

STACK

Nature and Implementation

Martzel P. Baste



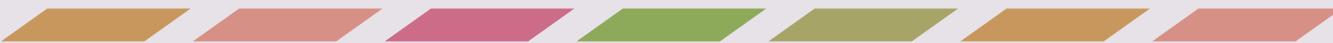
Intended Learning Outcomes

At the end of the lesson, the students are expected to:

Explain the definition and importance of Stack Algorithm



Identify Stack basic operations



Implement ADT Stack using array-based structure



Examine the application of ADT Stack



Stacks: Nature

- It is a linear list in which all additions and deletions are restricted to one end called **TOP**.
- **LAST-IN FIRST-OUT** (LIFO)

- Examples:
 - Stack of Coins
 - Stack of Dishes



Why Study Stacks?

Stack applications can be classified into four broad categories:

- Reversing data
- Parsing data
- Postponing data usage
- Backtracking steps



Reversing Data

- Reversing data requires that a given set of data be reordered so that the first and last elements are exchanged.
- Example:

Reverse a list of numbers

► Input : 1 2 3 4 5

► Output : 5 4 3 2 1

Parsing

- ▶ Parsing is any logic that breaks data into independent pieces for further processing.
- ▶ Example: To translate a source program into machine language, a compiler must parse the program into individual parts such as keywords, names, and tokens
- ▶ Example: parsing parenthesis () - to make sure all opening-closing parentheses are paired

Postponement

► Often the logic of an application requires that the usage of data be deferred until some later point

► Examples:

