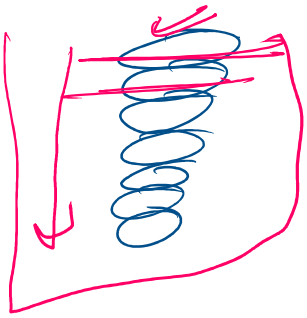
3.10 - Stacks - 1

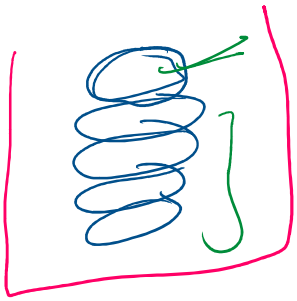
Sunday, July 27, 2025 10:41 AM

Stack

Stack of plates

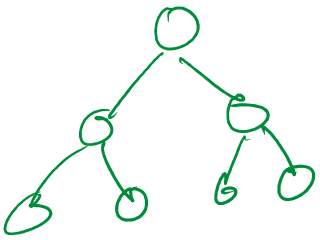


Data structure

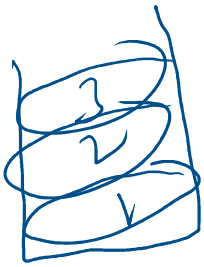


⇒ Linear DS: Arrays, String & LL, Array list, stacks, Queues.

⇒ Non Linear DS: Trees, Graph



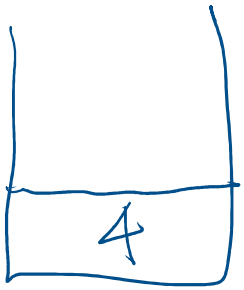
⇒ LIFO → Last In First Out



Basic operation of Stack

⇒ push(): insert

⇒ pop(): removes the top.



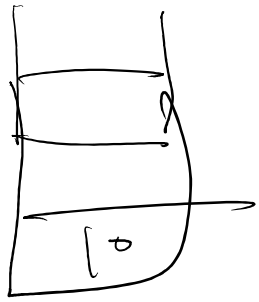
Last element

(A) In stack, we only have the access to top.

⇒ push 10
⇒ push 7
⇒ push 8

⇒ pop()
⇒ push 11
⇒ pop()
⇒ pop()

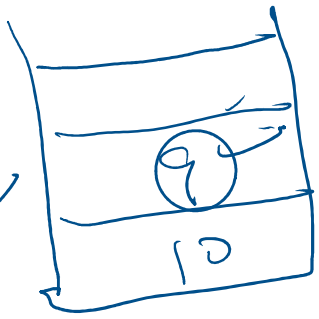
8, 11, 7



⇒ peek(): returns the top element. Doesn't remove the top.

Q) → push 10
→ push 9
→ push 8
→ pop()
→ peek()

