Stack

Get ready

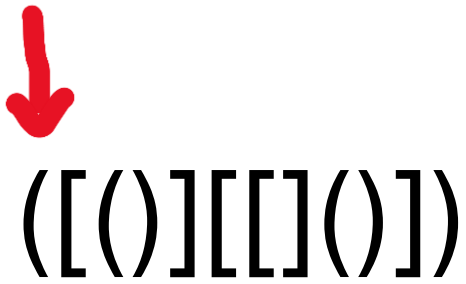
- Some students yet to have their midterm and I haven’t started grading yet, So let’s not discuss about it today.

- We will discuss about Midterm and solution and hopefully You will get your graded work on Monday next week.

- Let’s open Eclipse and prepare to learn for Midterm 2.

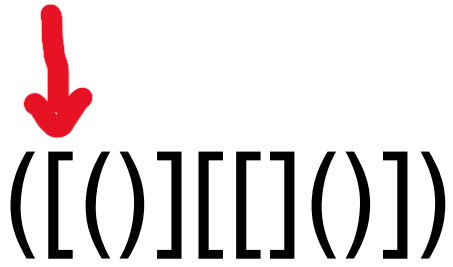
Example of Stack implementation

- test if a string composed of parentheses and square brackets is properly matched.
- Balanced parentheses: when every opening has closing in a corresponding order.



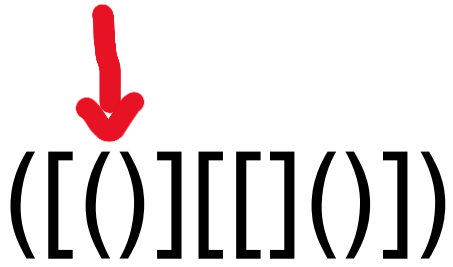
Example of Stack implementation

- test if a string composed of parentheses and square brackets is properly matched.



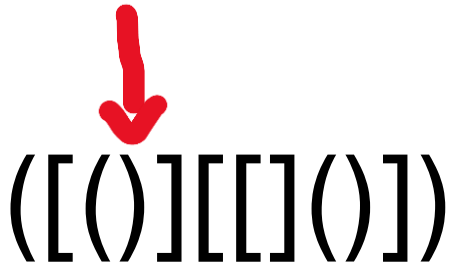
Example of Stack implementation

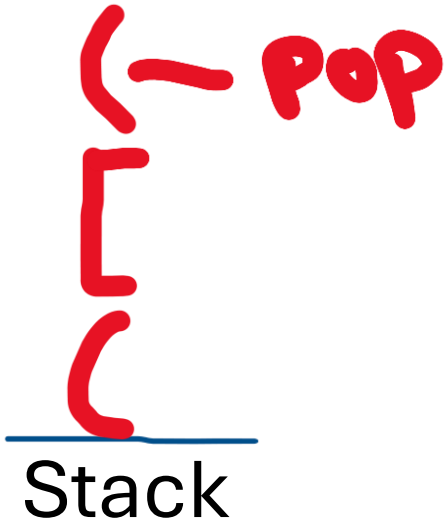
- test if a string composed of parentheses and square brackets is properly matched.



Example of Stack implementation

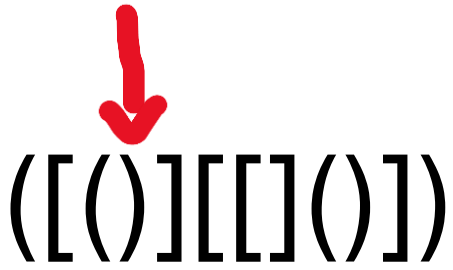
- test if a string composed of parentheses and square brackets is properly matched.





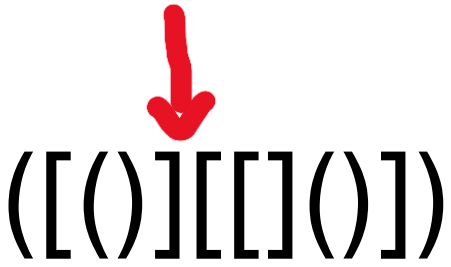
Example of Stack implementation

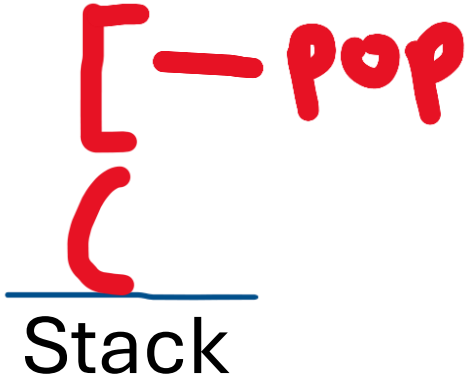
- test if a string composed of parentheses and square brackets is properly matched.



Example of Stack implementation

- test if a string composed of parentheses and square brackets is properly matched.





Try it yourself.....

