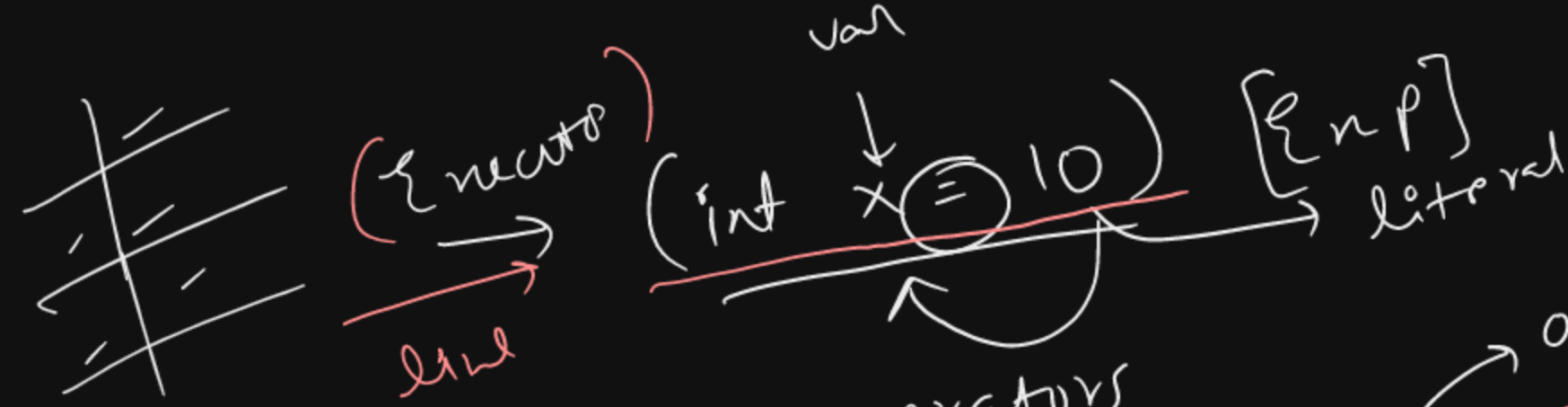


- > operators ✓
- > expressions and statements
- > printing and formatting
- > toString()
- > input buffers and scanners
- > control flow

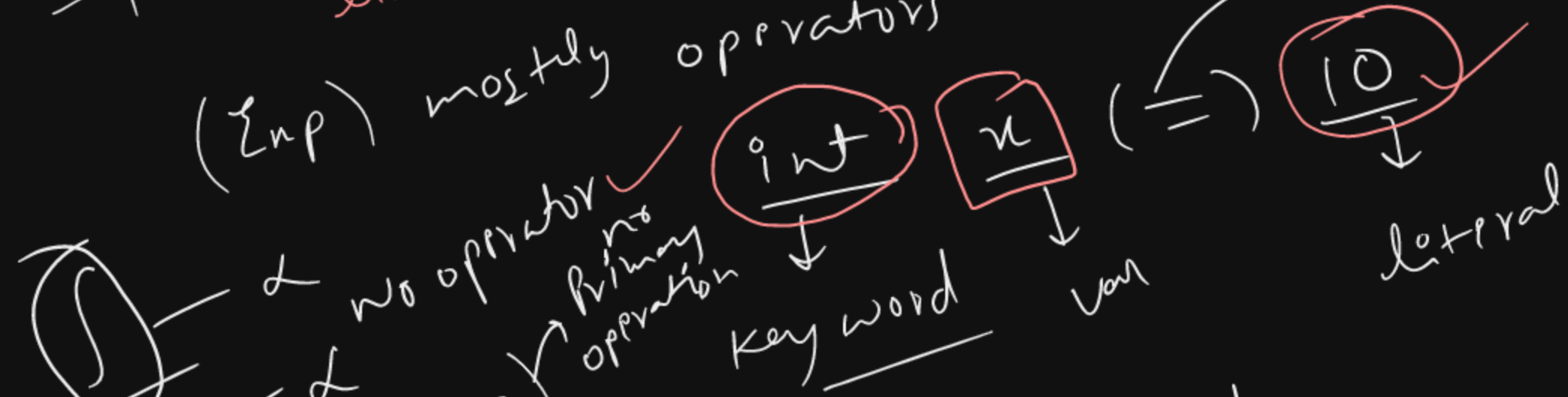
// Statement

Expression

Statement which executes  
and have a certain  
meaning to it



(Exp) mostly operators



identities  
literal  
or  
Object

(key word) → reserved word  
(identifier) → variable  
[Expressions]

