

Estruturas de Dados

Estruturas de Dados básicas
utilizando Vetores

- Introdução
- Pilhas usando Vetores
- Filas usando Vetores

Estruturas de Dados - Definição

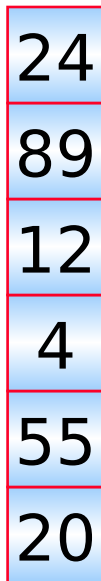
Estruturas de Dados é a disciplina que estuda as técnicas computacionais para a organização e manipulação eficiente de quaisquer quantidades de informação.

Estruturas de Dados - Aspectos

- Em um projeto de software, 2 aspectos devem ser considerados:
 - de que forma estão organizados os dados - qual a sua estrutura;
 - quais procedimentos atuam sobre estes dados - que operações podem ser realizadas sobre eles.
- Ao estudar estruturas de dados teremos sempre este par:
 - um conjunto estruturado de informações:
 - uma classe de objetos ou um tipo de dados;
 - um conjunto definido de operações sobre estes dados:
 - um conjunto de métodos ou funções.

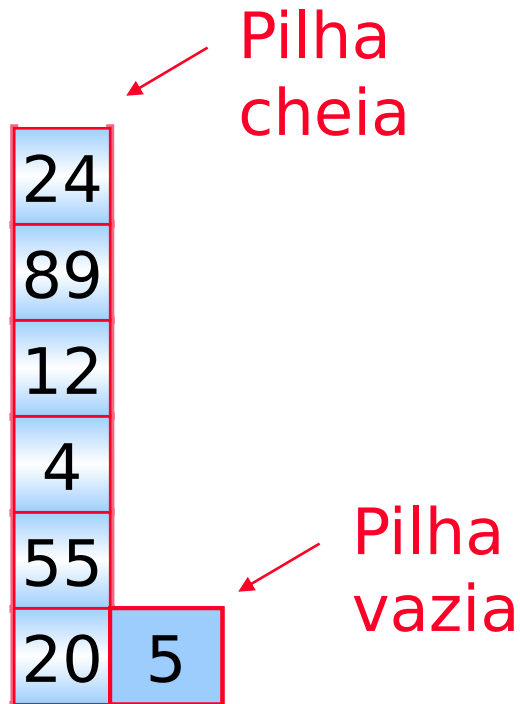
Pilhas

A Pilha é uma estrutura de dados cujo funcionamento é inspirado no de uma pilha “natural”.



24
89
12
4
55
20

Pilhas usando Vetores



- Vetores possuem um espaço limitado para armazenar dados;
- necessitamos definir um espaço grande o suficiente para a nossa pilha;
- necessitamos de um indicador de qual elemento do vetor é o atual topo da pilha.

Modelagem da Pilha

- Aspecto Estrutural:
 - necessitamos de um vetor para armazenar as informações;
 - necessitamos de um indicador da posição atual do topo da pilha;
 - necessitamos de uma constante que nos diga quando a pilha está cheia e duas outras para codificar erros.
- Pseudo-código:

```
constantes MAXPILHA = 100;
```

```
classe Pilha {  
    T m_dados[MAXPILHA]; // ou T* m_dados e instancia no  
    inteiro topo;        // construtor m_dados = new T[MAXPILHA];  
};
```

Modelagem da Pilha

- Aspecto Funcional:
 - colocar e retirar dados da pilha;
 - testar se a pilha está vazia ou cheia;
 - Inicialização.
- Colocar e retirar dados da pilha:
 - Empilha(dado)
 - Desempilha(dado)
 - Topo
- Testar se a pilha está vazia ou cheia:
 - PilhaCheia
 - PilhaVazia
- Inicializar ou limpar:
 - InicializaPilha

Algoritmo InicializaPilha

```
Nulo inicializaPilha()
```

```
início
```

```
    topo <- -1;
```

```
    //m_dados <- new T[MAXPILHA];
```

```
fim;
```


Algoritmo PilhaCheia

```
Booleano MÉTODO pilhaCheia()  
início  
    SE (topo = MAXPILHA - 1) ENTÃO  
        RETORNE (Verdadeiro) ;  
    SENÃO  
        RETORNE (Falso) ;  
fim;
```

Algoritmo PilhaVazia

```
Booleano MÉTODO pilhaVazia()
```

```
início
```

```
    SE (topo = -1) ENTÃO
```

```
        RETORNE (Verdadeiro)
```

```
    SENÃO
```

```
        RETORNE (Falso) ;
```

```
fim;
```

