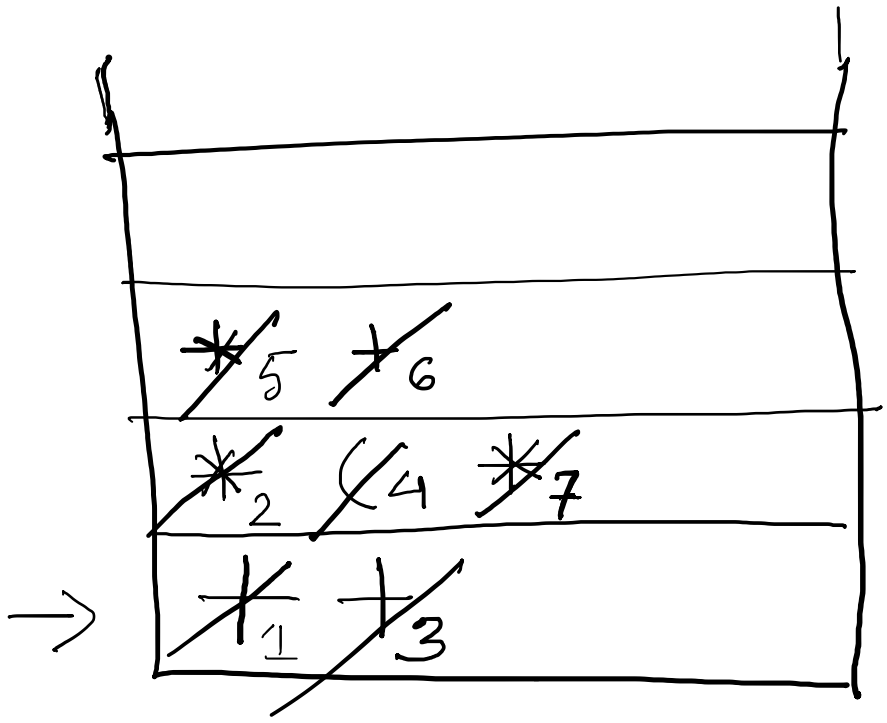
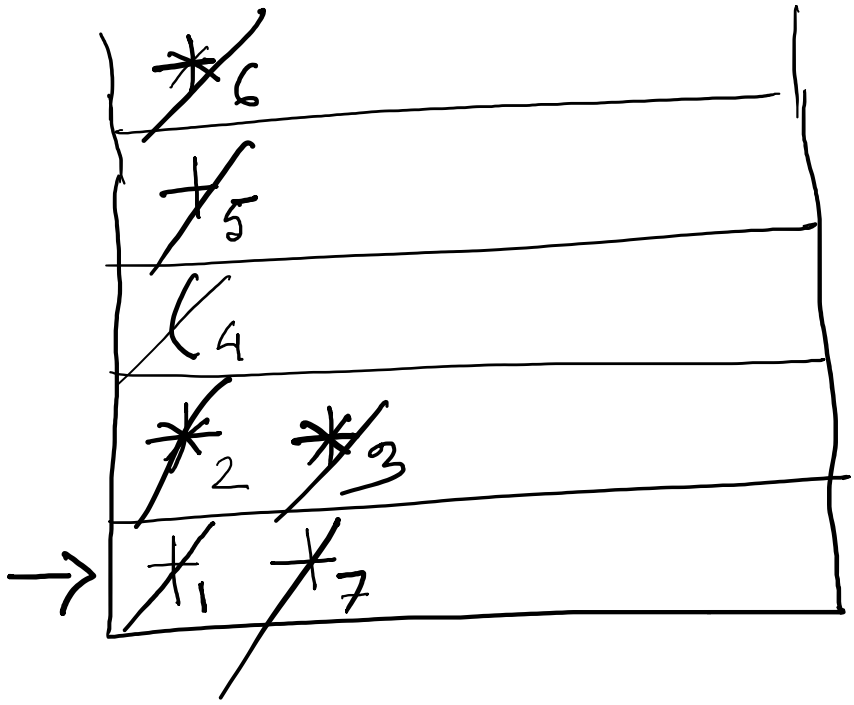
$a + b * c + (d * e + f) * g$

