

# **Curso: Ciência da Computação**

## **Disciplina: Estrutura de Dados 1**

Professor: Clayton Zambon

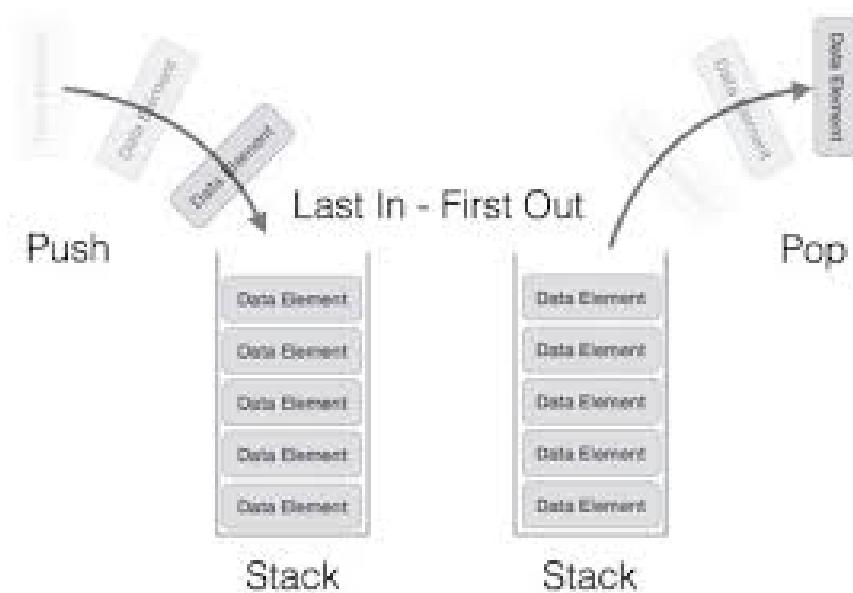
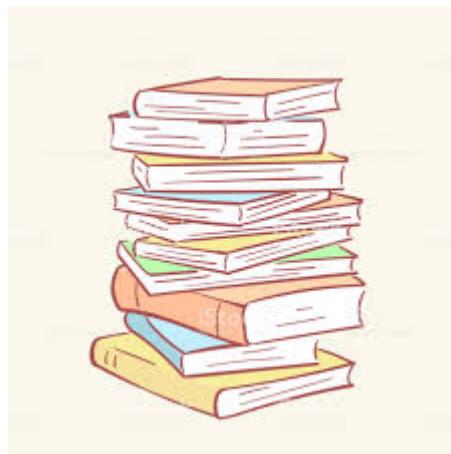
## 4. Pilha (Stack)

4.1. Conceitos e Aplicação

## 4. Pilha (Stack)

### 4.1. Conceitos e Aplicação

- Uma estrutura do tipo Pilha é uma sequência de elementos do mesmo tipo, como as Listas e Filas.
- Seus elementos possuem estrutura interna abstraída, ou seja, sua complexidade é arbitrária e não afeta o seu funcionamento.



## 4. Pilha (Stack)

### 4.1. Conceitos e Aplicação

- Uma Pilha é um tipo especial de Lista:

- Inserções e Exclusões de elementos ocorrem apenas no início da Lista (no Topo);

- Uma Pilha é manipulada pelo mecanismo LIFO (*Last In First Out*). O último a chegar é sempre o primeiro a sair, e a entrada de novos elementos sempre se dá no fim da Pilha;

- Sua estrutura é análoga ao conceito que temos de uma Pilha em geral como Pilha de pratos ou pilha de livros por exemplo.

- Aplicações de Pilhas:

- Análise de expressões matemáticas;
- Gerenciamento de memória em Tempo de Compilação;
- Operações como desfazer e refazer em aplicações;
- Controle de navegação em Browsers;
- Usada em softwares de sistemas incluindo compiladores e interpretadores (A maioria dos compiladores C usa pilha quando passa argumentos para funções);

## 4. Pilha (Stack)

### 4.1. Conceitos e Aplicação

- Em uma Pilha podemos realizar as seguintes operações:

- Criar a Pilha;
- Inserir um elemento no INÍCIO (Topo) da Pilha (Push);
- Excluir um elemento do INÍCIO (Topo) da Pilha (Pop);
- Acessar um elemento do INÍCIO (Topo) da Pilha;
- Destruir a Pilha;

- Essas operações dependem do tipo de alocação de memória usada:

- Estática;
- Dinâmica;

## 4. Pilha (Stack)

## **4.1. Conceitos e Aplicação**

## - Pilha – Representação Gráfica

## Ação:

1. push (A)
  2. push (B)
  3. push (C)
  4. pop ( ) – recupera C
  5. push (D)
  6. pop ( ) – recupera D
  7. pop ( ) – recupera B
  8. pop ( ) – recupera A

		C		D			
	B	B	B	B	B		
A	A	A	A	A	A	A	
1.	2.	3.	4.	5.	6.	7.	8.

## 4. Pilha (Stack)

### 4.1. Conceitos e Aplicação

#### - Alocação Estática:

- O espaço de memória é alocado no momento da compilação;
- Exige a definição do número máximo de elementos da Pilha;
- Acesso sequencial. Os elementos ficam consecutivos na memória.

#### - Alocação Dinâmica:

- O espaço de memória é alocado em tempo de execução;
- A Pilha cresce à medida que novos elementos são armazenados, e diminui à medida que elementos são removidos;
- Acesso encadeado: cada elemento pode estar em uma área distinta da memória. Para acessar um elemento, é preciso percorrer todos os seus antecessores na Pilha.

# Referências

EDELWEISS, Nina; GALANTE, Renata. Estruturas de Dados. Porto Alegre, BOOKMAN, 2009.

HEINZLE, Roberto. Estruturas de Dados: implementações com C e Pascal. Blumenau, DIRETIVA, 2006.

TENENBAUM, Aron M. Estrutura de Dados usando C. São Paulo, Makron Books, 1995.

FORBELLONE, André Luiz Villar: EBERSPÄCHER, Henri Frederico. Lógica de Programação: a construção de algoritmos e estruturas de dados. 3. ed. São Paulo, PRENTICE HALL, 2005.

KOFFMAN, Elliot B.; WOLFGANG, Paul A. T. Objetos, abstração, estruturas de dados e projeto usando C++. Rio de Janeiro, LTC, 2008.

PEREIRA, Silvio do Iago. Estruturas de dados fundamentais: conceitos e aplicações. São Paulo, Érica, 1996.

VILLAS, Marcos Viana et al. Estruturas de dados – Conceitos e técnicas de implementação. Rio de Janeiro, Campus, 1993.

VELOSO, Paulo et al. Estrutura de dados. Rio de Janeiro, Campus, 1996.

Canal do Youtube: Linguagem C Programação Descomplicada

Site: <https://programacaodescomplicada.wordpress.com/>

# Obrigado!