Algoritmo Empilha

```
Void MÉTODO empilha(T dado)
início
    SE (pilhaCheia) ENTÃO
        JogueExceção (ERROPILHACHEIA) ;
    SENÃO
        topo <- topo + 1
        dados[topo] <- dado;
        RETORNE (topo) ;
    FIM SE
fim;
```

Algoritmo Desempilha

T MÉTODO desempilha()

início

SE (pilhaVazia) ENTÃO

JogueExceção (ERROPILHAVAZIA) ;

SENÃO

topo <- topo - 1;

RETORNE (dados[topo+1]) ;

FIM SE

fim;

Algoritmo Topo

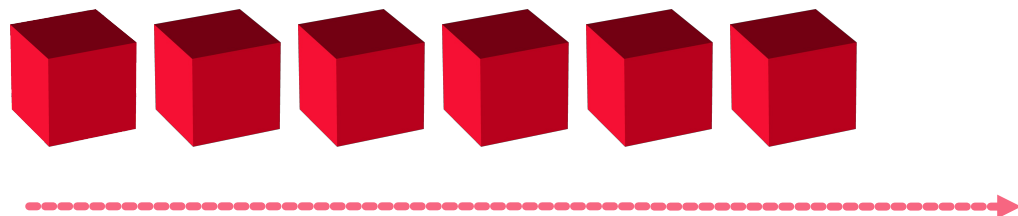
```
T MÉTODO topo()  
início  
    SE (pilhaVazia) ENTÃO  
        JogueExceção (ERROPILHAVAZIA) ;  
    SENÃO  
        RETORNE (dados [ topo ] ) ;  
    FIM SE  
fim;
```

Modelagem da Pilha com Vetor - Trabalho 1

- Implemente uma classe Pilha todas as operações vistas;
- Implemente a pilha usando Templates
- Implemente a pilha com um numero de elementos variável definido na instanciação
- Use as melhores práticas de orientação a objetos
- Documente todas as classes, métodos e atributos.
- Aplique os testes unitários disponíveis no moodle da disciplina para validar sua estrutura de dados.

Filas usando Vetores

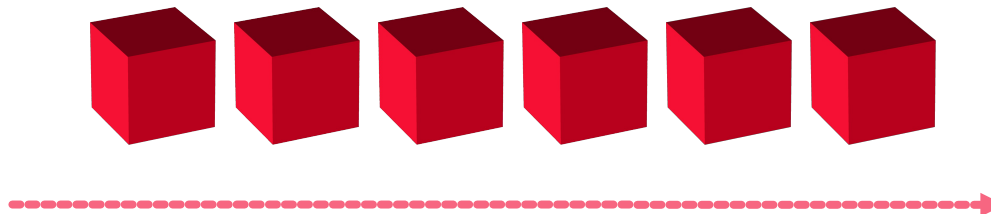
- A Fila é uma estrutura de dados que simula uma fila da vida real (ou quase).
- Possui duas operações básicas:
 - incluir no fim da fila;
 - retirar do começo da fila;
 - chamada de Estrutura-FIFO:
First-In, First-Out - O primeiro que entrou é o primeiro a sair...



