

Estruturas de Dados Básicas

- Pilha

Estruturas de Dados



Universidade de Brasília

Departamento de Ciência da Computação

Conceito de Pilha

Uma pilha é uma estrutura de dados simples usada para armazenar dados.

A principal diferença da pilha para outras estruturas semelhantes é a **ordem** de entrada e saída dos elementos/nós da pilha.

Definição de Pilha

Uma pilha é uma lista ordenada na qual a inserção e a exclusão são feitas em **uma** extremidade, chamada *topo*. O **último** elemento inserido é o primeiro a ser excluído.

LAST IN FIRST OUT (LIFO)

FIRST IN LAST OUT (FILO)

Definição de Pilha



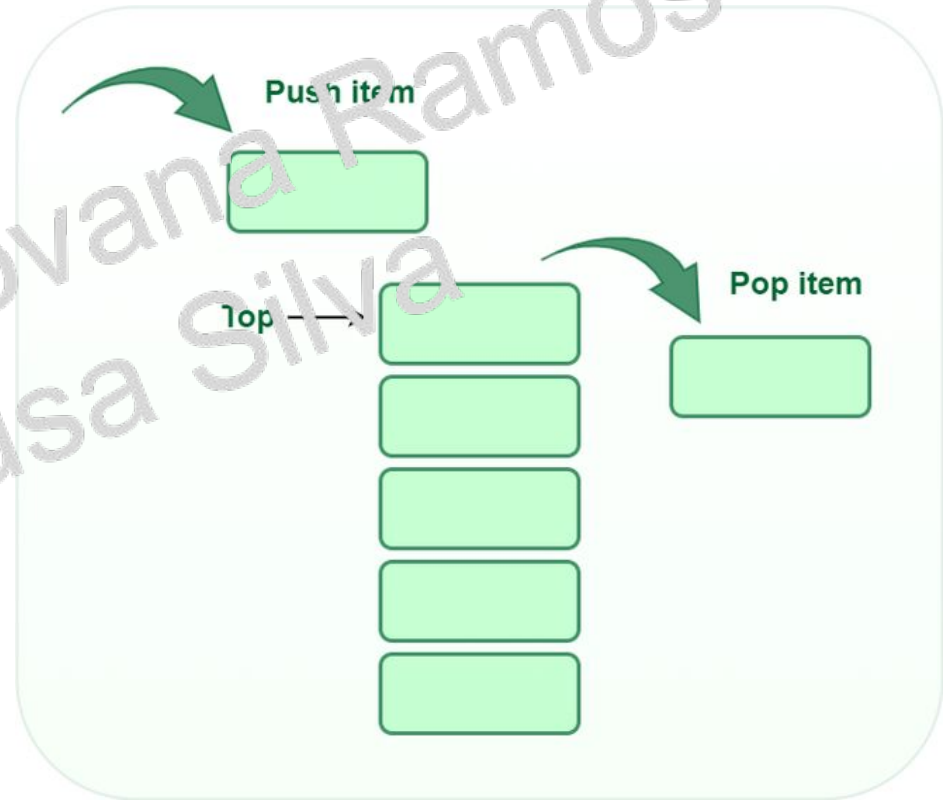
Definição de Pilha



Profa. Geovana Ramos
Souza Silva

Operações de Pilha

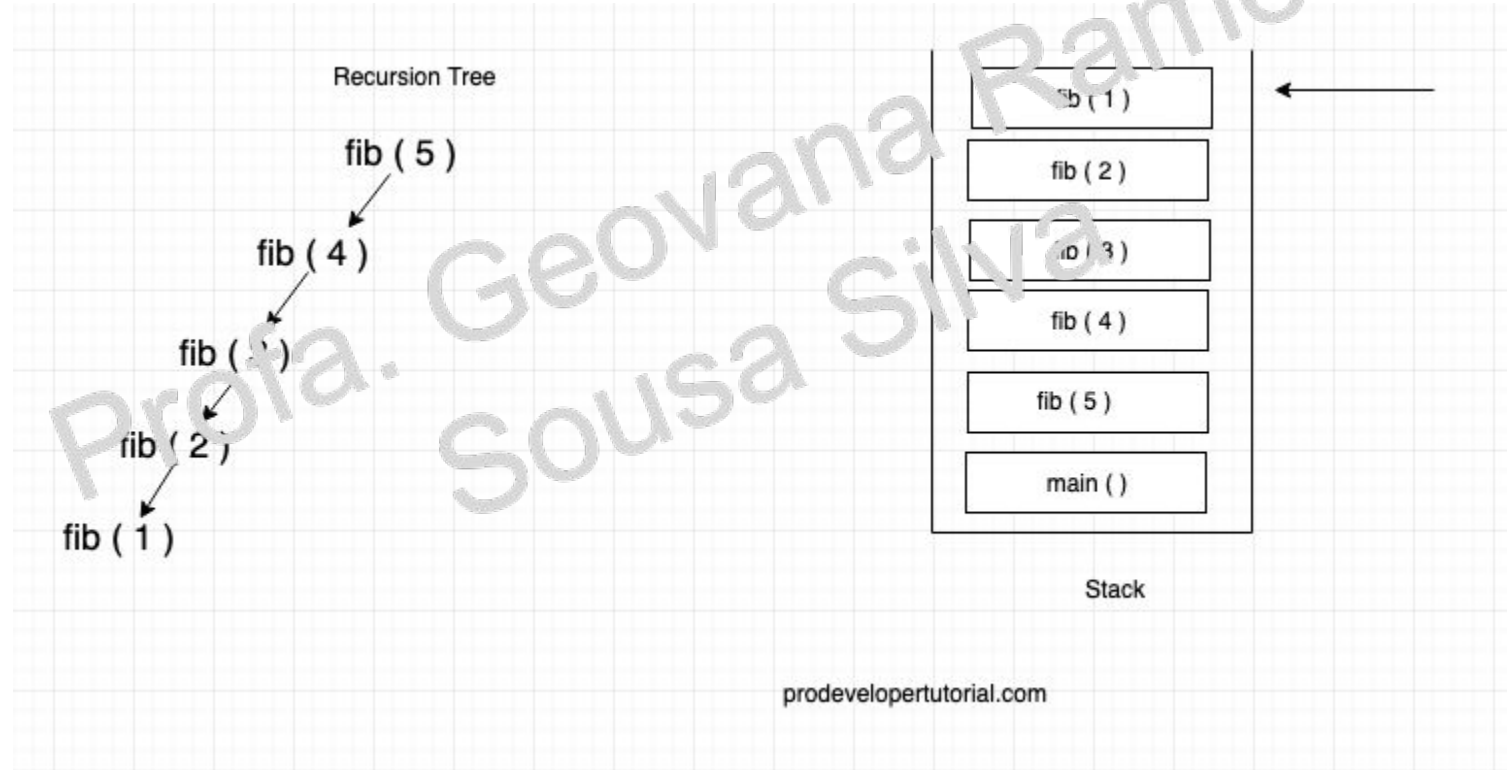
- push (empurrar):
 - inserir na pilha
- pop (estourar):
 - remover da pilha