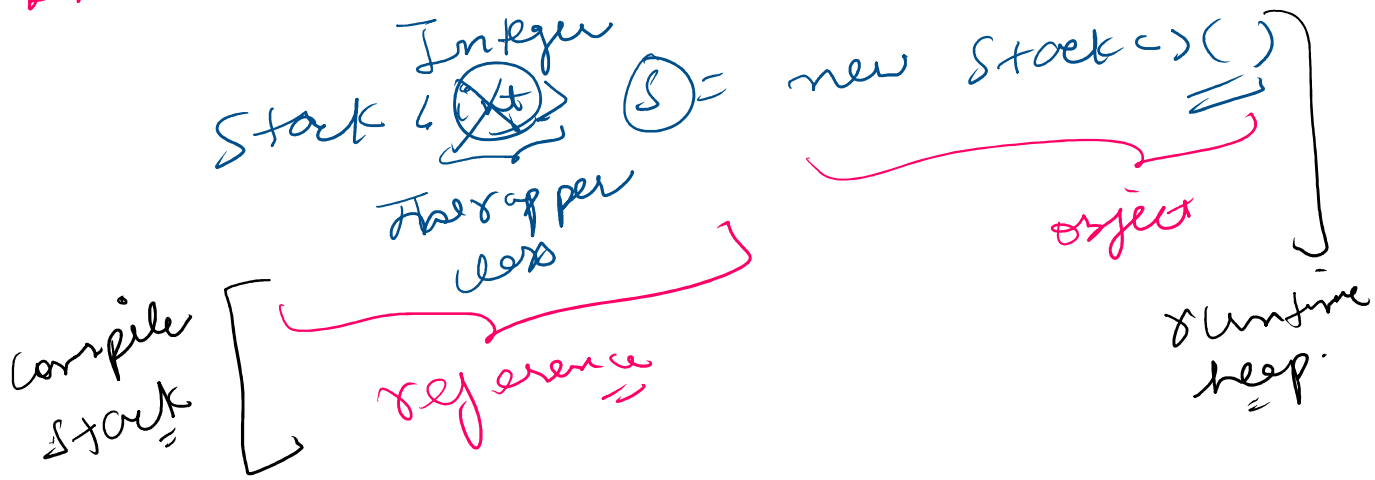
→ push 12
→ peek() ✓ 8 9
→ pop() ✓ 12 12
→ peek() ✓ 9



Show off knowledge

Internally, Stack in Java is implemented using [vectors] resizable arrays

⇒ Inbuilt class called Stack.



⇒ Complexity of push(), pop(), peek()

↳ O(1)

(*) pop() and peek() on empty stack will result in EmptyStackException.

(*) while solving the questions, before peek() or pop(), always check whether the stack is empty or not.

Think about stack in which type of problems

Think about stack in which type of problems

⇒ Incoming Parenthesis ( )

⇒ Span of an array.

Extra Brackets

⇒ ((a+b))

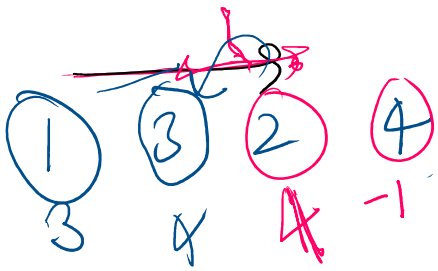
⇒ (a+((a+b))

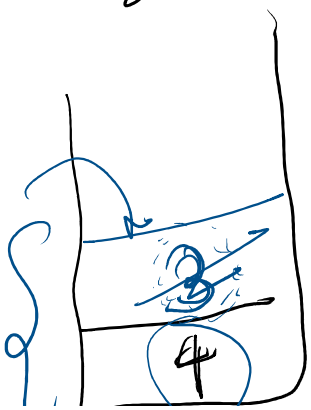
⇒ (a+((a+b)))

(a+ε) ⇒

⇒ A pair of Bracket is extra when there is no expression b/w them.