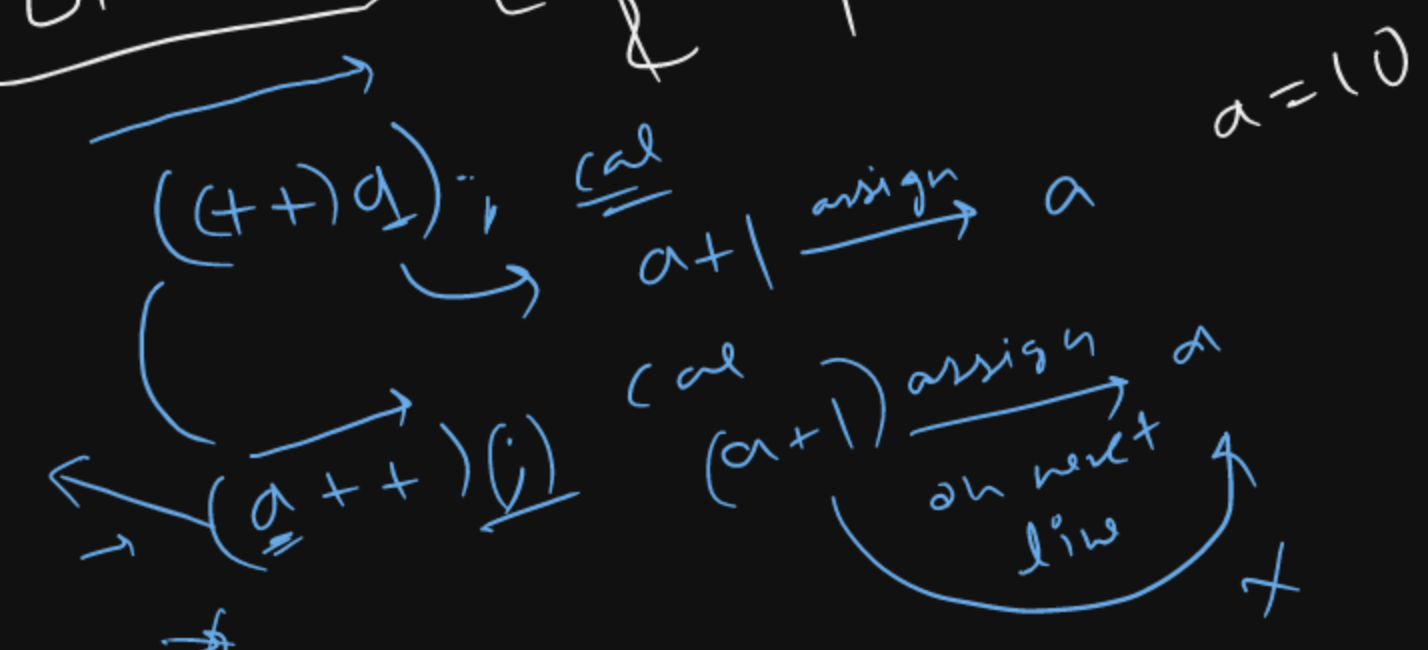
