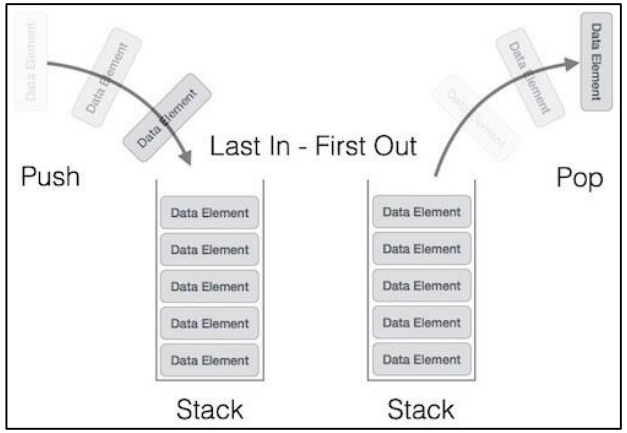
Stack

- A Stack is a linear data structure that follows the LIFO (Last-In-First-Out) or FILO (First-In-Last-Out) principle. A stack can be defined as a container or list in which insertion and deletion can be done from the one end known as the top of the stack.



Standard Stack Operations

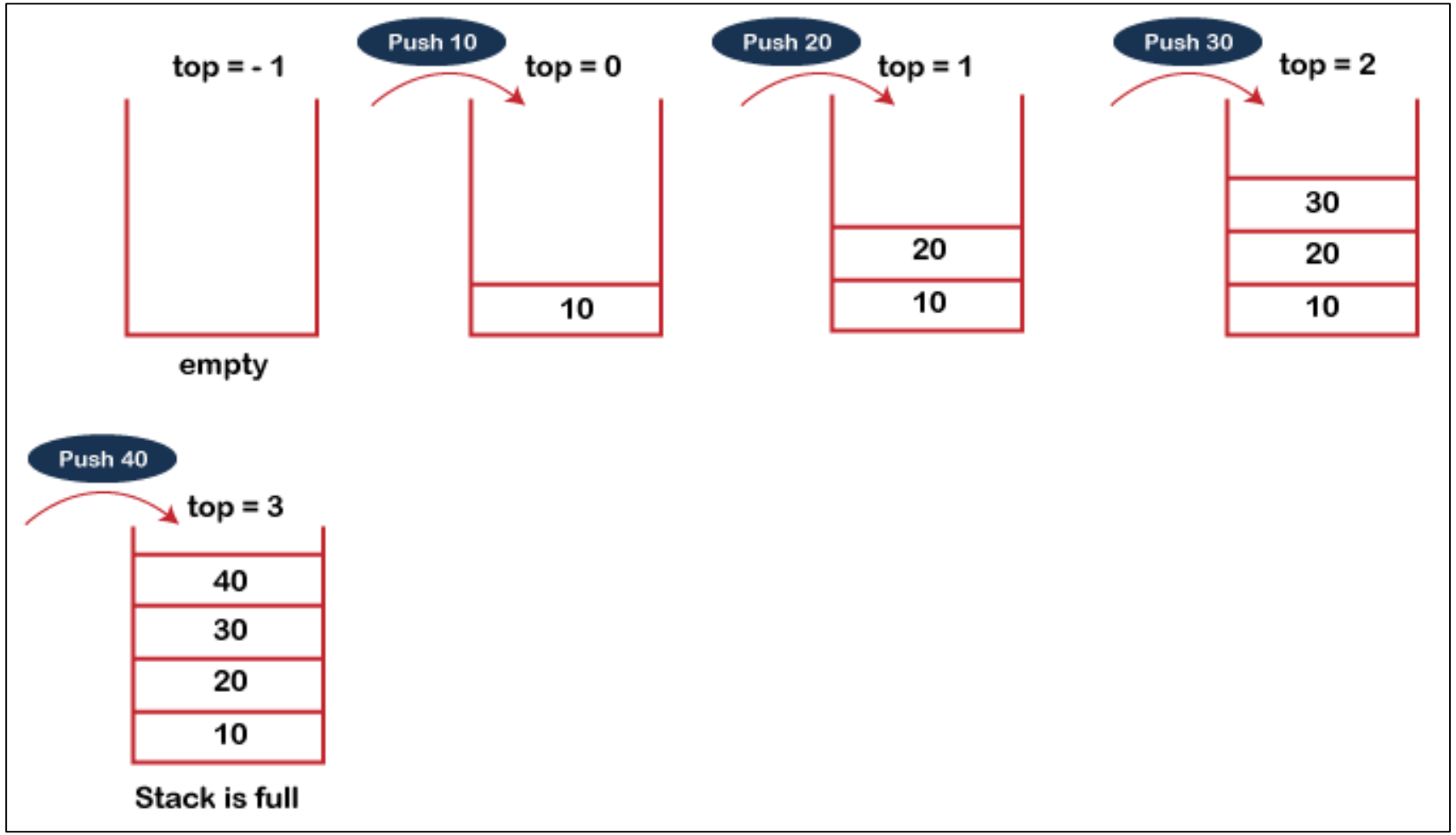
- push(): Insert an element in a stack.
- pop(): Delete an element from the stack.
- isEmpty(): It determines whether the stack is empty or not.
- isFull(): It determines whether the stack is full or not.
- peek(): It returns the element at the given position.

