

Stacks / Queues  
Recursions  
Complexity Analysis

⇒ Easily implement the stack as a SLL.

↓  
top() → c

→ All operations done in constant time.

Stacked list? Limitation → Strength

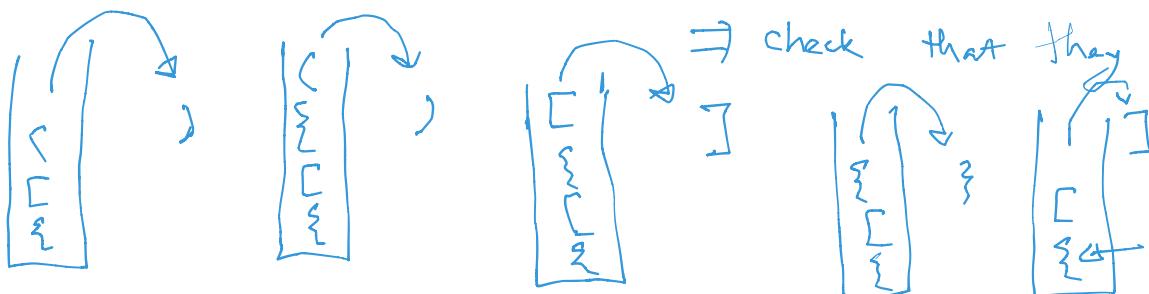
Eg. Match parentheses

{ [ ( ) { ( ) [ ] } ] ( ) }

\* Scan through the string char by char

\* Left paren → Push onto a stack  
{ [ (

\* Right paren → Pop its counterpart off  
{ ] ) the stack.



⇒ check that they match.

(1) Mismatch on pop

(2) Try to pop from an empty stack

(3) Get to the end of the string and stack is NOT empty.

⇒ Params NOT Matched.