

ESTRUTURA DE DADOS

PILHA

8/4/2008

Estrutura de Dados
Prof. Ademar Schmitz, M.Sc.

1

BIBLIOGRAFIA

- GOODRICH, Michael. T., TAMASSIA, Roberto. **Estruturas de Dados e Algoritmos em Java**. São Paulo: Bookman, 2002.
 - Capítulo 4: Pilhas, Filas e Deques.
- WEISS, Mark A. **Data Structures & Algorithm Analysis in Java**. Addison Wesley Longman, 1999.
 - Chapter 3: Lists, Stacks, and Queues

8/4/2008

Estrutura de Dados
Prof. Ademar Schmitz, M.Sc.

2

DEFINIÇÃO: PILHA

- Estrutura de dados na qual o último elemento a entrar é o primeiro a sair.
- Conhecida como LIFO (Last-In First-Out).
- Pilhas (stack) é uma lista linear onde as operações de inserção e remoção são feitas por um único extremo denominado topo.

8/4/2008

Estrutura de Dados
Prof. Ademar Schmitz, M.Sc.

3

EXEMPLO: PILHA

- Mundo real:
 - Pilha de pratos
 - Pilha de livros
- Computação:
 - Navegadores de Internet
 - Editores de texto
 - Chamada entre métodos
 - Avaliação de expressões aritméticas

8/4/2008

Estrutura de Dados
Prof. Ademar Schmitz, M.Sc.

4

TAD PILHA: STACK

- **push(o)**: insere o objeto *o* no topo da pilha; ocorre um erro se a pilha estiver cheia.
 - Entrada: objeto
 - Saída: nenhuma
- **pop()**: retira o objeto no topo da pilha e o retorna; ocorre um erro se a pilha estiver vazia.
 - Entrada: nenhuma
 - Saída: objeto

8/4/2008

Estrutura de Dados
Prof. Ademar Schmitz, M.Sc.

5

TAD PILHA: STACK

- **size()**: retorna o número de objetos na pilha.
 - entrada: nenhuma
 - saída: inteiro
- **isEmpty()**: retorna um booleano indicando se a pilha está vazia ou não.
 - entrada: nenhuma
 - saída: booleano
- **top()**: retorna o objeto no topo da pilha, sem retirá-lo; ocorre um erro se a pilha estiver vazia.
 - entrada: nenhuma
 - saída: objeto

8/4/2008

Estrutura de Dados
Prof. Ademar Schmitz, M.Sc.

6

TAD PILHA: STACK

- Implementar um TAD envolve dois passos:
 - Definir uma interface que descreve os nomes dos métodos que o TAD suporta e como eles são declarados e usados.
 - Fornecer uma classe concreta que implemente os métodos descritos na interface associada com o TAD.

8/4/2008

Estutura de Dados
Prof. Ademar Schmitz, M.Sc.

7

INTERFACE STACK

```
public interface Stack {  
  
    public int size();  
  
    public boolean isEmpty();  
  
    public Object top() throws EmptyStackException;  
  
    public void push(Object element) throws  
        FullStackException;  
  
    public Object pop() throws EmptyStackException;  
}
```

8/4/2008

Estutura de Dados
Prof. Ademar Schmitz, M.Sc.

8

IMPLEMENTAÇÃO COM VETOR

- Definir uma variável *top* que possui o seguinte significado:
 - *top* é o índice do último elemento em S.

8/4/2008

Estutura de Dados
Prof. Ademar Schmitz, M.Sc.

9

IMPLEMENTAÇÃO COM VETOR

0	Processo1	top
1	Processo2	
2	Processo3	
3		
4		
5		
6		
7		
8		
9		

8/4/2008

Estutura de Dados
Prof. Ademar Schmitz, M.Sc.

10

IMPLEMENTAÇÃO COM VETOR

- Cada um dos métodos usa um número constante de operações envolvendo operações aritméticas, comparações e atribuições.
- Simples, eficiente e muito usada em uma variedade de aplicações.

8/4/2008

Estutura de Dados
Prof. Ademar Schmitz, M.Sc.

11

IMPLEMENTAÇÃO COM VETORES

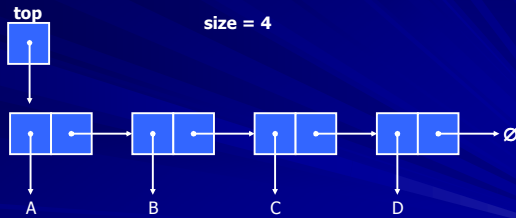
- Aspecto negativo:
 - Deve assumir um limite superior *n* para o tamanho máximo da pilha.
 - Pode precisar de muito menos espaço e, neste caso, teríamos desperdício de memória.
 - Pode precisar de mais espaço e, neste caso, nossa implementação de pilha pode derrubar a aplicação.

8/4/2008

Estutura de Dados
Prof. Ademar Schmitz, M.Sc.

12

IMPLEMENTAÇÃO COM LISTA ENCADEADA



8/4/2008

Estutura de Dados
Prof. Ademar Schmitz, M.Sc.

13

ANÁLISE ASSINTOTICA DAS OPERAÇÕES

OPERAÇÃO	VETOR	LISTA
size()	$O(1)$	$O(1)$
isEmpty()	$O(1)$	$O(1)$
top()	$O(1)$	$O(1)$
push(o)	$O(1)$	$O(1)$
pop()	$O(1)$	$O(1)$

8/4/2008

Estutura de Dados
Prof. Ademar Schmitz, M.Sc.

14