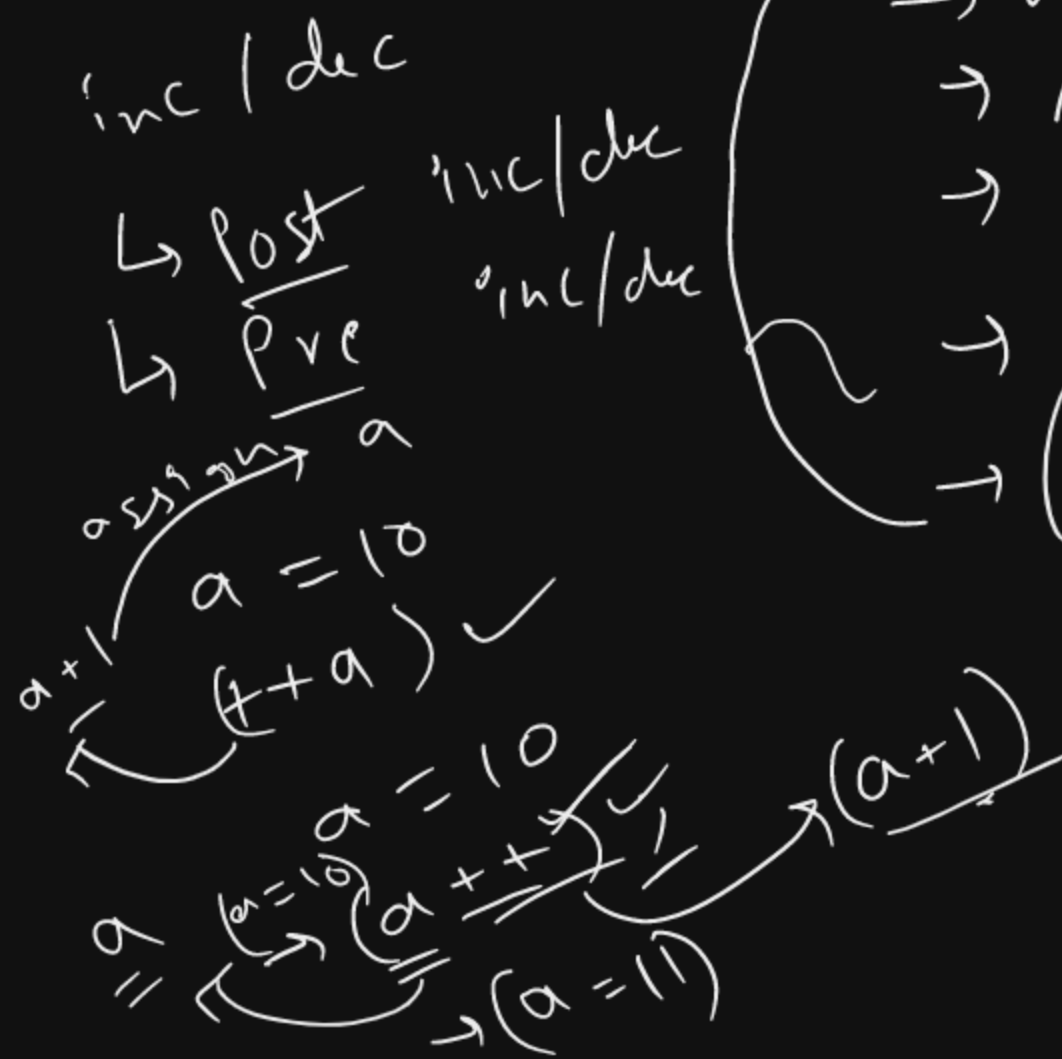
provides functionality