output: $abc*_2 +_1 de*_5 f +_6 g*_7 +_3$



1 + 2 * 3 * (4 + 5 * 6) + 7

Output: 1 2 3 *2 4 5 6 *6 +5 *3 +1 7 +7

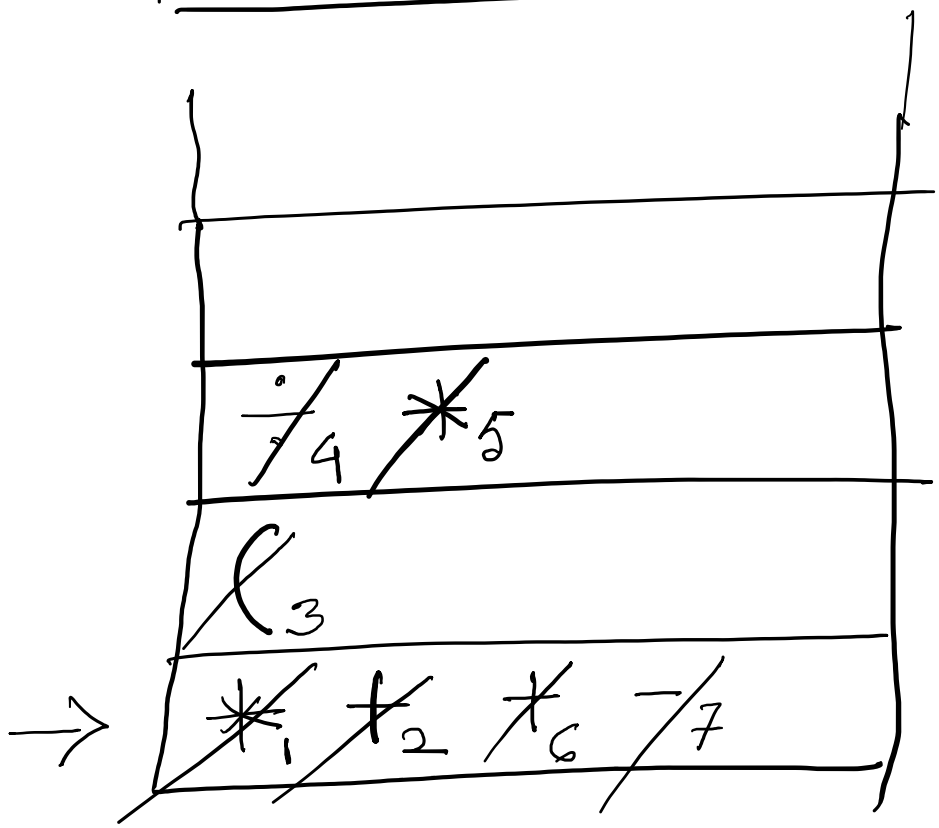


- 5 * 3 + (6 ÷ 3 * 4) + 8 − 1
- (5 − 2 * 2) + 8 ÷ 2 − 1 * (4 + 9)