PUSH Operation

The steps involved in the PUSH operation is given below:

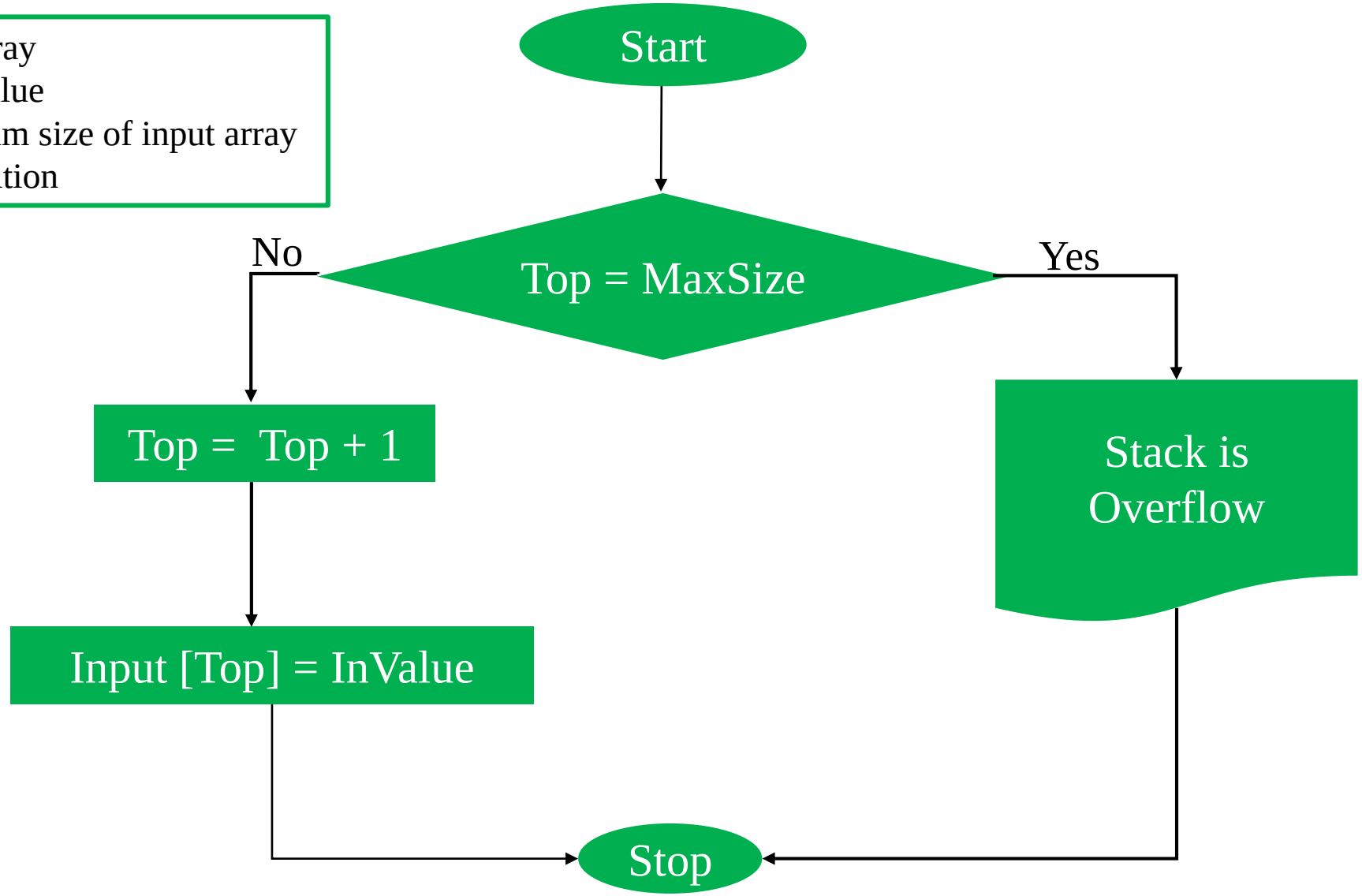
- Before inserting an element in a stack, we check whether the stack is full.
- If we try to insert the element in a stack, and the stack is full, then the overflow condition occurs.
- When we initialize a stack, we set the value of top as -1 to check that the stack is empty.
- When the new element is pushed in a stack, first, the value of the top gets incremented, i.e., top=top+1, and the element will be placed at the new position of the top.
- The elements will be inserted until we reach the max size of the stack.

