

LIFO = Last In First Out

Es mucho más fácil usar Double Linked List que una lista, ya que no tendremos problemas en la memoria, copiando todos los valores.

<https://docs.python.org/es/3/library/collections.html>

***CLASS COLLECTIONS.DEQUE([ITERABLE [, MAXLEN]])***

Los deques son O(1), ya que las función appends y pops son eficientes desde cualquier lado. La diferencia con las list es que incurren en 0(n) y que cambian de capacidad dependiendo de la cantidad de datos.

from collections import deque

stack = deque ()

stack.append("https://www.cnn.com/")

stack.append("https://www.cnn.com/world")

stack.append("https://www.cnn.com/india")

stack.append("https://www.cnn.com/china")

print(stack)



stack.pop()

stack.pop()

stack.pop()

print(stack)



Lo siguiente es una función, no es necesario hacer esto, ya que se puede usar las DEQUE por si solas tal cual.

class Stack:

    def \_\_init\_\_(self):

        self.container = deque()

    def push(self, val):

        self.container.append(val)

    def pop(self):

        return self.container.pop()

    def peek(self):

        return self.container[-1]

    def is\_empty(self):

        return len(self.container) == 0

    def size(self):

        return len(self.container)

Version 2.0 del código de invertir una cadena de texto:

from collections import deque

stack = deque()

class Stack:

    def \_\_init\_\_(self):

        self.container = deque()

    def push(self, val):

        self.container.append(val)

        print(self.container)

    def reverse\_string(self, val):

        self.push(val)

        for str in self.container:

            print(str[::-1])

s = Stack()

s.reverse\_string("91-DIVOC someratsiuqnoC")