$5 * 3 + (6 \div 3 * 4) + 8 - 1$

Output: $5\ 3\ *_1\ 6\ 3\ \div_4\ 4\ *_5\ +_2\ 8\ +_6\ 1\ -_7$

53*63÷4*+8+1−

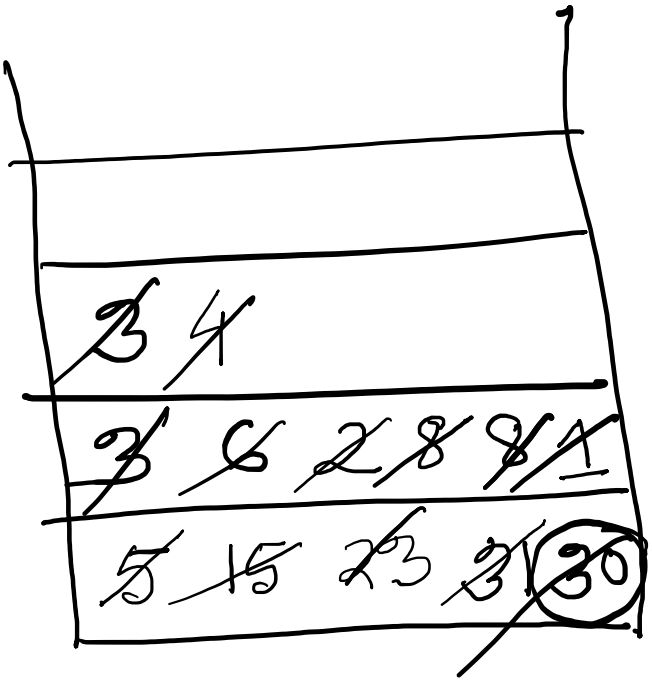


$15 + 8 + 7$

$= 30$

$=$

5 3 * 6 3 ÷ 4 + 8 + 1 −



5 * 3 = 15

6 ÷ 3 = 2

2 * 4 = 8

15 + 8 = 23

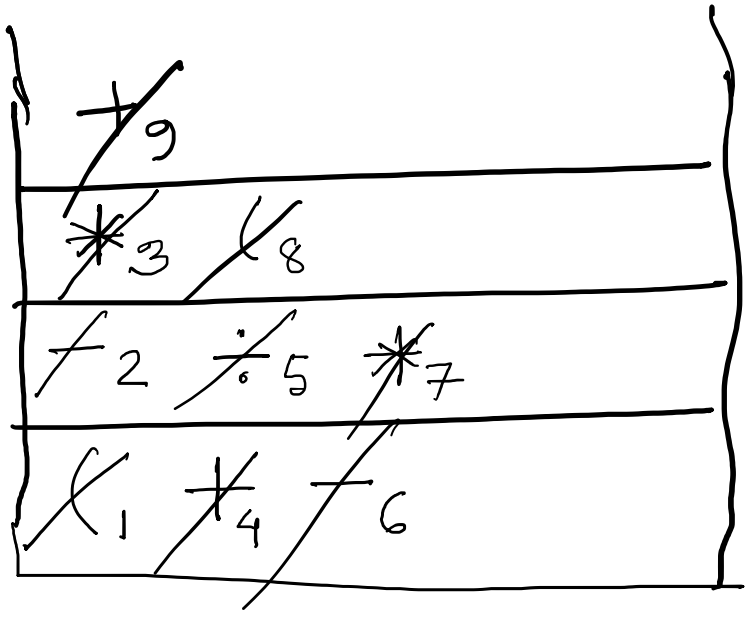
23 + 8 = 31

31 − 1 = 30

$(5-2*2)+8\div2-1*(4+9)$

Output: $5\ 2\ 2\ *_3\ -_2\ 8\ 2\ \div_5\ +_4\ 1\ 4\ 9\ +_9\ *_7\ -_6$

$5\ 2\ 2\ *\ -\ 8\ 2\ \div\ +\ 1\ 4\ 9\ +\ *\ -$



5 2 2 * − 8 2 ÷ + 1 4 9 + * −

↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑

$2 * 2 = 4$

$5 - 4 = 1$