Infix, prefix and postfix

- Infix $(a + b) * (c + d)$
- Prefix $* + ab + c$d
- Postfix $ab + cd + *$

- Why to use Pre and Post fix?(easier for computer to process, no need to worry about PEMDAS rule)

Practice

- Evaluate the following prefix notation expressions (convert to infix)
- + 4 + 2 3
- + * 5 5 + 2 2
- + * * 2 2 3 5

Infix
- 4+(2+3)
- (5*5)+(2+2)
- 2*2*3+5

Now make the converted infix to postfix

Postfix
- 423++
- 55*22++
- 22*3*5+

From Infix to Postfix

- Infix
  - (2+3) +4
  - (2+2)+(5*5)
  - 2*2*(3+5)

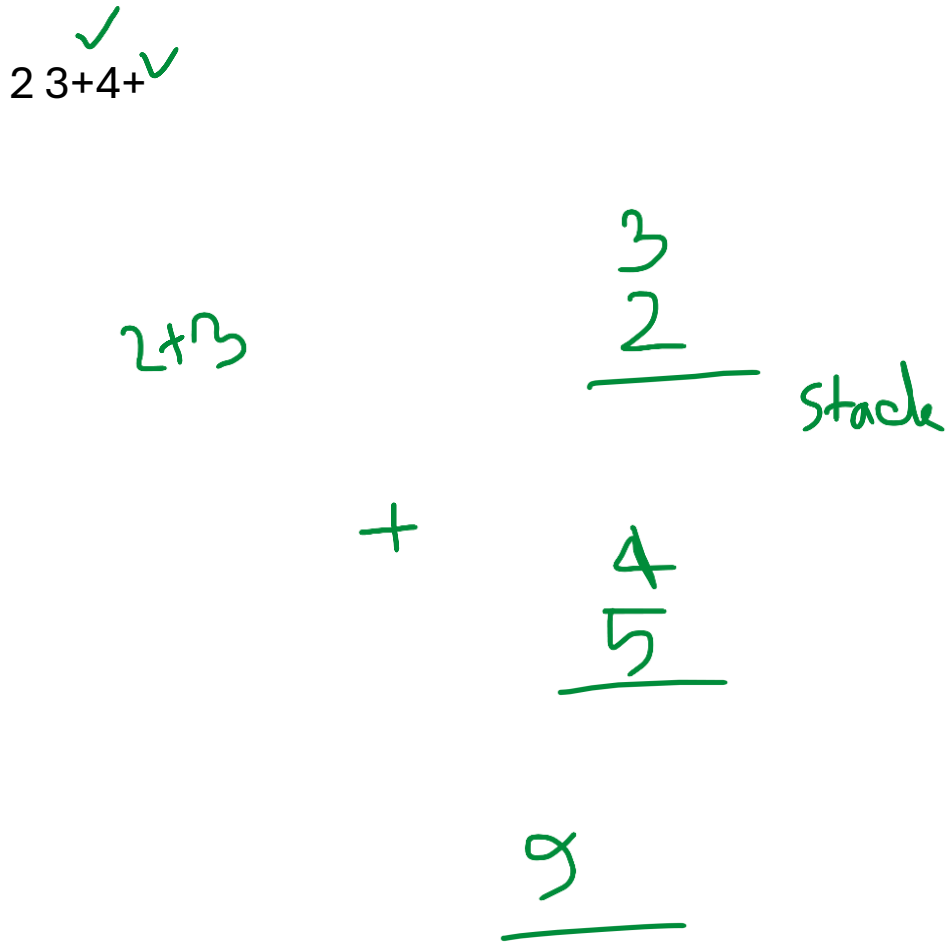
  Postfix
- 2 3+4+
- 2 2 + 5 5 ∗ +
- 2 2 * 3 5 +*

Infix to prefix for special operations

- Exponentiation
  - Example 2^{3^2} , we usually write it as 2^3^2 or 2^(3^2) – infix
  - Prefix: ^2^32
  - Postfix: 232^^
- Logarithm
  - log(2 + 3) –infix
  - Prefix: log+23
  - Postfix: 23+log
- Trigonometry function
  - Infix: sin(x) * cos(y)
  - Prefix: * sin x cos y
  - Postfix: x sin y cos *

Evaluating PostFix using a stack

- Process the input string from left to right, separating it into tokens that represent operators and operands as you go.
- When you encounter an operand, push it on top of the stack.
- Upon reading an operator, pop the top two operands from the stack, apply the operator, and then push the result back onto the stack.



Conversion of a char digit to int digit

Way1:

```
 char digitChar = '7';
 int digitInt = digitChar - '0'; // digitInt will be 7
```

Way2:

```
 char digitChar = '3';
 int digitInt = Character.getNumericValue(digitChar); // digitInt will be 3
```

Way3:

```
 char digitChar = '9';
 int digitInt = Integer.parseInt(String.valueOf(digitChar)); // digitInt will be 9
```

Sample problems for practice at home

- Find Duplicate parenthesis in a string
- Finding a palindrome using Stack
- https://medium.com/techie-delight/stack-data-structure-practice-problems-and-interview-questions-9f08a35a7f19

- https://www.geeksforgeeks.org/dsa/top-50-problems-on-stack-data-structure-asked-in-interviews/

Input and Output from and to a file

- Input from a file
- There are others ways too.
- https://www.geeksforgeeks.org/different-ways-reading-text-file-java/

```java
import java.io.*;
class Main {
  public static void main(String[] args) throws Exception{
    // Passing the path to the file as a parameter
    FileReader fr = new FileReader("file.txt");

    // Declaring loop variable
      int i;
      // Holds true till there is nothing to read
      while ((i = fr.read()) != -1)

        // Print all the content of a file
        System.out.print((char)i);
 }
}
```

Output to a file

- Do it yourself using
- FileWriter class

- There are others ways too https://www.geeksforgeeks.org/java-program-to-write-into-a-file/