- Infix to Transformation
- Postfix evaluation

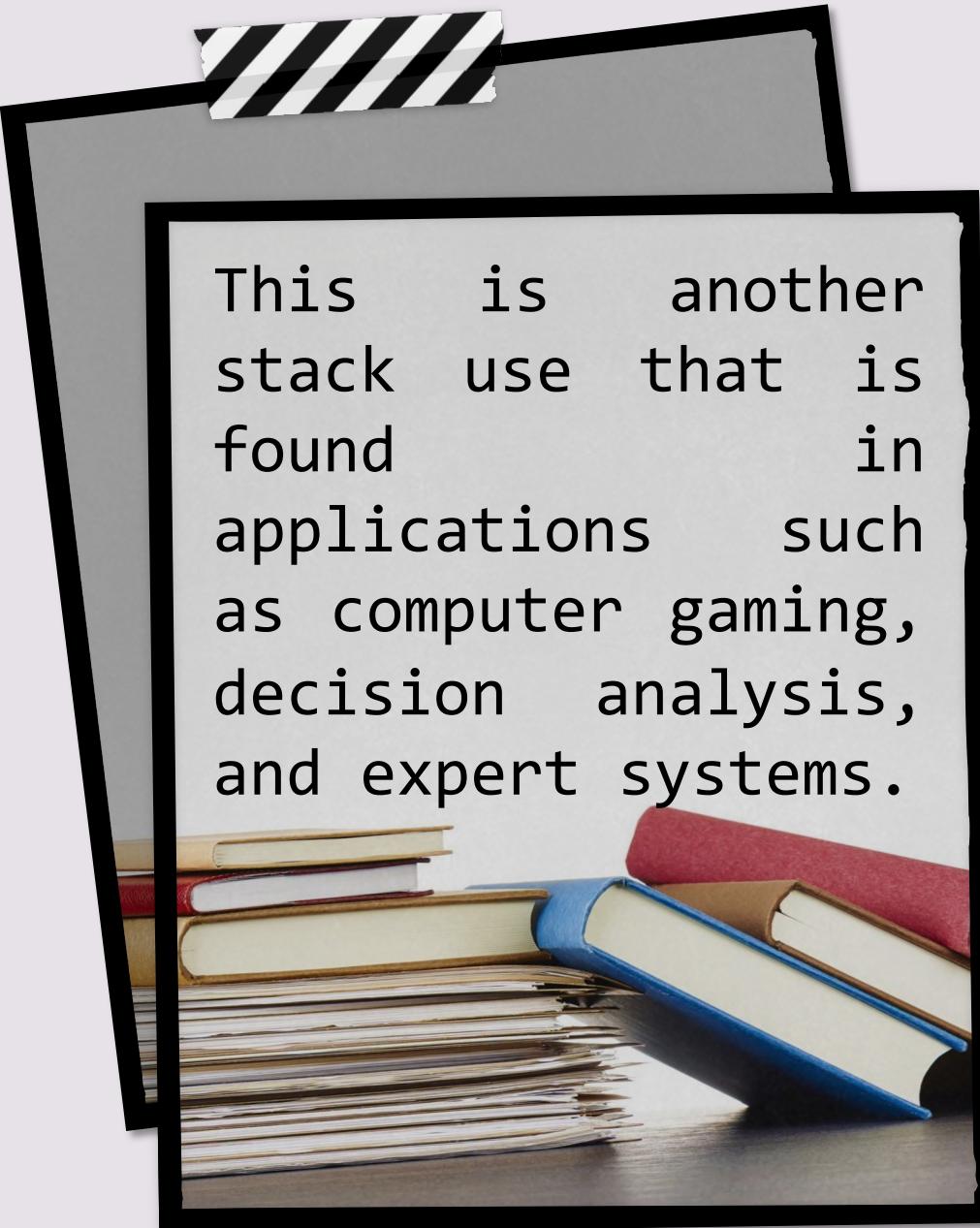
Postfix

► Prefix : + ab

► Postfix : ab +

► Infix : a + b





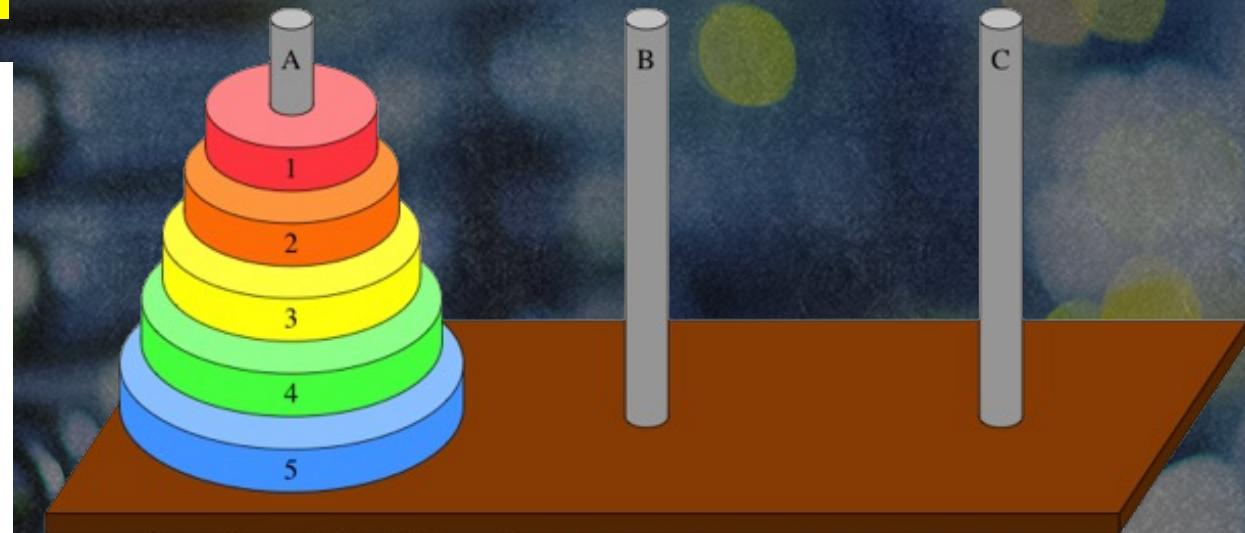
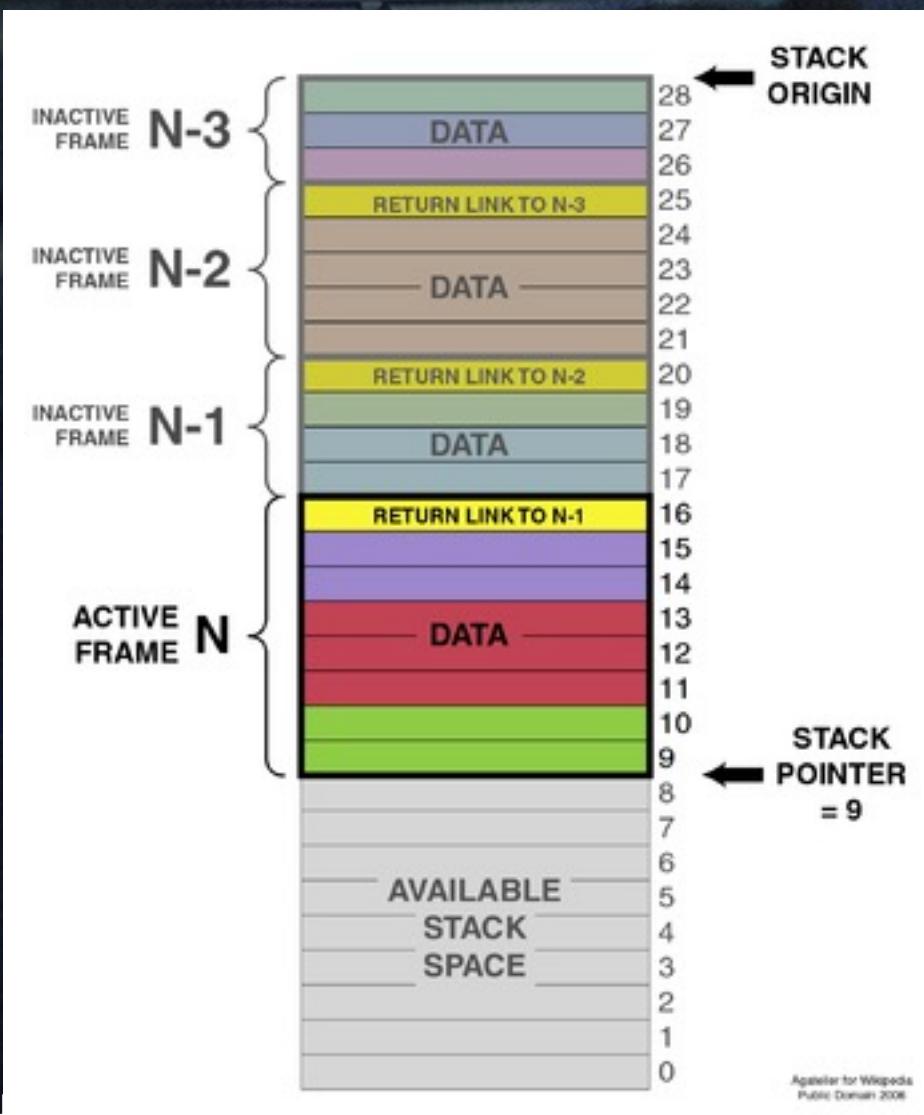
This is another stack use that is found in applications such as computer gaming, decision analysis, and expert systems.

Backtracking

Goal seeking

Eight queens problem

Other Applications



<https://www.khanacademy.org/computing/computer-science/algorithms/towers-of-hanoi/a/towers-of-hanoi>