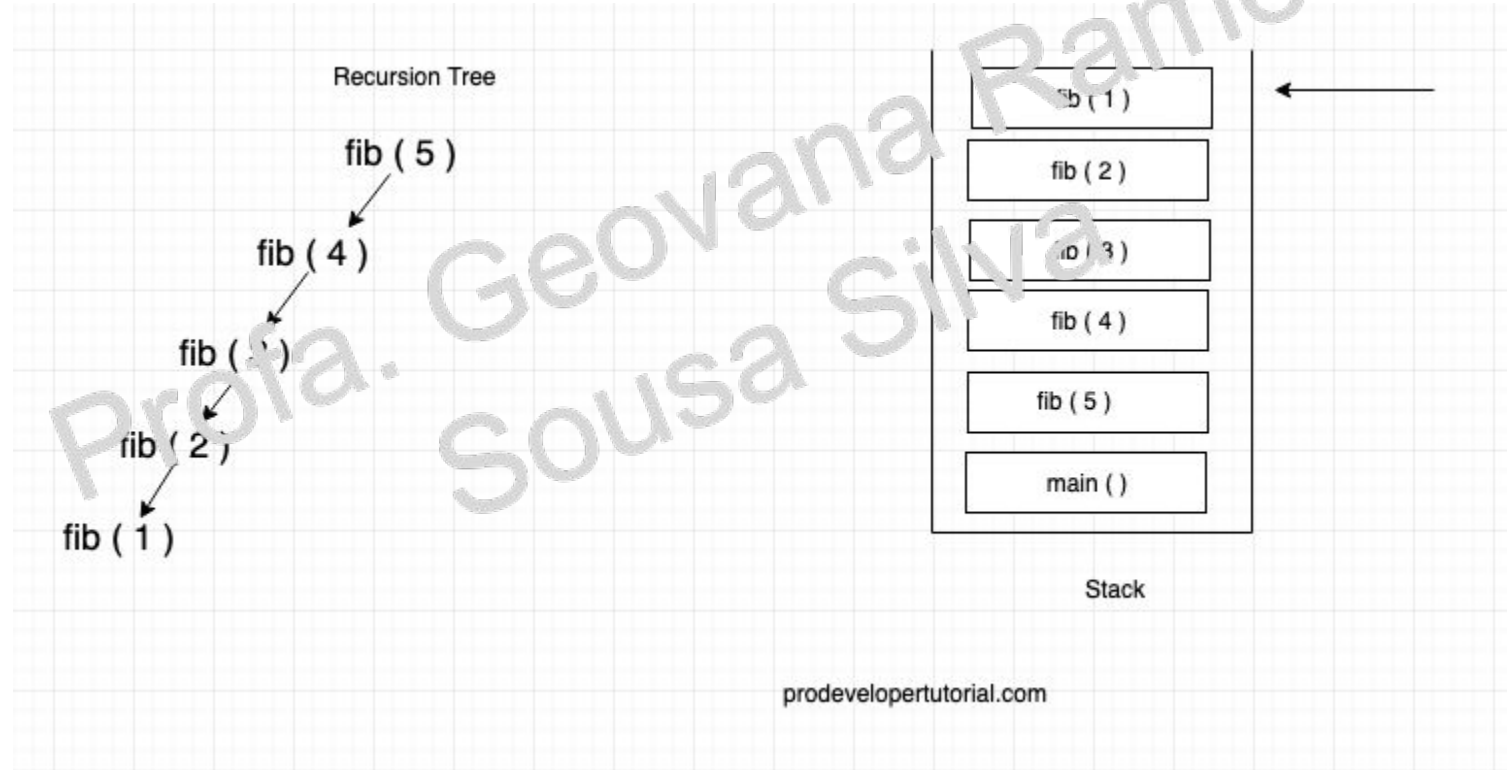
Exemplo de Aplicação

```
def fib(n):  
    if n==1:  
        return 0  
    elif n==2:  
        return 1  
    else:  
        return fib(n-1) + fib(n-2)
```

Exemplo de Aplicação



Exemplo de Aplicação



Implementação

- Array
- Lista encadeada

Profa. Geovana Ramos
Sousa Silva

Implementação como Array

```
class ArrayPilha:
```

```
    def __init__(self):  
        self.itens = []
```

```
    def push(self, item):  
        self.itens.append(item)
```

```
    def pop(self):  
        return self.itens.pop() # último item por padrão
```

Implementação como Lista Encadeada

Considerando as operações abaixo, como implementar uma pilha?

- Inserir no início
- Inserir no meio
- Inserir no fim
- Deletar do início
- Deletar do meio
- Deletar do fim

Implementação como Lista Encadeada

Considerando as operações abaixo, como implementar uma pilha?

- Inserir no início
- Inserir no meio
- Inserir no fim
- Deletar do início
- Deletar do meio
- Deletar do fim

Implementação como Lista Encadeada

Considerando as operações abaixo, como implementar uma pilha?

- Inserir no início
- Inserir no meio
- Inserir no fim
- Deletar do início
- Deletar do meio
- Deletar do fim

Implementação como Lista Encadeada

```
class Pilha:  
    def __init__(self):  
        self.cabeca = None  
  
    def push(self, valor):  
        novo_no = Noh(valor)  
        novo_no.proximo = self.cabeca  
        self.cabeca = novo_no  
  
    def pop(self):  
        removido = self.cabeca.valor  
        self.cabeca = self.cabeca.proximo  
        return removido
```

Curiosidades

- Remoção de uma lista vazia -> erro de *underflow*
- Inserção na pilha cheia -> erro de *overflow*

Profa. Geovana Ramos
Sousa Silva

Para casa

Usando as classes `ArrayPilha` e `Pilha`:

- Faça um método que imprima a pilha.
- Faça um método que verifique se a pilha está vazia.
- Faça um método (`peek`) que retorne o primeiro elemento da pilha (sem deletar).
- Faça um método que inverta a pilha usando *pop* e *push* e retorne uma nova pilha invertida.
- Adicione tratamento de erros para pilha vazia e cheia nos métodos *push* e *pop*. Defina você mesmo um tamanho máximo.

Bibliografia

