

Stack Applications: parenCheck

Ideas generated from
JanaH's "IDEA PARTY"

Test 1: matching parens

$(2 + (3 + 4) + (5 + 6))$

Stack Trace

(~~X~~ → C ~~X~~ → C ~~X~~ empty stack

Test 2: mismatched - extra open

$((2 + 3) + (4 + 5))$

((~~X~~ → ((~~X~~ → (leftover items on stack

Test 3: mismatched - extra close

$(2 + 3))(4 + 5)$

(~~X~~ → empty can't pop!
so... stop!

```
While (index < length of string)
  if (open paren)
    push open parens to stack
  else if (close paren)
    if (stack not empty)
      pop close parens to stack
    else
      return false
```

return stack.isEmpty()

Stack Applications: is Palindrome

Case 1: racecar length: 7 mid: 3

r a c
 r a r

(even)
case 2: $p \overset{0}{0} \overset{1}{:} \overset{2}{0} \overset{3}{p}$ length: 4 mid: 1.5
* use 1

- use int division to calculate middle $(\frac{\text{last index}}{2})$
- push chars from index: 0 \rightarrow index: middle
- if odd, pop one off to balance
- for loop: index: middle+1 \rightarrow index: length-1
 - if pop NOT EQUAL char @ index
 - return false
- return results of checking empty stack