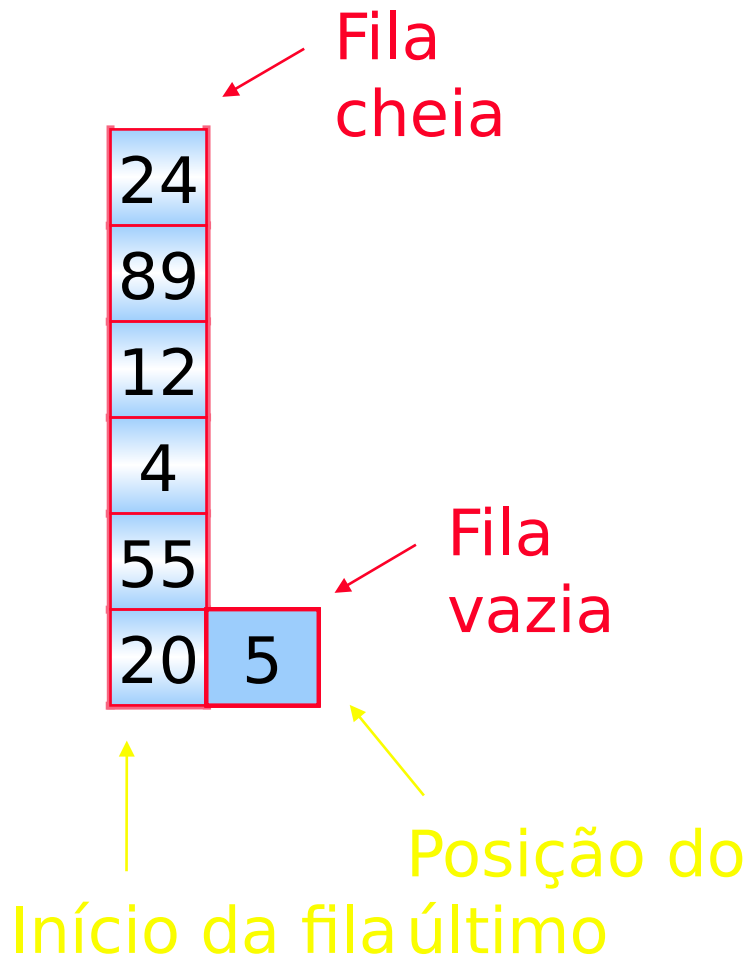
Filas

- É uma estrutura de dados importantíssima para:
- gerência de dados/processos por ordem cronológica:
 - Fila de impressão em uma impressora de rede;
 - Fila de pedidos de uma expedição ou tele-entrega.
- simulação de processos seqüenciais:
 - chão de fábrica: fila de camisetas a serem estampadas;
 - comércio: simulação de fluxo de um caixa de supermercado;
 - tráfego: simulação de um cruzamento com um semáforo.

Fila



Filas usando Vetores

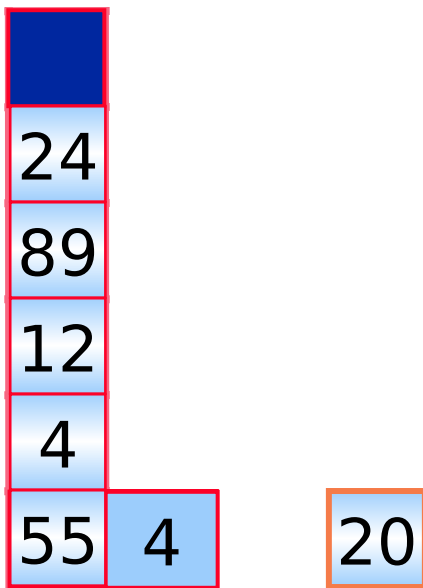


- Vetores possuem um espaço limitado para armazenar dados;
- necessitamos definir um espaço grande o suficiente para a nossa fila;
- necessitamos de um indicador de qual elemento do vetor é o atual fim da fila (último);
- incluímos sempre no fim.

Algoritmo Retira

- Procedimento:

- testamos se há elementos;
- decrementamos o fim da fila (último);
- salvamos o primeiro elemento em variável auxiliar;
- empurramos tudo para a frente.



Início da fila último

Posição do

Modelagem da Fila com Vetor

- Aspecto Funcional:
 - colocar e retirar dados da fila;
 - testar se a fila está vazia ou cheia;
 - Inicialização.
- Colocar e retirar dados da fila:
 - Inclui(dado)
 - Retira
 - Último
- Testar se a fila está vazia ou cheia:
 - FilaCheia
 - FilaVazia
- Inicializar ou limpar:
 - InicializaFila

Modelagem da Fila com Vetor - Exercício

- Inserir e retirar dados da fila: elabore um algoritmo para retirar um elemento de uma fila com vetor:
 - utilize a mesma filosofia definida na implementação da pilha;
 - utilize um laço para percorrer o vetor;
 - lembre-se de testar antes se há elementos;
 - lembre-se de que se há só um elemento, a fila ficará vazia.

Modelagem da Fila com Vetor - Trabalho 2

- Implemente uma classe Fila todas as operações vistas;
- Implemente a fila usando Templates
- Implemente a fila com um numero de elementos variável definido na instanciação
- Use as melhores práticas de orientação a objetos
- Documente todas as classes, métodos e atributos.
- Aplique os testes unitários disponíveis no moodle da disciplina para validar sua estrutura de dados.



Atribuição-Uso Não-Comercial-Compartilhamento pela Licença 2.5 Brasil

Você pode:

- copiar, distribuir, exibir e executar a obra
- criar obras derivadas

Sob as seguintes condições:

Atribuição — Você deve dar crédito ao autor original, da forma especificada pelo autor ou licenciante.

Uso Não-Comercial — Você não pode utilizar esta obra com finalidades comerciais.

Compartilhamento pela mesma Licença — Se você alterar, transformar, ou criar outra obra com base nesta, você somente poderá distribuir a obra resultante sob uma licença idêntica a esta.

Para ver uma cópia desta licença, visite <http://creativecommons.org/licenses/by-nc-sa/2.5/br/> ou mande uma carta para Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.