Primary Data types -  
in limitation of computing  
(+, -, /, \*, %)

- > operators ✓
- > expressions and statements
- > printing and formatting
- > toString()
- > input buffers and scanners
- > control flow

Operators → Primary Datatype  
 Operators → functionality  
 Operators → computer limitation

- Relational -  $<, >, <=, >=$
- Assignment -  $=, (=), -=$
- Arithmetic -  $+, -, *, /, \%, ++, --$
- Logical -  $\&\&, ||, !$  boolean
- Comparison -  $==, !=$
- (Bitwise) -  $\&, |, \sim, \wedge, \ll, \gg, \text{xor, left, right}$



a += 10

a += 1

→ [a++]

- > operators ✓
- > expressions and statements
- > printing and formatting
- > toString()
- > input buffers and scanners
- > control flow

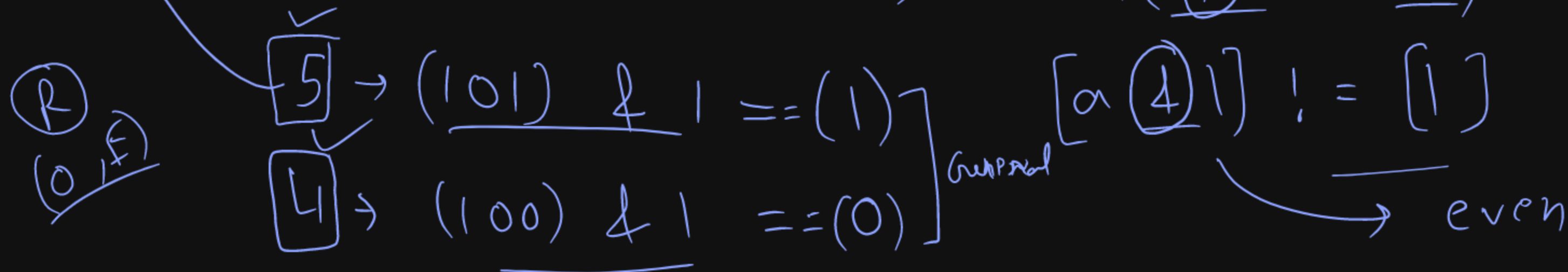
$$\begin{array}{r} 1011 \\ 8021 \\ \hline 0001 \end{array}$$

a & b		Rs
1	0	0
0	1	0
1	1	1
0	0	0

O r	
1	0
0	1
1	1
0	0

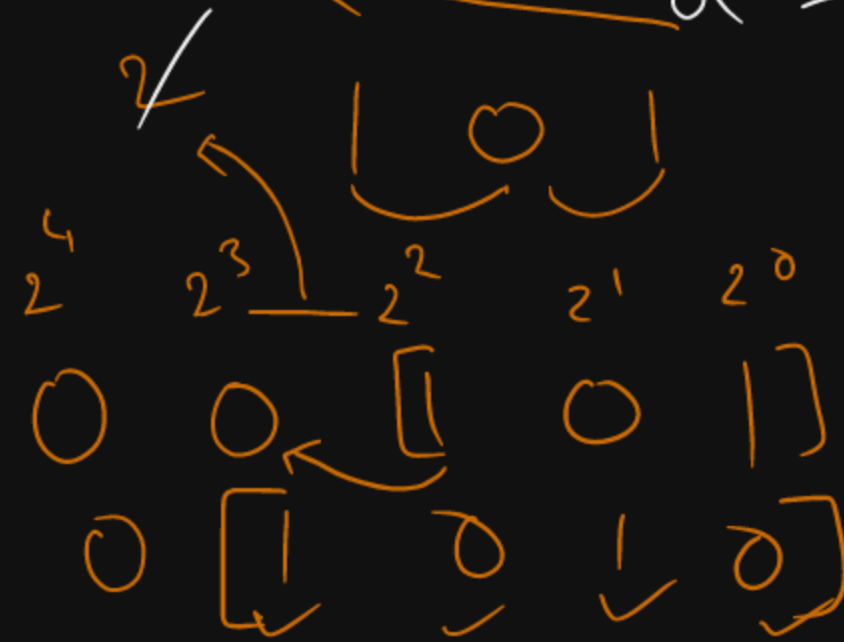
Xor	
1	0
0	1
1	1
0	0

isEven(a) → return (a % 2 == true) Pseudo code



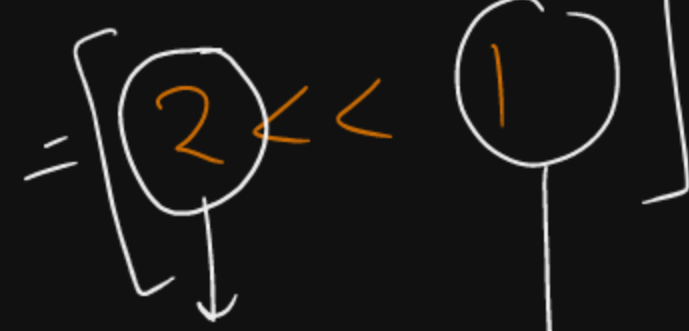
- > operators ✓
- > expressions and statements
- > printing and formatting
- > toString()
- > input buffers and scanners
- > control flow