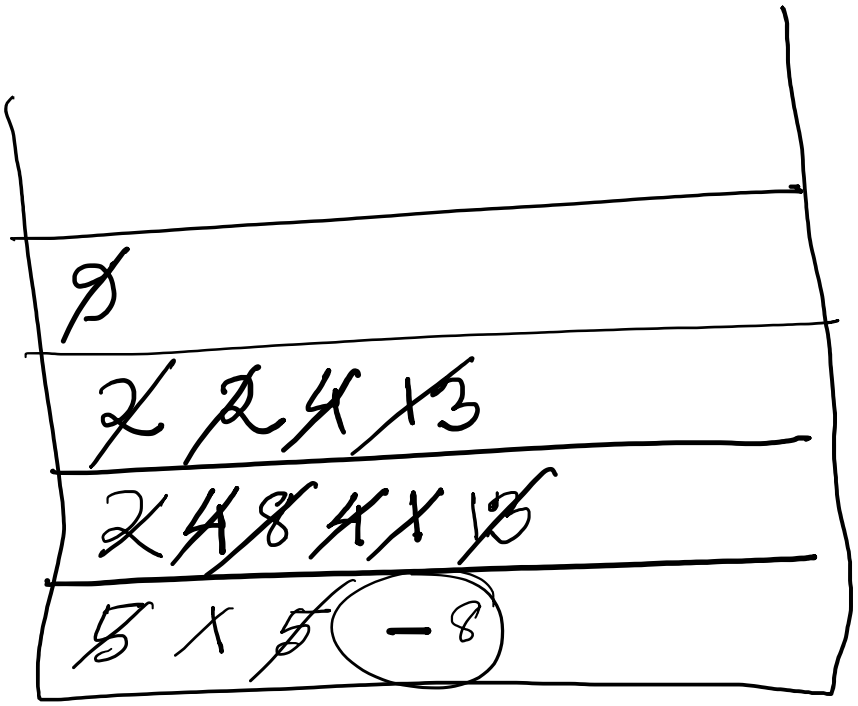
$8 \div 2 = 4$

$1 + 4 = 5$

$4 + 9 = 13$

$1 * 13 = 13$

$5 - 13 = -8$



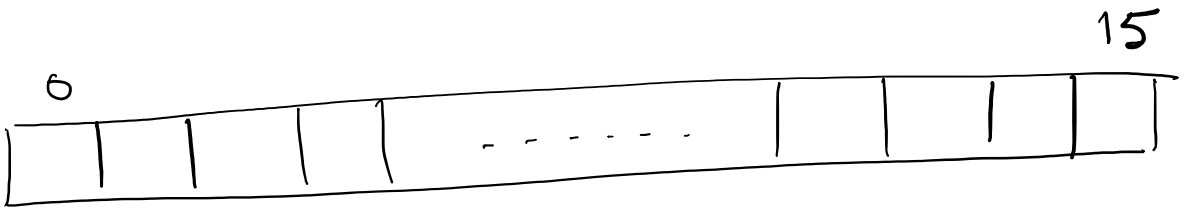
Queue

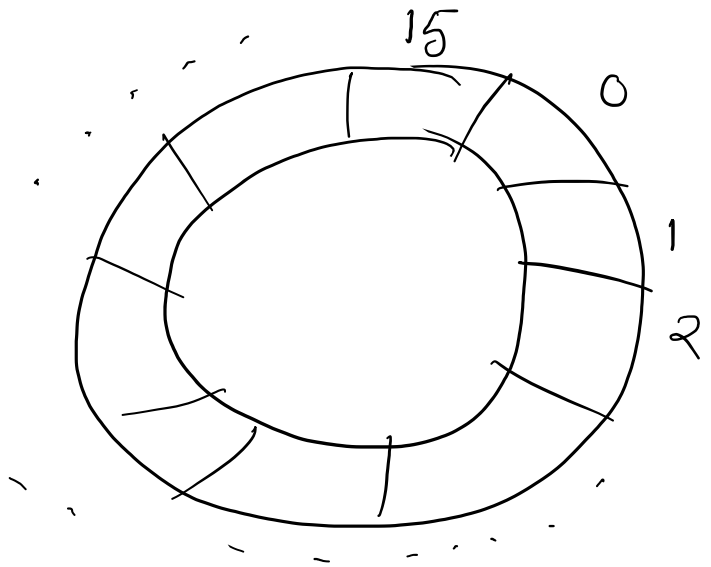
Operations

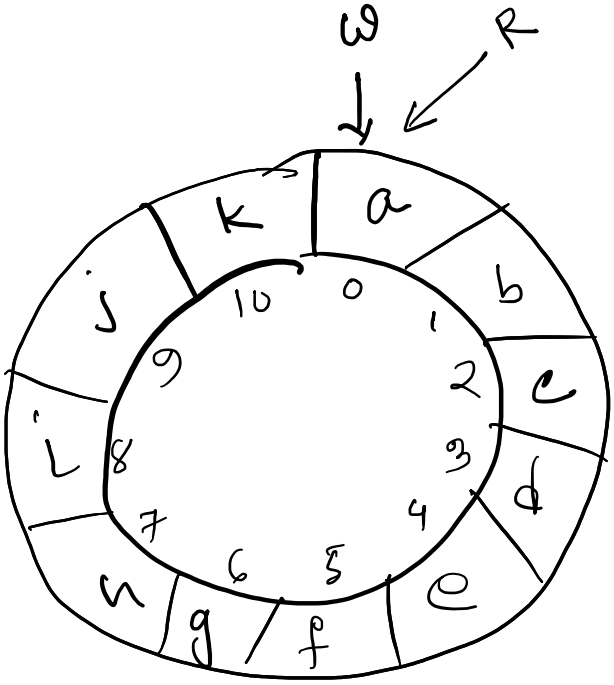
→ enqueue
- ↳ insert at the end
- ↳ rear / back / write

→ dequeue
- ↳ remove an element from front
- ↳ front / read

→ FIFO







R == W → empty

R == W → full

ambiguity ??

check if the cell is empty or not

↳ if empty
↳ the list is empty

↳ if not
↳ the list is full

OR

check size

index = (index + 1) % N → array size