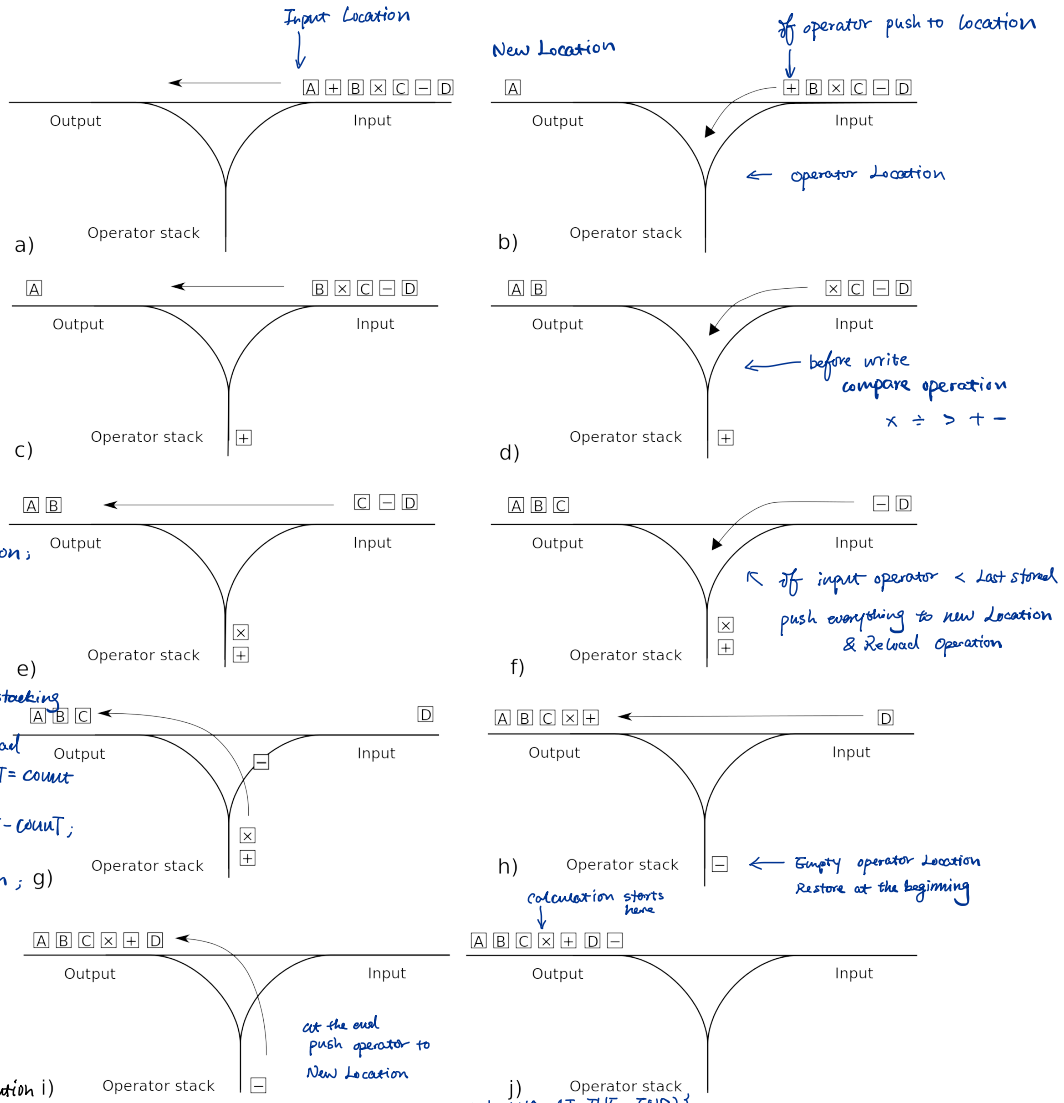

Upon using $sp

① save to $sp
② $sp = $sp − 1

Upon Reading:

M1: while (NOT_END_OF_Location) {
Load $sp value ;
save to desired Location ;
$sp = $sp + 1 ;
}

M2: Count length of operator
① Have a counter to count number of operation when stacking
② use count to backward load operator ③ FIXcount = count
④ while (count ≠ 0) {
$s1 = $sp + FIXcount − count ;
lb $t1, ($s0) ;
sb $t1, NewLocation ; count −− ;
}

```
# set $sp to desired location
la $s1, newLocation
la $s0, userInput

start:
lb $t0, ($s0)

beq $t0, $zero, EXIT

beq $t0, '+', operatorStack
beq $t0, '-', operatorstack
beq $t0, '*', operatorstack
beq $t0, '/', operatorstack

sb $t0, $s1
addi $s1, $s1, 1
addi $s0, $s0, 1
j start
```

```
operatorstack:
lb $t9, -1($sp)

bne $t9, $zero, operator compare
j storeoperator
Operator compare:

beq $t0, '*', store operator
beq $t0, '/', storeoperator

beq $t9, '*', pushoperator
beq $t9, '/', push operator

j store Operator
```

```
while (NO_AT_THE_END) {
Look through string and find first operator ;
Load two numbers before ;
operate ;
}
```

```
store Operator:   ← add counter
sb $t0, ($sp)
subi $sp. $sp. 1
addi $s0, $s0, 1
j start
push operator:
# Reference Up

addi $s0, $s0, 1
# Advance Input Location
j start

EXIT:
# Do whatever you gotta do next
```