12



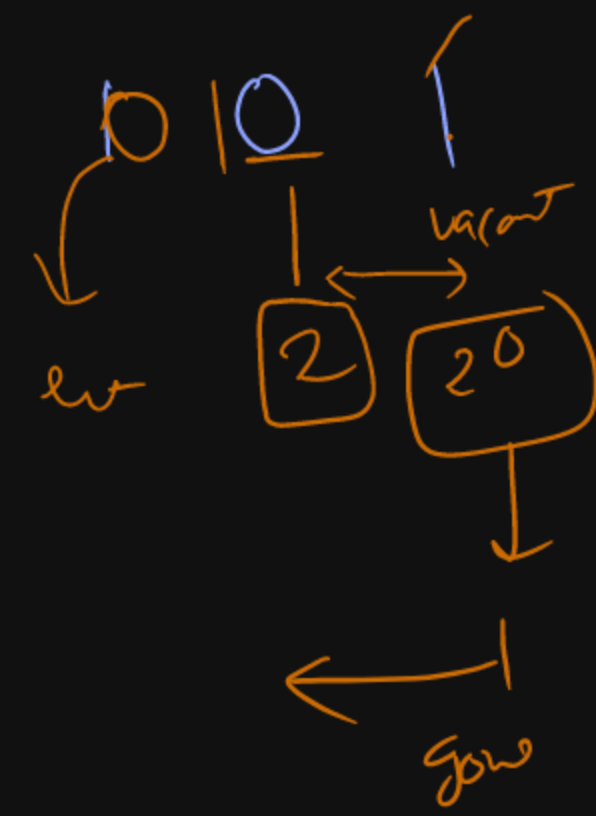
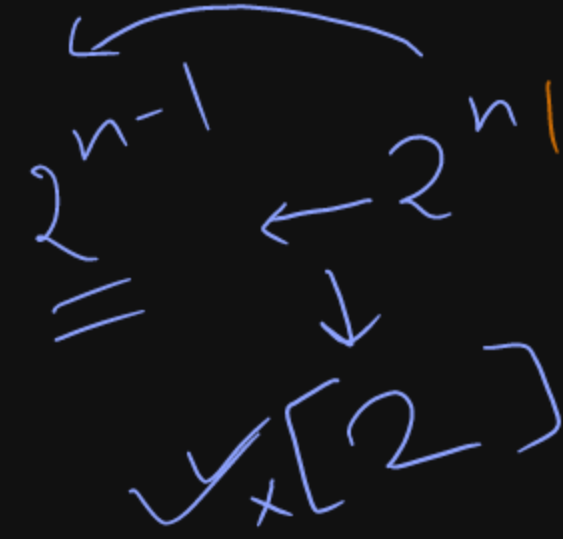
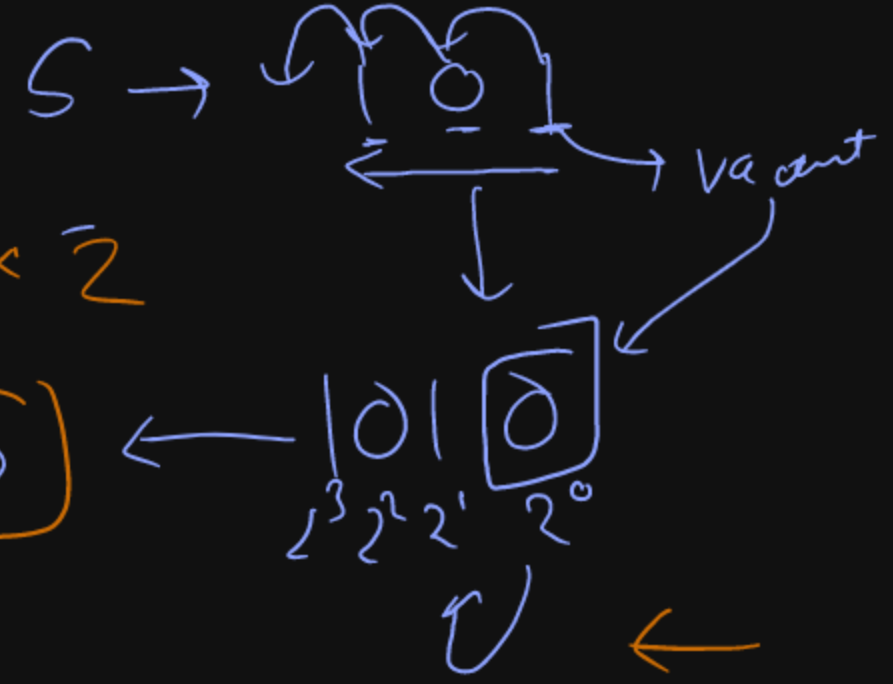
LS  
 $\underline{<< (*2)}$

