Theory

* Stack

A stack is a linear data structure that follows the last in first out principle meaning the last element added to the stack is the first one to be removed stack can be implemented using array or linked lists and support two main operation

1) push : Adds an element to the top of the stack
2) pop : Removes the element from the top of the stack.

* Well - formed parenthises

Well formed parentheses refer to a sequence of parentheses that are correctly matches and nested for a string of parenthesis to be well formed every opening pakenthesis must have a corresponding closing parenthesis in the correct order

eg

① Well - formed
- ()
- (() ())

2) Not well formed
- ( (unmatched opening)
- ()) (extra closing)

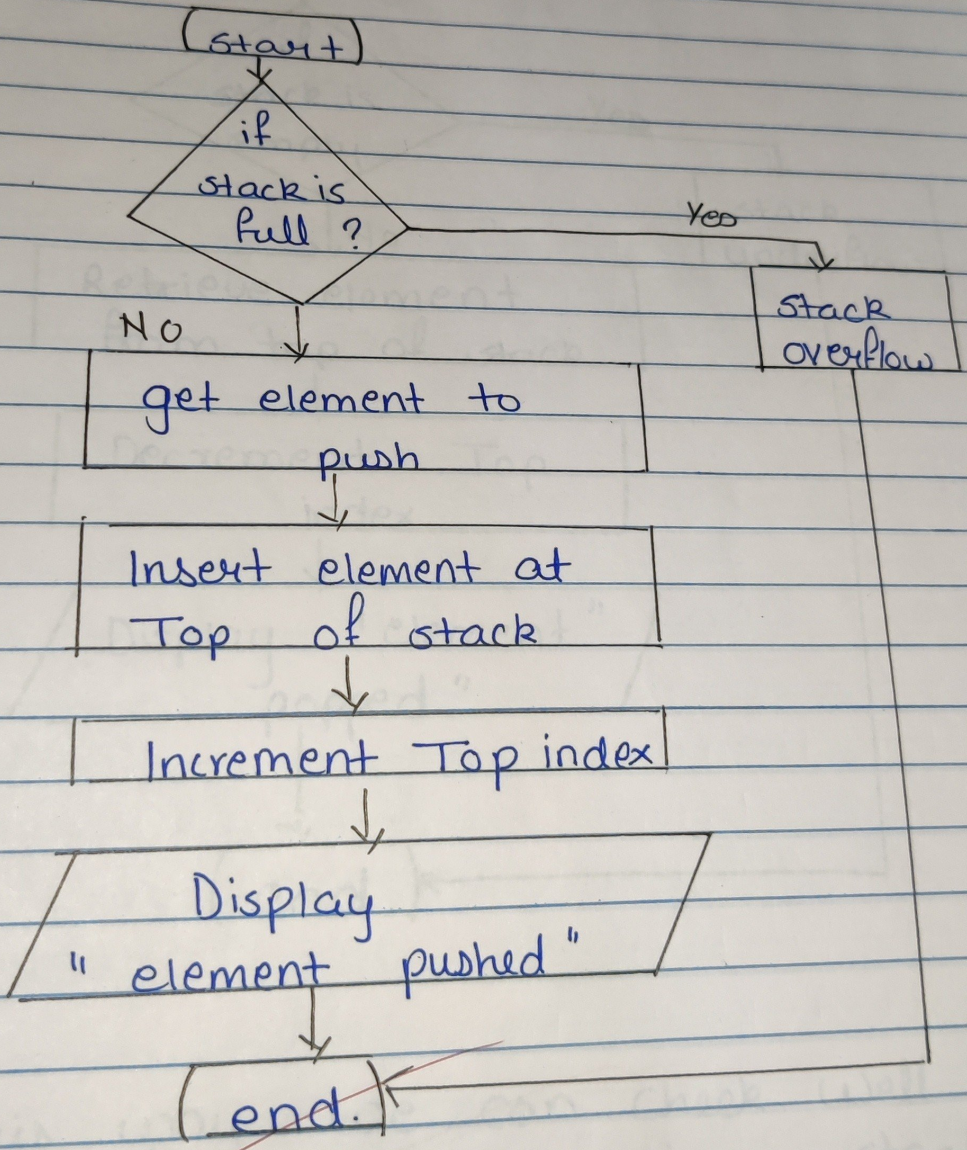
Algorithm

1) push operation of stack
step1) start
step2) Check stack has some space or stack is full
step3) If the stack has no space then display "overflow" and ext
step4) if the stack has space then increase top by 1 to print point next empty space
step5) Add item to the newly stack location where top is pointing
step6) Push operation performed successfully
step7) end

