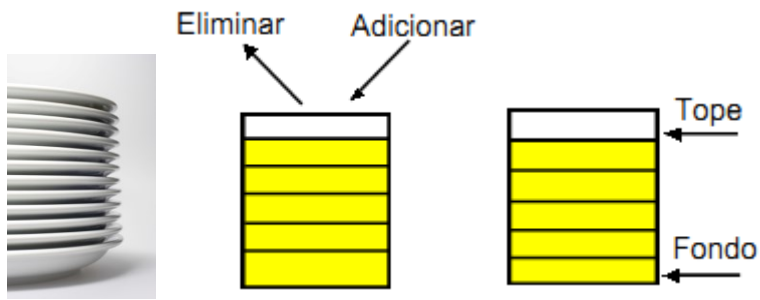


Pilha – Stack



Classe ArrayStack (Pilha implementada por um array (uma lista))

```
# atributos, propriedades ou estados
self._data : armazena os dados da pilha
def __init__(self): construtor. cria e retorna uma pilha vazia
def __len__(self): retorna quantos elementos estão na pilha
def is_empty(self): retorna True se a pilha estiver vazia, else False.
def push(self, e): adiciona o elemento e no topo da pilha
def top(self): retorna o elemento do topo da pilha, mas não o remove
def pop(self): remove e retorna o elemento do topo da pilha
```

```
S = ArrayStack() # cria uma pilha vazia contents: [ ]
S.push(5) # empilha(põe no topo da pilha) o elemento 5 # contents: [5]
S.push(3) # empilha(põe no topo da pilha) o 3 # contents: [5, 3]
print(len(S)) # contents: [5, 3]; outputs 2 (tem dois elementos na pilha)
S.pop() # desempilha(retira do topo da pilha) o 3 # contents: [5];
print(S.is_empty()) # print False, pois, a pilha está cheia # [5];
topo = S.pop() # desempilha o 5 e retorna este valor(fica armazenado na
variável topo) # contents: []; Agora a pilha ficou vazia
print(S.is_empty()) # print True, pois, a pilha está vazia # [];
```

Implementar todas as operações do TAD Pilha utilizando somente os atributos e métodos do TAD Lista com Iterador.

Considerar duas situações:

- o topo é o primeiro elemento da Lista
- o topo é o último elemento da Lista