RS  
 $\underline{>> (/2)}$



number

kitni bar  
 shift  
 krha



- > operators ✓
- > expressions and statements ✓
- > printing and formatting ✓
- > (toString()) (oops)
- > input buffers and scanners
- > control flow

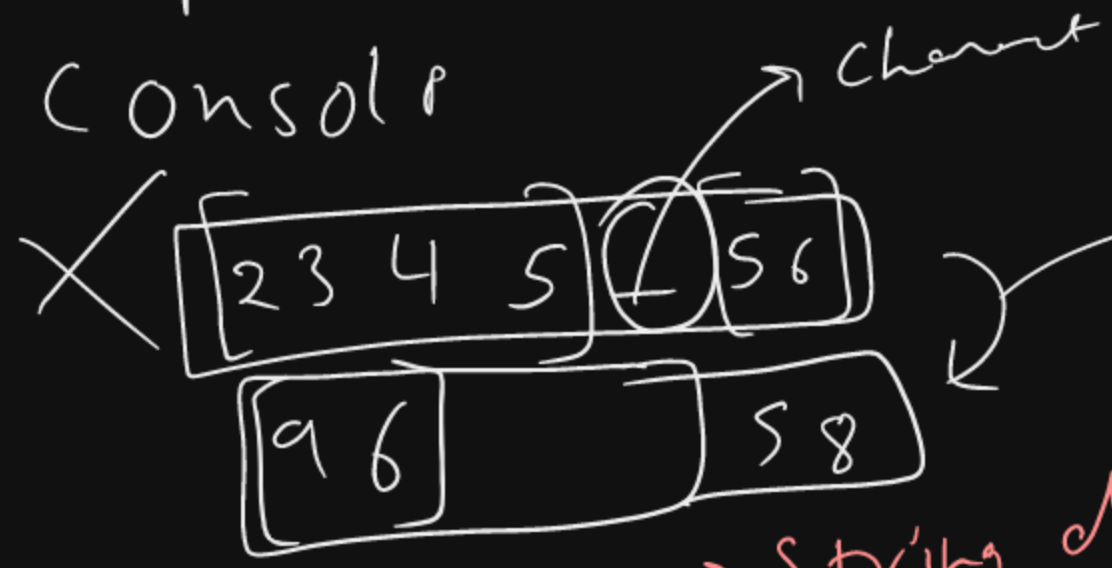
(Escape character)

- > space = " "
- > Enter = "\r\n"
- > Newline = "\n"
- > tab = "\t"

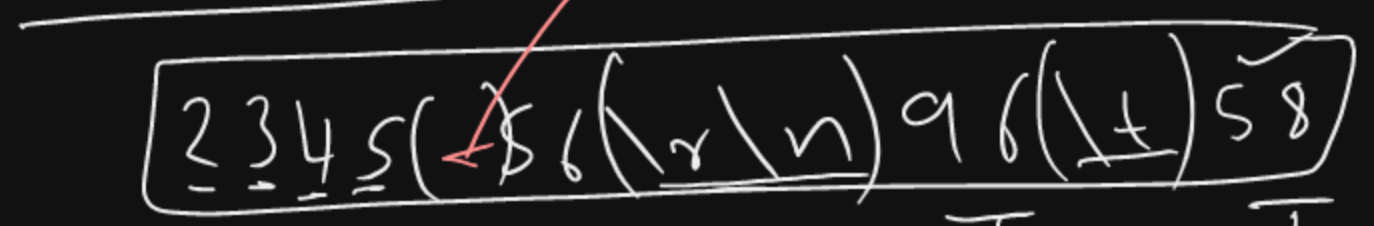
int → a  
int → b  
int → c  
int → d

input buffer

Console



string ✓



String don't treat as Esc(char)

nextInt()