<https://www.mathsisfun.com/games/towerofhanoi.html>

Basic Operation



Push

Adds an item at the **top** of the stack



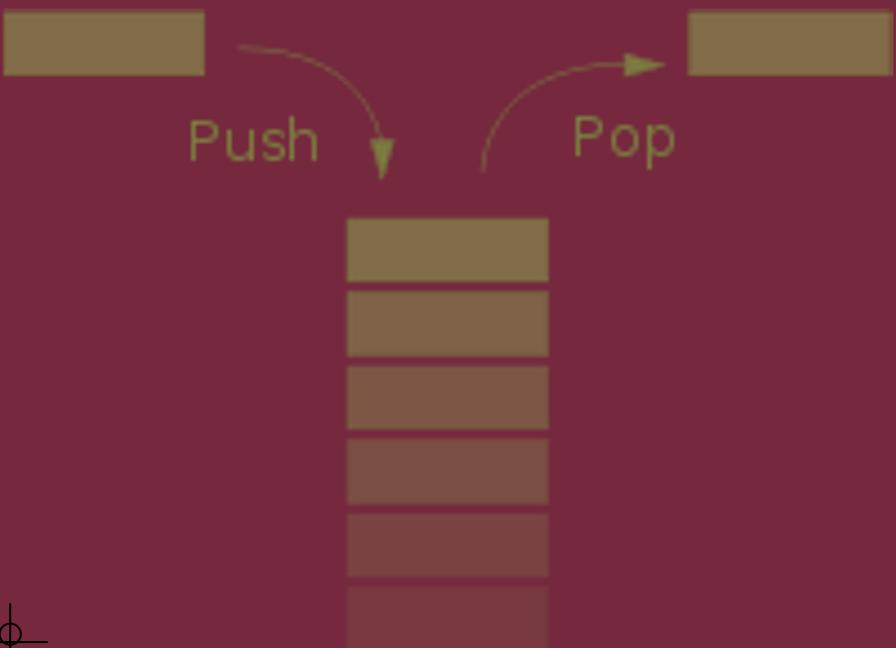
Pop

Removes an item at the **top** of the stack and returns it to the user

