  
        }  
    }  
    private int tam;  
    private No fim, inicio;  
    public Fila() {  
        tam = 0;  
        fim = inicio = null;  
    }  
}
```

```
public int getTam() {
    return tam;
}
public boolean vazia() {
    return (inicio == null);
}
public void
enfileirar(Pessoa aux) {
    No novo = new No(aux);
    if(vazia()) {
        fim = inicio = novo;
        tam++;
    }
    else {
        fim.ant = novo;
        fim = novo;
        tam++;
    }
}
```

```
public Pessoa desenfileirar() {  
    if(vazia()) return null;  
    Pessoa aux = inicio.dados;  
    inicio = inicio.ant;  
    tam--;  
    if(inicio == null) fim = null;  
    return aux;  
}  
}
```

```
package filaencadeada;

import java.util.Scanner;

public class Main {
    static Scanner leia = new Scanner(System.in);

    public static Pessoa obtemPessoa() {
        String titulo, autor, editora;
        titulo = leia.nextLine();
        System.out.println("Digite o Título do Livro.\n");
        titulo = leia.nextLine();
        System.out.println("Digite o Autor do Livro.\n");
        autor = leia.nextLine();
        System.out.println("Digite a Editora do Livro.\n");
        editora = leia.nextLine();
        Pessoa aux = new Pessoa(titulo, autor, editora);
        return aux;
    }
}
```

```
public static int menu() {  
    int valor;  
    System.out.println("Digite:");  
    System.out.println("1 - para enfileirar um livro.");  
    System.out.println("2 - para desenfileirar um livro.");  
    System.out.println("3 - para encerrar o programa.");  
    valor = leia.nextInt();  
    return valor;  
}
```

```
public static void main(String[] args) {  
    Fila fila = new Fila();  
    int n;  
    Pessoa novo = null;  
    do {  
        n = menu();  
        switch(n) {  
            case 1:  
                novo = obtemPessoa();  
                fila.enqueue(novo);  
                System.out.println("Pilha com "+fila.getTam()+" livros.");  
                novo = null;  
                break;  
        }  
    }  
}
```

case 2:

 novo = fila.desenfileirar();

if(novo == **null**) System.**out**.println("Erro!");

else System.**out**.println(novo.toString());

 System.**out**.println("Pilha com "+fila.getTam()+" livros.");

 novo = **null**;

break;

case 3:

 System.**out**.println("Programa encerrando!");

 System.**out**.println("Pilha com "+fila.getTam()+" livros.");

}

} **while**(n != 3);

}

}