Scanner

Integer  
String

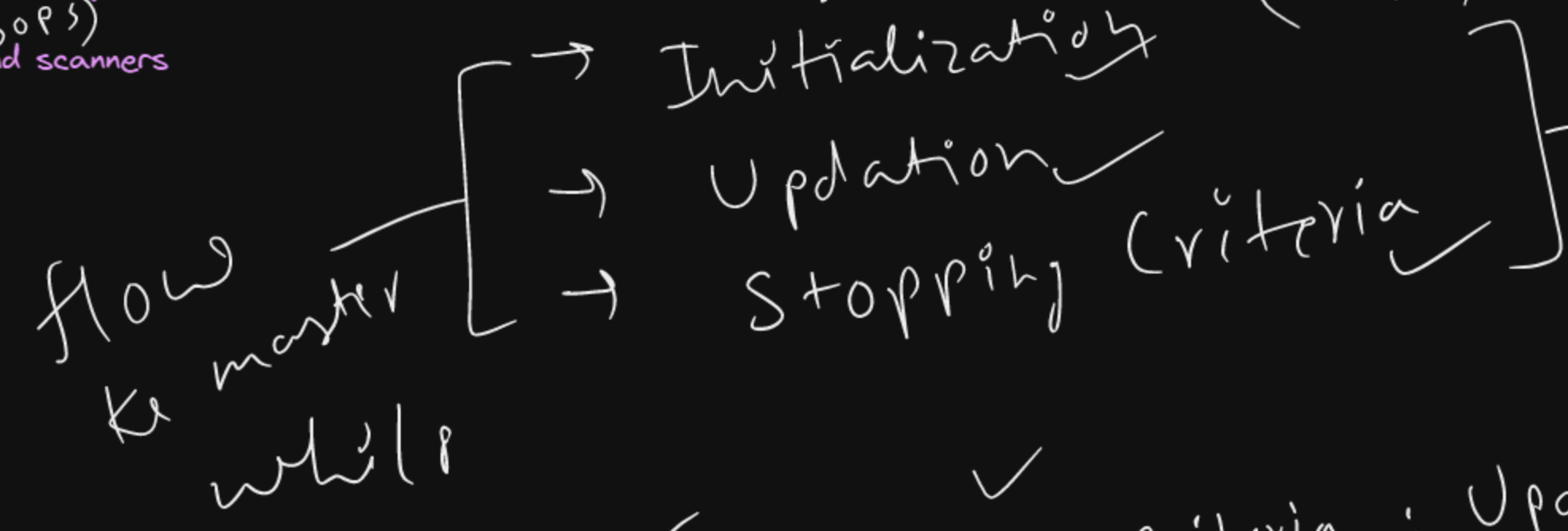
next()  
nextLine()



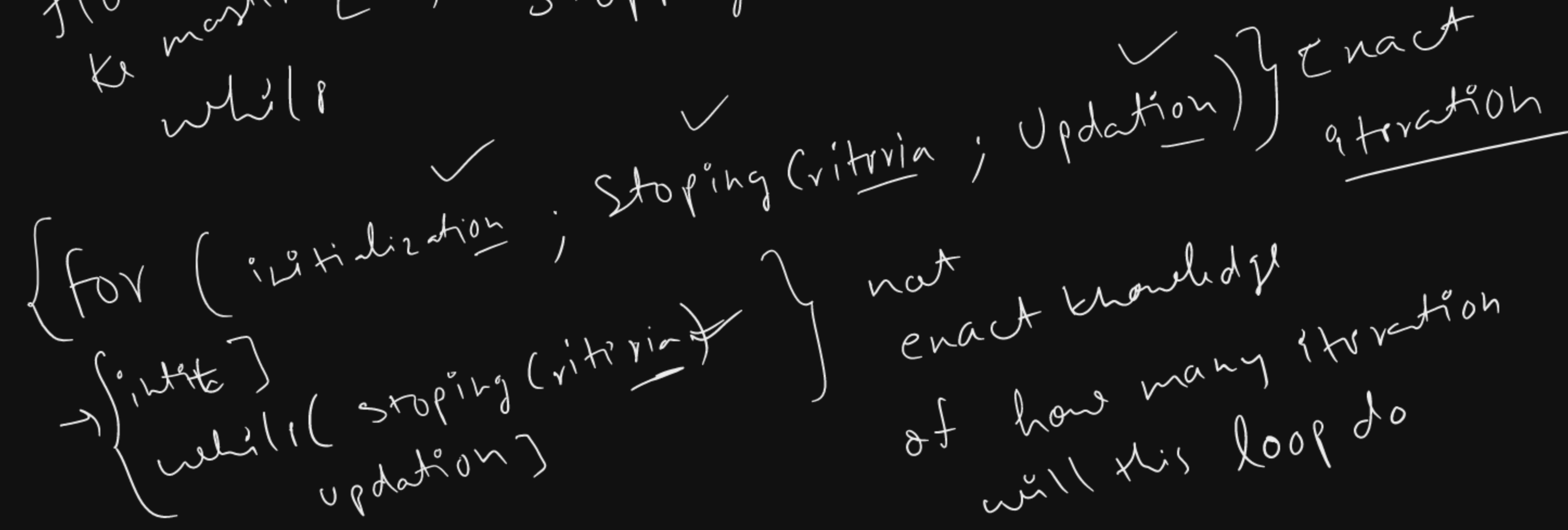
- > operators ✓
- > expressions and statements ✓
- > printing and formatting ✓
- > toString() (oops) ✓
- > input buffers and scanners ✓
- > control flow ✓