PUSH Operation



Flowchart of PUSH Operation

Input [] = Input Array
InValue = Insert Value
MaxSize= Maximum size of input array
Top = Inserted Position

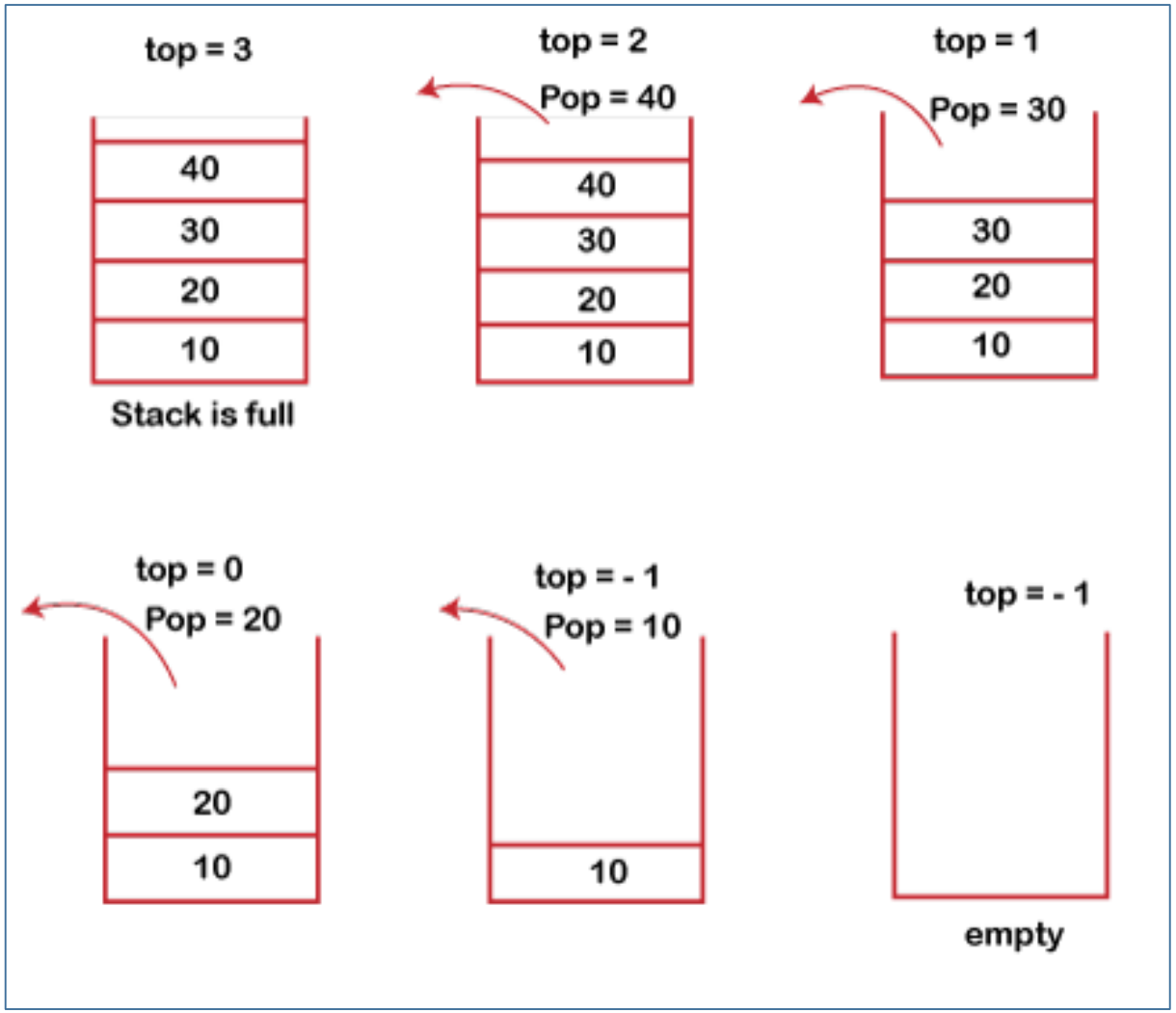


POP Operation

The steps involved in the POP operation is given below:

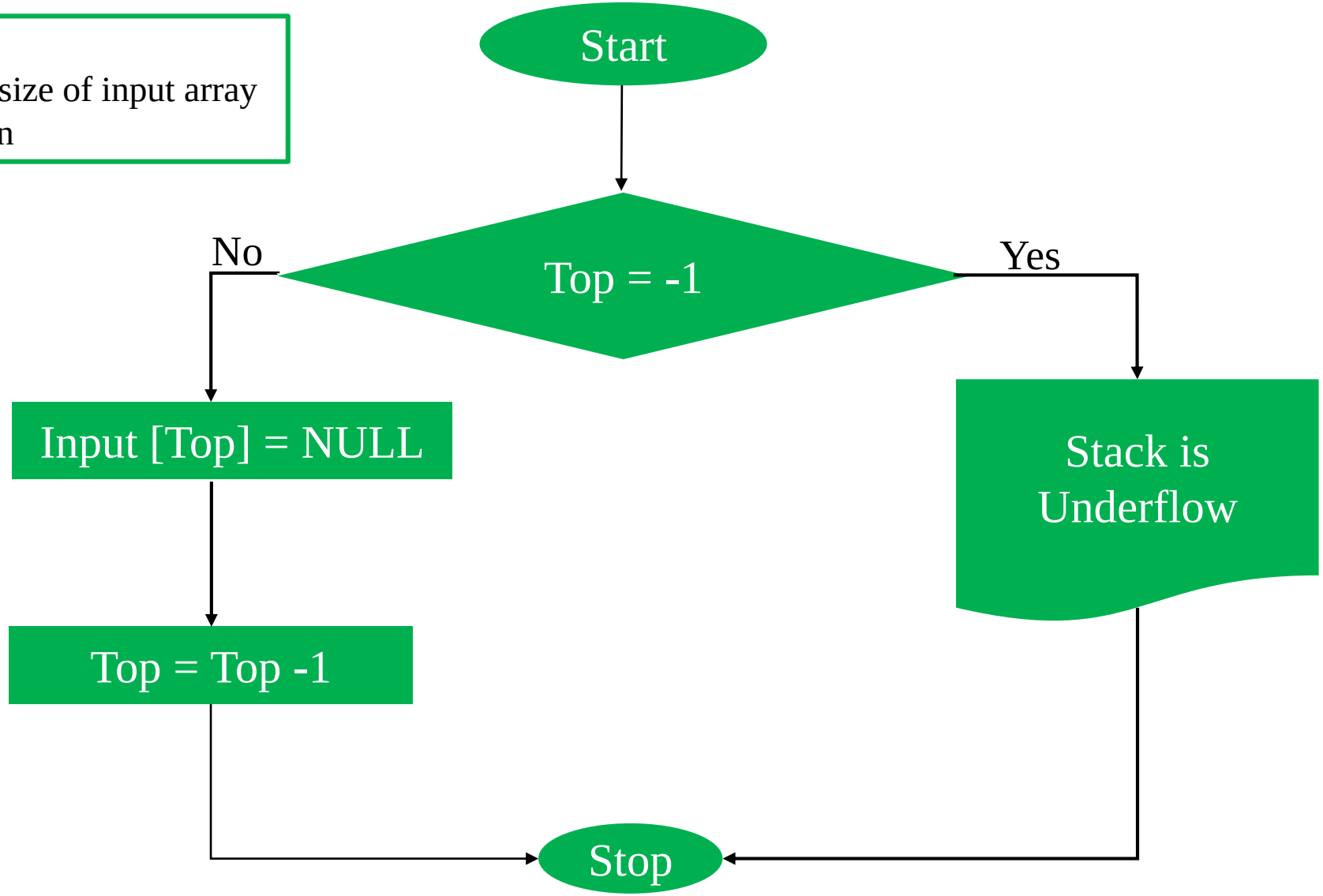
- Before deleting the element from the stack, we check whether the stack is empty.
- If we try to delete the element from the empty stack, then the underflow condition occurs.
- If the stack is not empty, we first access the element which is pointed by the top
- Once the pop operation is performed, the top is decremented by 1, i.e., top=top-1.