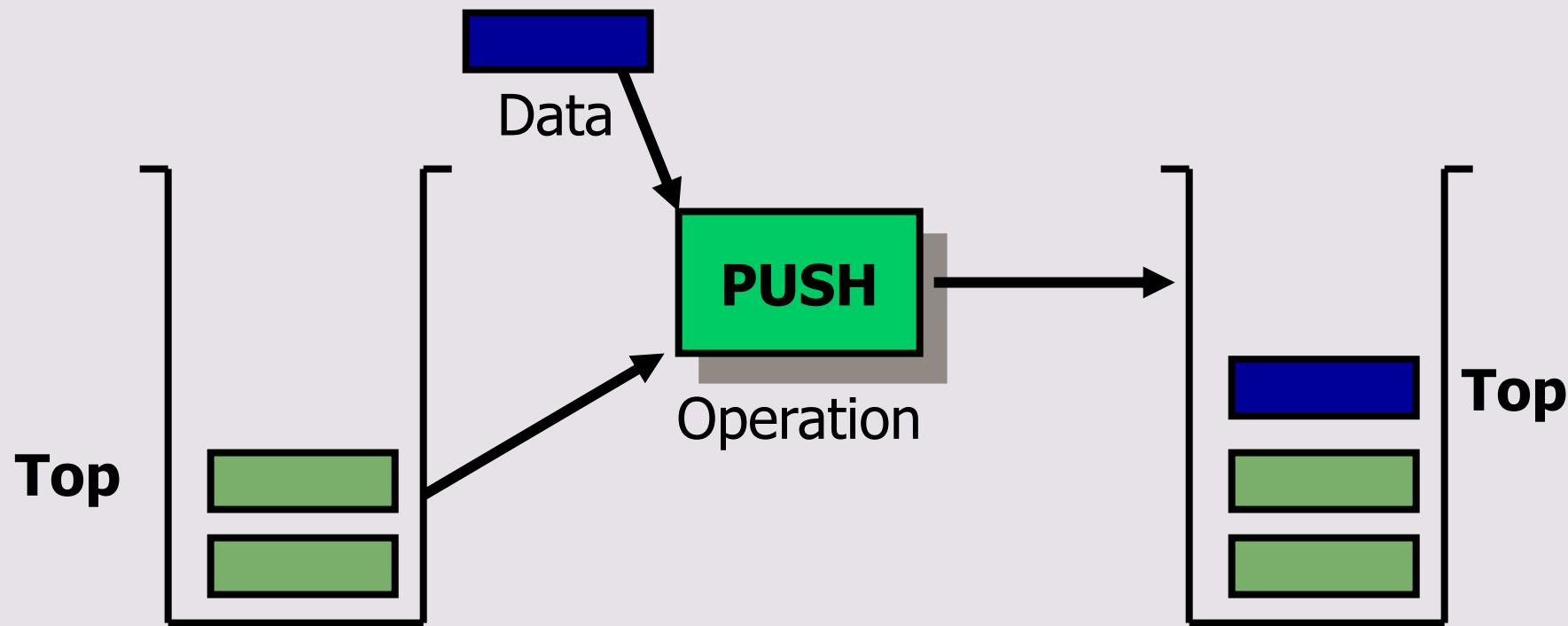


Stack Top

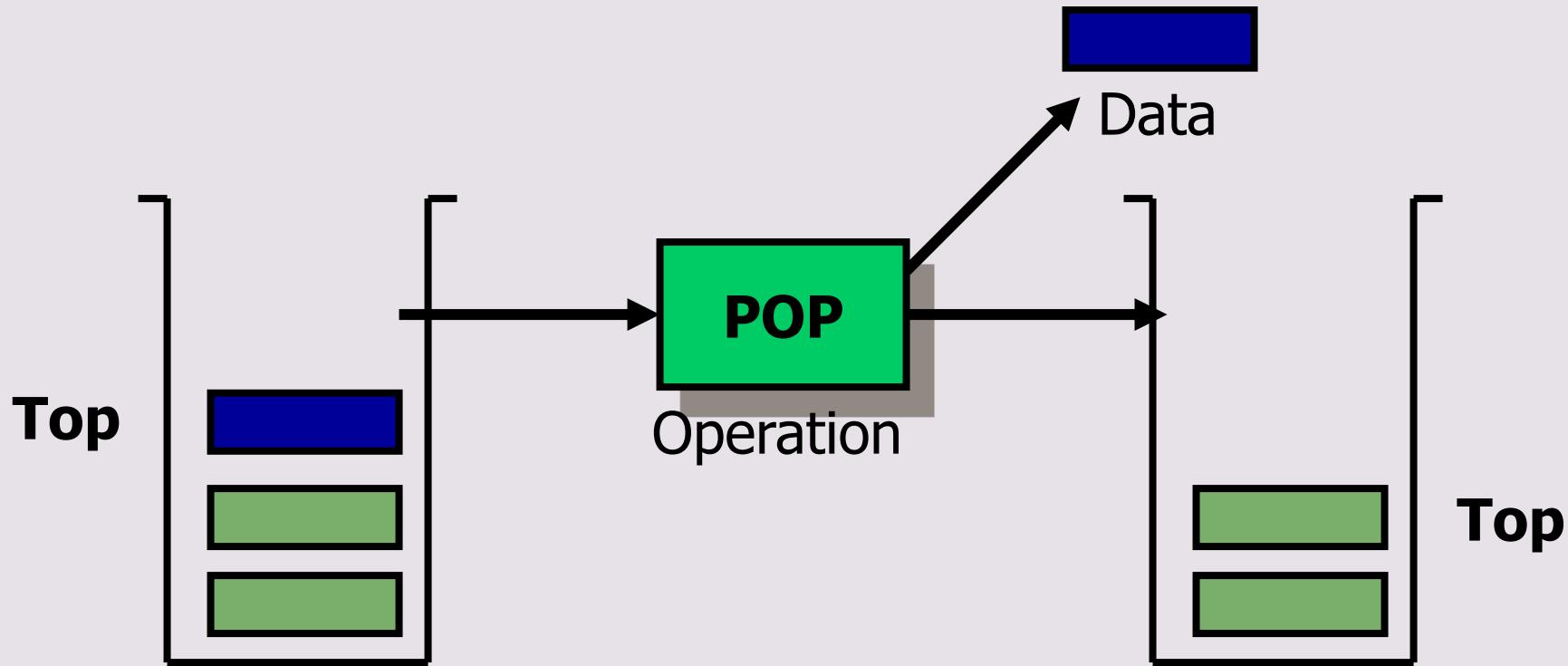
Copies the item at the **top** of the stack. It returns to the user the data in the **top** element but does not delete it.



