# **Data Structures and Algorithms**

**Rao Muhammad Umer** Lecturer, **CS** and **IT Department**, The University of Lahore. Web: <u>raoumer.github.io</u>

**Data Structure and Algorithms** 



1

# outline

## **Stacks**

What are Stacks?	Application
Operations on Stack	Towe
<ul> <li>Push</li> </ul>	Evalu
■ Рор	■ Ir ■ Ir
Stack as an Array	■ P
Stack as a Linked List	■ P
	- C.



**Data Structure and Algorithms** 

### **Applications of Stacks**

- ers of Hanoi
- uation of arithmetic expression
- nfix to Prefix Conversion
- nfix to Postfix Conversion
- Postfix to Prefix Conversion
- Postfix to Infix Conversion
- Evaluation of Postfix Expression

# Stack

- **Stack** is a data structure in which addition of new element or deletion of an existing element always takes place at the same end i.e. top of stack.
- ADT (Abstract Data Type)
- Example:
  - Stack of dishes in a cafeteria
- Also known as LIFO (Last In First Out).





# **Pictorial Representation of Stack**





https://en.wikibooks.org/wiki/Data\_Structures/Stacks\_and\_Queues

# **Operations on Stack**

- Push: Allows adding an element at the top of the stack.
- Pop: Allows to remove an element from the top of the stack.
- See <u>Animation</u> for working of a stack



# **Representation of Stack**

### Stack as an Array

## Stack as a Linked List



**Data Structure and Algorithms** 

6

# **Stack as an Array**



See <u>source code in C++</u> for Stack as an Array 



**Data Structure and Algorithms** 



7

# **Stack as a Linked List**



See <u>source code in C++</u> for Stack as a Linked List 





# **Stack Limitations**

**Remember:** You can't loop over a stack like you do a list. 



- Instead, you pull contents out of the stack to view them.
  - Idiom: Remove each element until the stack is empty.

while (s != NULL) do something with s.pop();

# **Applications of Stacks**

**Programming languages:** 

> Method (or Function) calls are placed onto a stack (call=push, return=pop)

- Matching up related pairs of things:
  - find out whether a string is a palindrome
  - examine a file to see if its braces { } and other operators match
- Sophisticated algorithms:
  - searching through a maze with "backtracking"
    - many programs use an "undo stack" of previous operations



# **Applications of Stacks**

- Towers of Hanoi
- Evaluation of arithmetic expression
  - Polish Notation
    - Infix to Prefix Conversion
    - Infix to Postfix Conversion
    - Postfix to Prefix Conversion
    - Postfix to Infix Conversion
    - Evaluation of Postfix Expression



# **Tower of Hanoi**

## Tower of Hanoi

- See <u>Animation</u> of Tower of Hanoi
- See <u>source code in C++</u> of Tower of Hanoi





# **Infix to Prefix Conversion**

Infix Expression:

■ e.g. A + (B / C) – D, A + B \* C, etc.

- Prefix Expression:
  - A + B (infix)  $\rightarrow$  + AB (prefix)
- Postfix Expression:
  - A + B (infix)  $\rightarrow AB + (postfix)$
- Infix to Prefix Conversion
  - See <u>Animation</u> of Infix to Prefix conversion
  - See <u>source code in C++</u> of Infix to Prefix conversion

# **Infix to Postfix Conversion**

## Infix to Postfix Conversion

- See <u>Animation</u> of Infix to Postfix conversion
- See <u>source code in C++</u> of Infix to Postfix conversion



# **Postfix to Prefix Conversion**

- Postfix to Prefix Conversion
  - See <u>Animation</u> of Postfix to Prefix conversion
  - See <u>source code in C++</u> of Postfix to Prefix conversion



**Data Structure and Algorithms** 

## า onversion

# **Postfix to Infix Conversion**

## Postfix to Infix Conversion

- See <u>Animation</u> of Postfix to Infix conversion
- See <u>source code in C++</u> of Postfix to Infix conversion



# **Evaluation of Postfix Expression**

- Evaluation of Postfix Expression
  - See <u>Animation</u> of evaluation of Postfix expression
  - See <u>source code in C++</u> of evaluation of Postfix expression



**Data Structure and Algorithms** 

ession fix expression

## Exercise

- Write a method symbolsBalanced that accepts a String as a parameter and returns whether or not the parentheses and the curly brackets in that String are balanced as they would have to be in a valid C++ program.
  - Use a Stack to solve this problem.



# Acknowledgement

- Mostly Slides taken from Book: "Data Structures through C++" by Yashavant P. Kanetkar
- https://en.wikibooks.org/wiki/Data\_Structures/Stacks\_and\_Queues