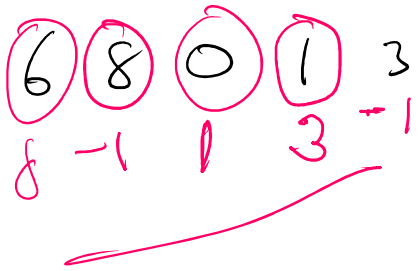
Next greater Element to the Right

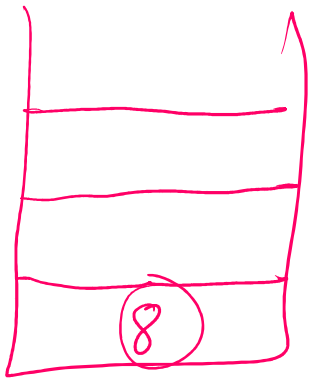




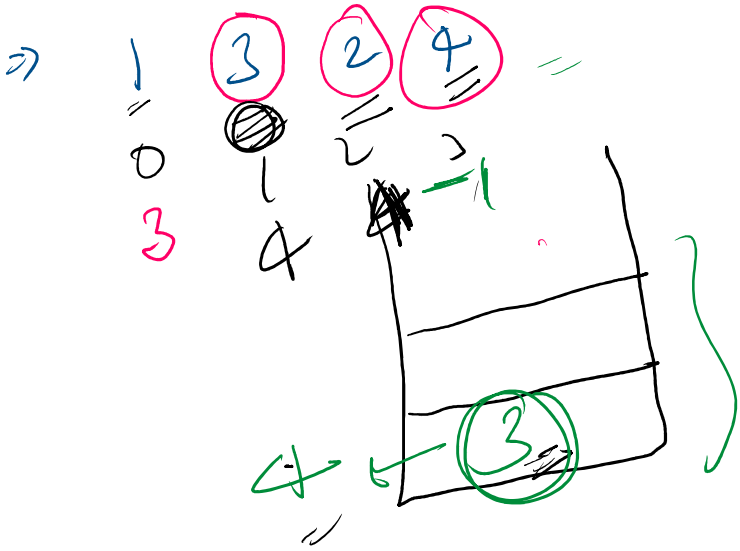
potentials

potentially ngl

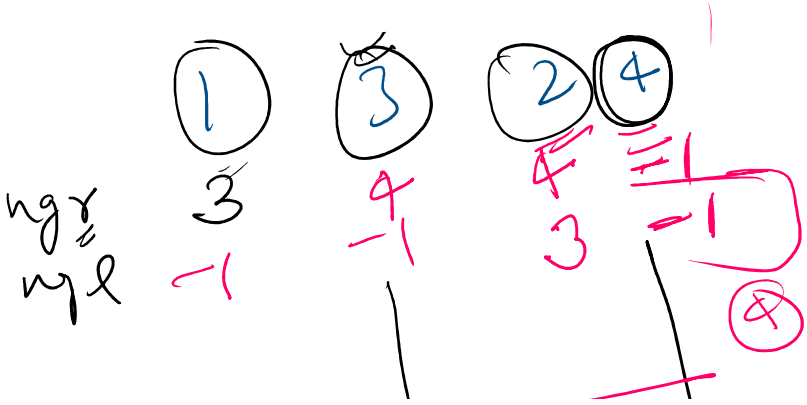


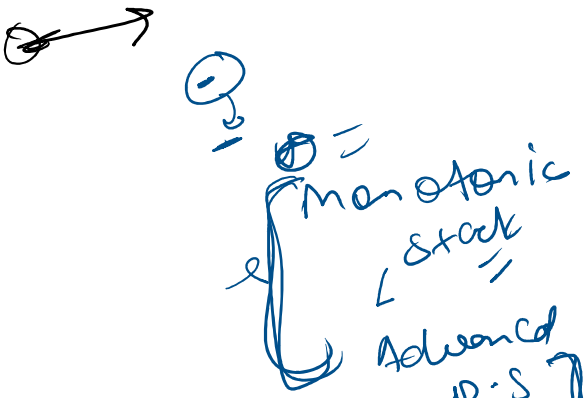


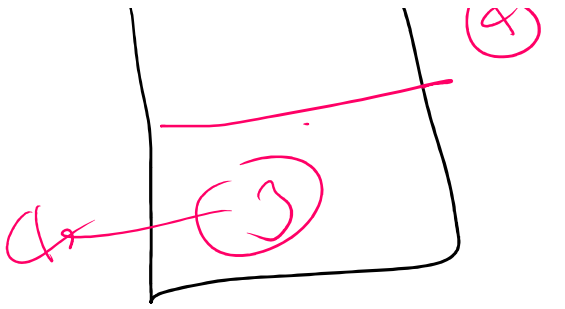
Approach2 (V. Imp)



elements looking for their ngl







Advanced D.S

Span