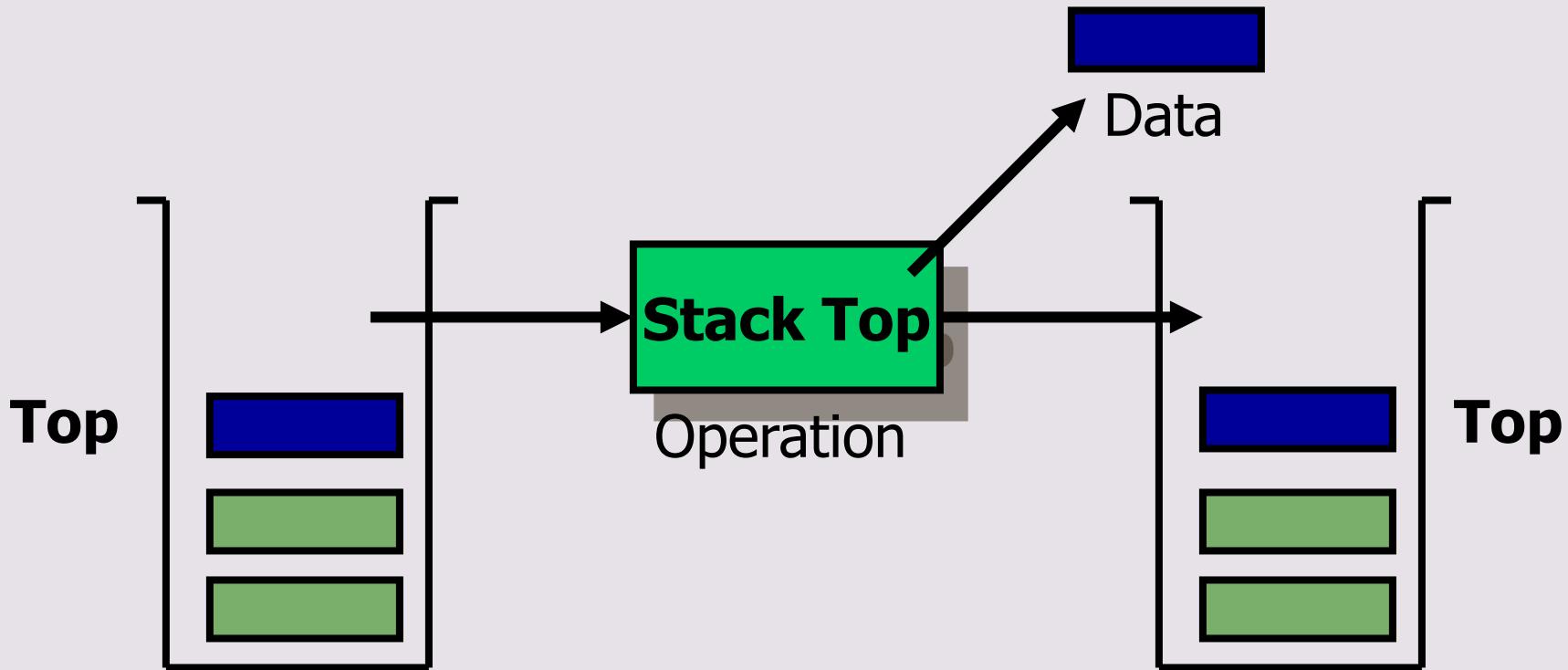
PUSH *operation*



POP Operation



STACKTOP Operation



Properties

Adding
element: $O(1)$

Accessing
Time: $O(n)$
[Worst Case]



deleting
element: $O(1)$

Only one end
allows
inserting and
deleting an
element.



THANKYOU!