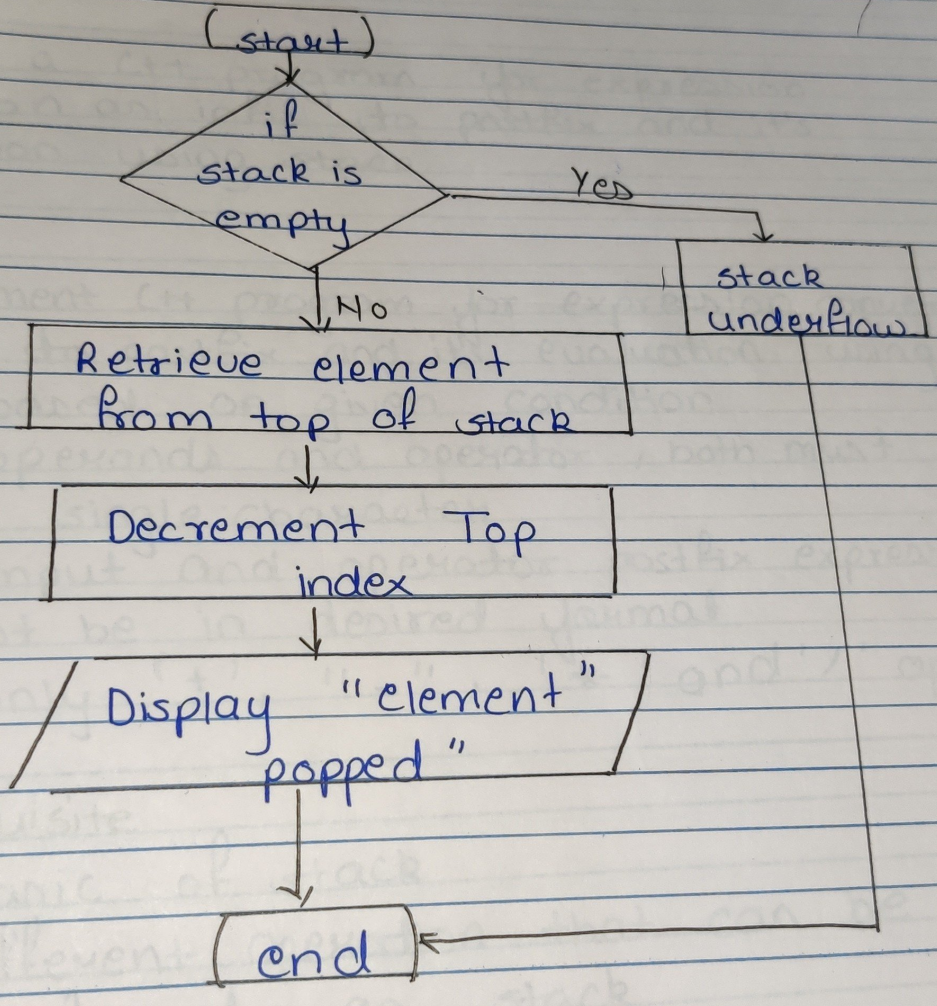
2) pop operation of stack
step1) start
step2) Check status has some element or stack empty
step3) If the stack has no element means it is empty then display "underflow"
step4) If the stack has element some element. accesses the data element at which top pointing
step5) Decreases the value of Top by 1
step6) pop operations performed successfully
step7) End.

flowchart

push operation



pop operations




Conclusion

By this way we can check well yumedness of parenthesis using stack.