3 things

(Loop)



for



- > operators ✓
- > expressions and statements ✓
- > printing and formatting ✓
- > toString() (oops)
- > input buffers and scanners
- > control flow

how to think

Pseudo code (loop)

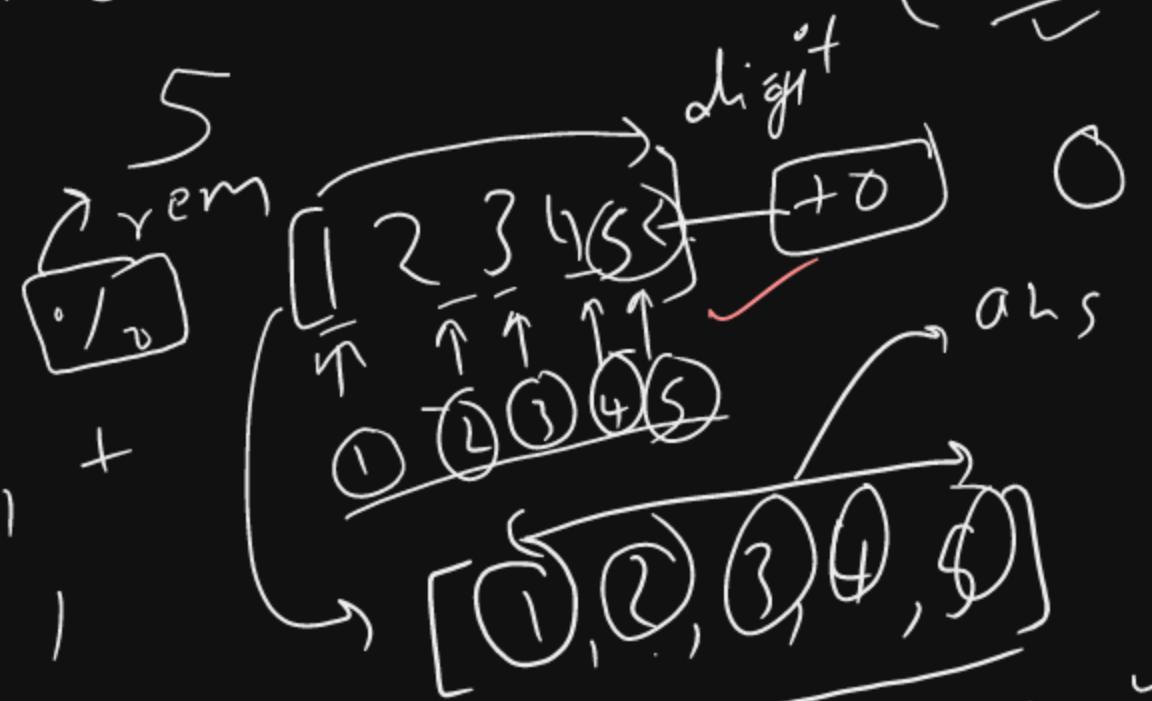
- > pura number todna right se left tk
- > digits jo multiply havi h ( $10^n$ )
- > count those
- > Ans

count = 0 (++, ++, ++)

(1 2 3 4 5)

number of Digits

3  $\left[ \begin{array}{l} \text{init} \\ \text{up} \\ \text{store} \end{array} \right]$



$$\rightarrow [(1 \times 10000) + (2 \times 1000) + (3 \times 100) + (4 \times 10) + 5]$$

$$\text{num} \quad \text{div} \quad \text{a} \\ 10000 = 1 \times 10000 + 0$$

Base 10

12345/10