Examples on Stack & Queue

1) Leetcode Pbm No. 20 :-

s = "()" o/p = true

s = "[](){}" o/p = true

s = "[()]" o/p = true

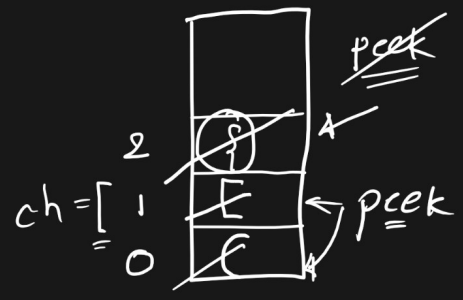
s = "[)" o/p = False

s = "{[]}" o/p = false

s = "][" o/p = false

s = "([{}])" (indexes 0 1 2 3 4 5)

```
import java.util.*;
Stack<Character> st = new Stack<Character>();
for(int i=0; i<s.length(); i++)
{
    char c = s.charAt(i);
    if(c=='(' || c=='{' || c=='[')
    {
        st.push(c);
    }
```



s = "[{]}"

i=0 s[0] = [= c

i=1 s[1] = { = c

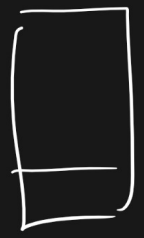
```
else
{   if (st.empty())
        return false;

    char ch = st.peek();
    if ((ch=='{' && c=='}') ||
     + (ch=='[' && c==']') ||
     + (ch=='(' && c==')'))
    {
        st.pop();
    }
    else
        return false;
}

return st.empty();
```

true

i = 2 s[2] =] = c

s = "][

{][x



Stack

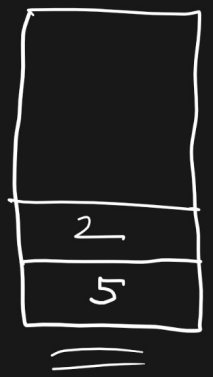
1) peek()
2) pop()
3) append()
4) empty()

2) Leetcode 682:-

string array

operation = ["5", "2", "D", "+", "C"]

o/p = sum of record



int stack

A) if incoming operation integer string value
= push inside the stack that integer record

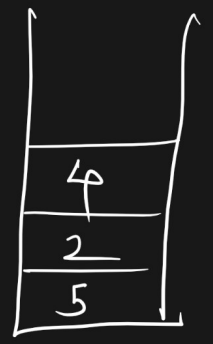
```
int r = Integer.parseInt("5")
st.push(r)
```

D) if incoming ~~char~~ string "D"

= take peek record & double that value & push into stack

peek = 2

= 2×2 = 4



3) if incoming string operation value "+"

= take last & last_second value of stack

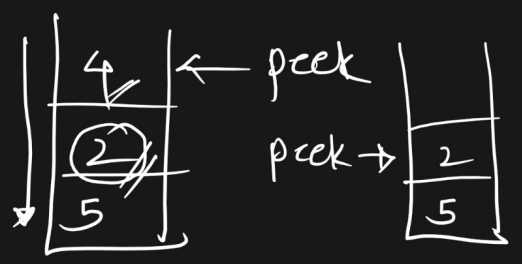
4 ← int last = st.peek(); ⇒ 4 ✓

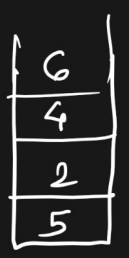
st.pop(); ⇒ remove 4

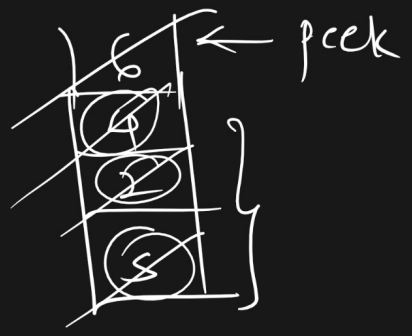
2 ← int last_sec = st.peek() ⇒ 2 ✓

st.push(last)

st.push(last + last_sec)







4) if incoming operation " C "

then we have to removing last record

⇒ st.pop();

5) Sum of records ⇒

```
int result = 0
while ( st.empty())
{
    result = result + st.pop();
}
```

6 ← 6 + 4

5 ← 4 + 2

6 + 4 = 4

4 + 2 = 6

6 + 5 = 11

return result; ⇒ // Ans

H.W. 1) Leetcode 682

2) Infix to postfix Conversion ⇒ Stuck

2+5 infix

⇒

25+ ⇒ postfix

+25 ⇐ prefix

infix

2+5-1

↳ 25+1- postfix