POP Operation: Cont…



Flowchart of POP Operation

Input [] = Input Array
MaxSize= Maximum size of input array
Top = Deleted Position



Mathematical Example

Give the correct output for the following sequence of operations. Show all steps? (Answer: 8 5 2 5 1)

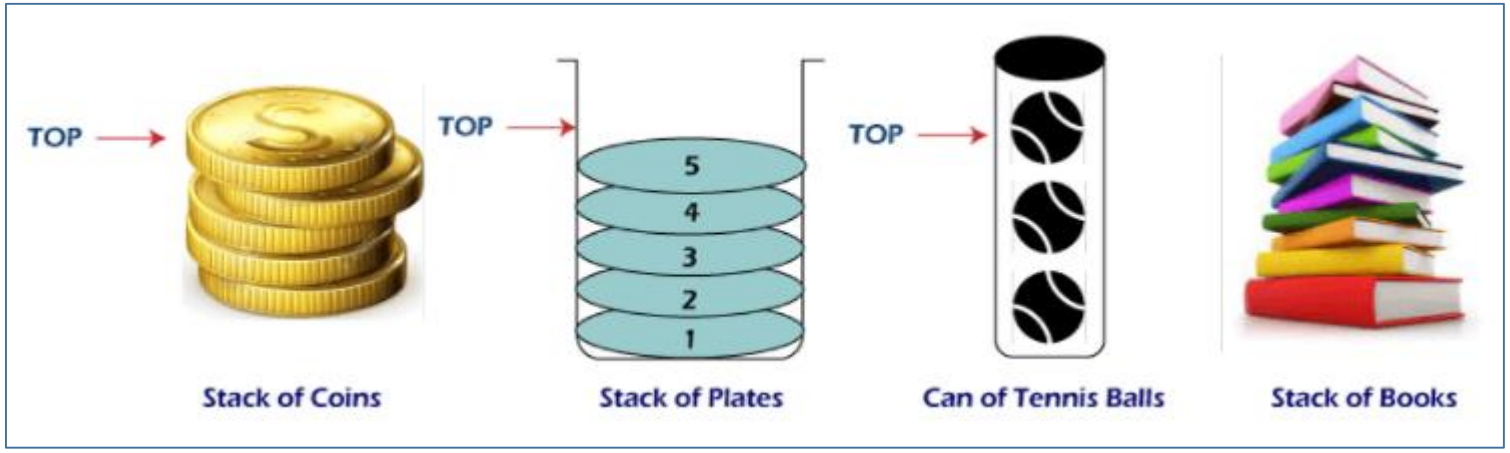
push(5)
push(8)
pop
push(2)
push(5)
pop
pop
pop
push(1)
pop

Explanation

Input	STACK	Output
push (5)	5	
push (8)	5, 8	
pop	5	8
push (2)	5, 2	
push (5)	5, 2, 5	
pop	5, 2	8, 5
pop	5	8, 5, 2
pop		8, 5, 2, 5
push (1)	1	
pop		8, 5, 2, 5, 1

Application of Stack

- String reversal
- UNDO/REDO operation
- Recursion
- Backtracking
- Floors in a Building



Advantage and Disadvantage of Stack

❑ *Advantages*

- Stack is easy to learn and implement for beginners.
- Does not allow resizing of variables.
- Stacks are used to solving problems that work on recursion.

❑ *Disadvantages*

- Random access of elements is impossible in stacks.
- Chances of stack overflow
- Stacks are neither flexible nor scalable.