Stack
Sunday, 12 March 2023 5:50 PM

Introduction to stacks.

→ What is a Stack ?

Stack is a lineas data structure that follows a Particulas ordes in which the operations are performed.

In othes words, a stack is a simply a list with restriction that insertion and deletion can be performed only from one end, called the top.

The ordes that a stack follows is LIFO (Last in first out)

→ stack as an ADT (Abstract data type)

What is ADT?
→ when we define a data structure as ADT, we only focus on the operations that data structure provides, we dont go to the implementation details.

ation

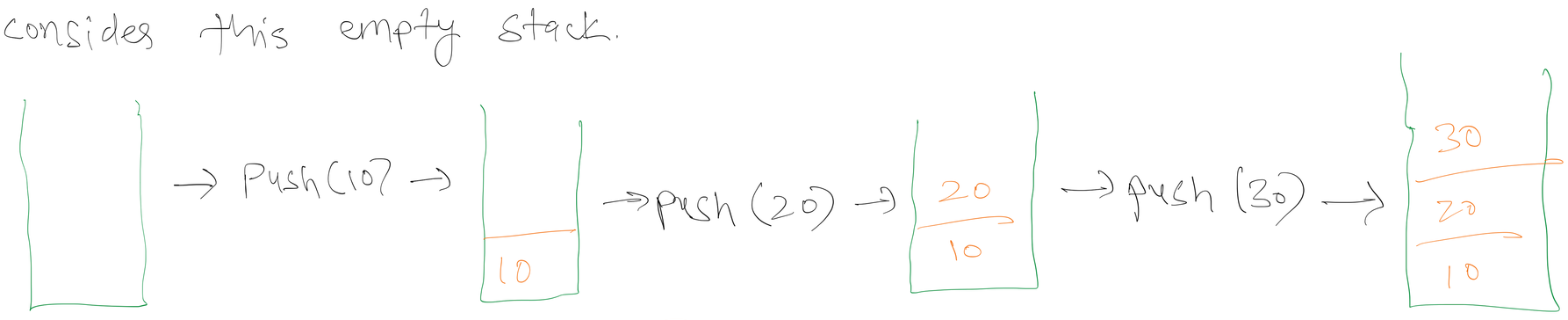
① Push(x)
↳ push an element on top of the stack.

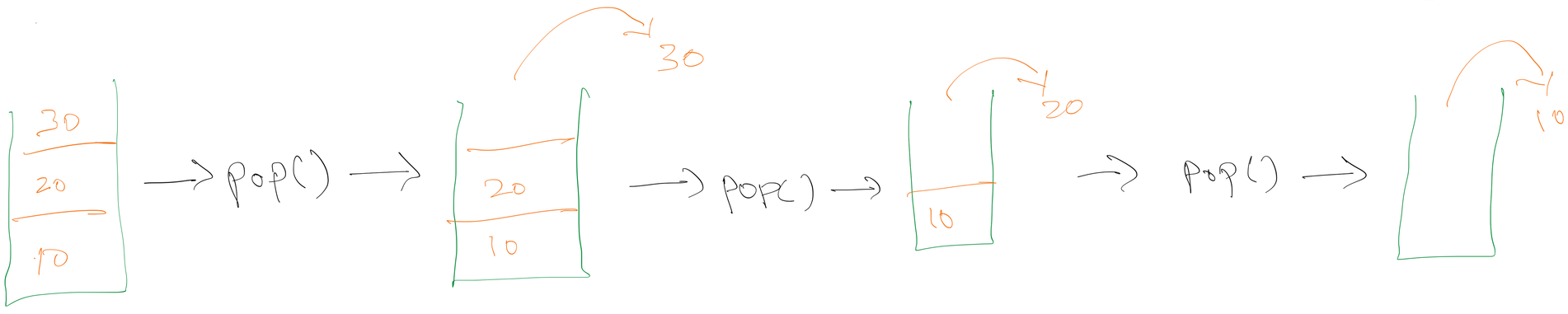
② Pop()
↳ pop / delete the current top element.

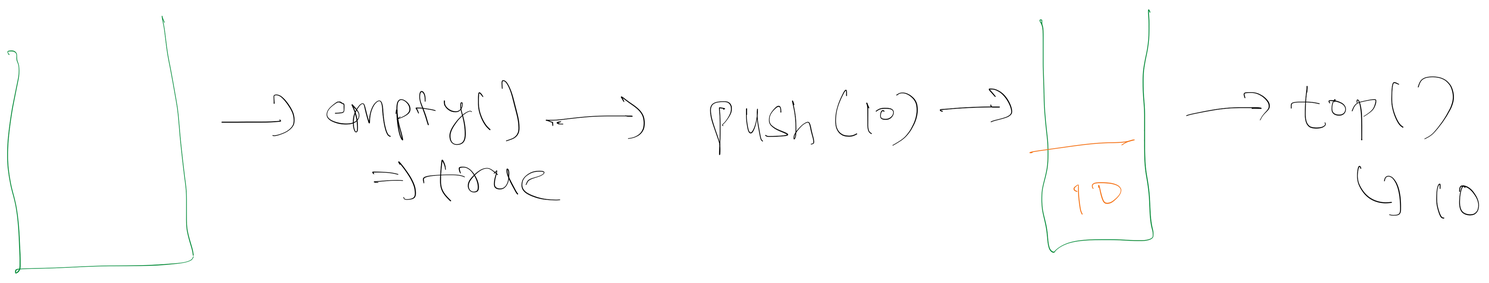
③ empty()
↳ returns true if stack is empty, else return false.

④ top()
↳ returns the top element of the stack.

Example.
consides this empty stack.







→ Applications of stack.

① Function calls | Recursion.
② Undo in an editor.
③ Balanced parenthes.

→ Stack in C++

Stack is already implemented in the c++ library.
It can be included by using #include<stack>
This stack provides similar operations as discussed before.

→ Problem

checking a given expression is balanced or not.

ex:- {([])}

this expression is balanced because for every opening bracket there is an matching closing bracket.

ex:- {([][])}

on the other hand this expression is not balanced because there mismatch for "[".

=> Solution.

we can easily solve this problem using stack.

[Solved during the class]