

L3-Basics of Programming

Special class

Explained
By Abhijay

Coentoy

Love Babbar • Apr 4, 2025

L3 - Basics of Programming

Operators, Conditionals & Loops



→ by Codehelp

Operators:

Arithmetical → +, -, *, /, %

++ , - =

Relational → ==, !=, >, <, >=, <=

assignment (=)

logical operators (&&, ||, !)

If we Bitwise operator

learn C++

Operators in C++

45 min

& | ^ ~ >> <<

l week

No. 1 / 7.1

/ 4w-2

Arithmetic

Binary Op

2 operands

+
-
*
/
%
/

operand
2 + 3
= 5

Unary Op

1 operand

++ --

M(Q)

Comment
by links

$++$

increment op

int $a = 5$

pre-increment

$\text{cout} \ll (++a)$

① Pehle increment karo

② Furhi increment karo

6

$++a$

a $++$

post-increment

int $a = 5$

$\text{cout} \ll (a++)$

③ Pehle increment karo

④ Furhi increment karo

6 6

a

$\text{cout} \ll a$

$\overleftarrow{-}$ \rightarrow decrement

$\overleftarrow{-}$ \rightarrow pre-decrement \rightarrow $(--a)$

$\overleftarrow{-}$ \rightarrow post-decrement \rightarrow $(a--)$

Pre-decrement \rightarrow ① Pehla decrement Karo
② fir un Karo

Int $a = 5$
 \downarrow
 $(cout << (--a)) \rightarrow Y$

$\rightarrow (cout << a) \rightarrow Y$

$Y = 4$

Ex

Post - decrement \rightarrow ① Fchl, wo Karo
② für decrement Karo

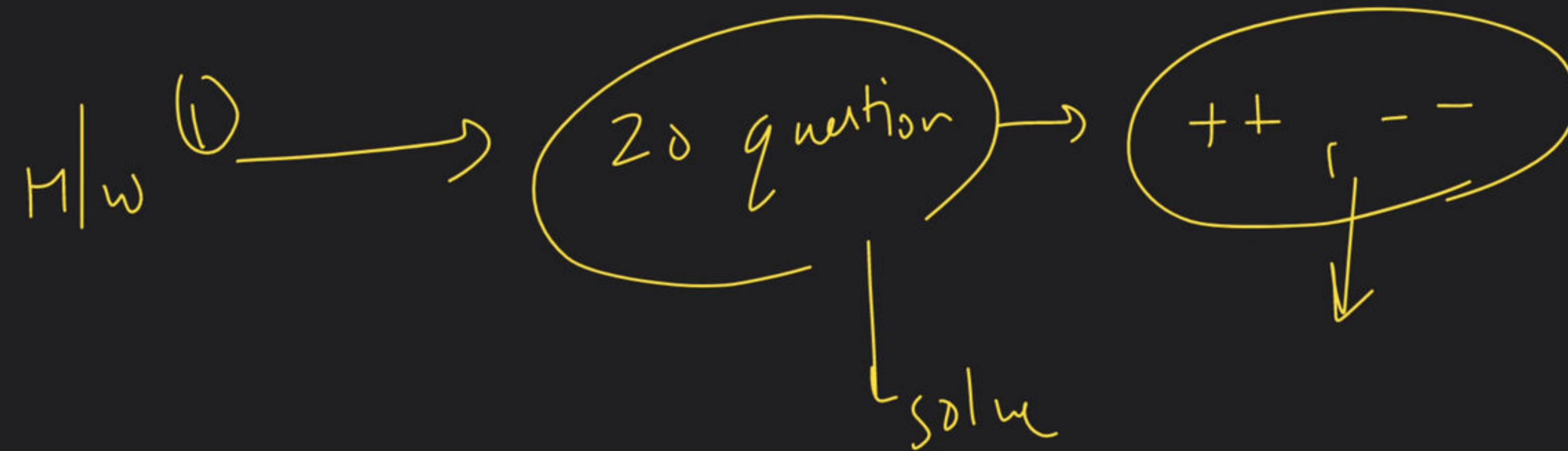
int a = 5;

cout << (a--);

→ cout << a+5 → 9

a





② Red Quiz → unlocked

```
graph LR; QuizIcon["Red Quiz"] -- crossed-out lines --> Unlocked["unlocked"]
```

Question: -

2 min

short-circuit
condition

int a = 5;

cout << (++a); → 6
cout << a; → 6
cout << (a++); → 6
cout << a; → 7
cout << (--a); → 6
(out << <; → 1
cout << (a--); → 6
cout << a; → 5

→ 5
→ 6
→ 7

H/W

any → looks hay
a = 5

int val =

(++a) * (a++) + (--a) *
(a--)

cout << val;

unary addition

$a + + +$

The diagram illustrates the expression $a + + +$. A curly brace under the first two '+' signs is labeled "unary addition", indicating that each of these operators acts on the variable a . An arrow points from the third '+' sign to the word "addition", indicating that this final operator is a binary addition operator that combines the result of the unary operations with the value b .

$a + + b$

$(a + +) + b$

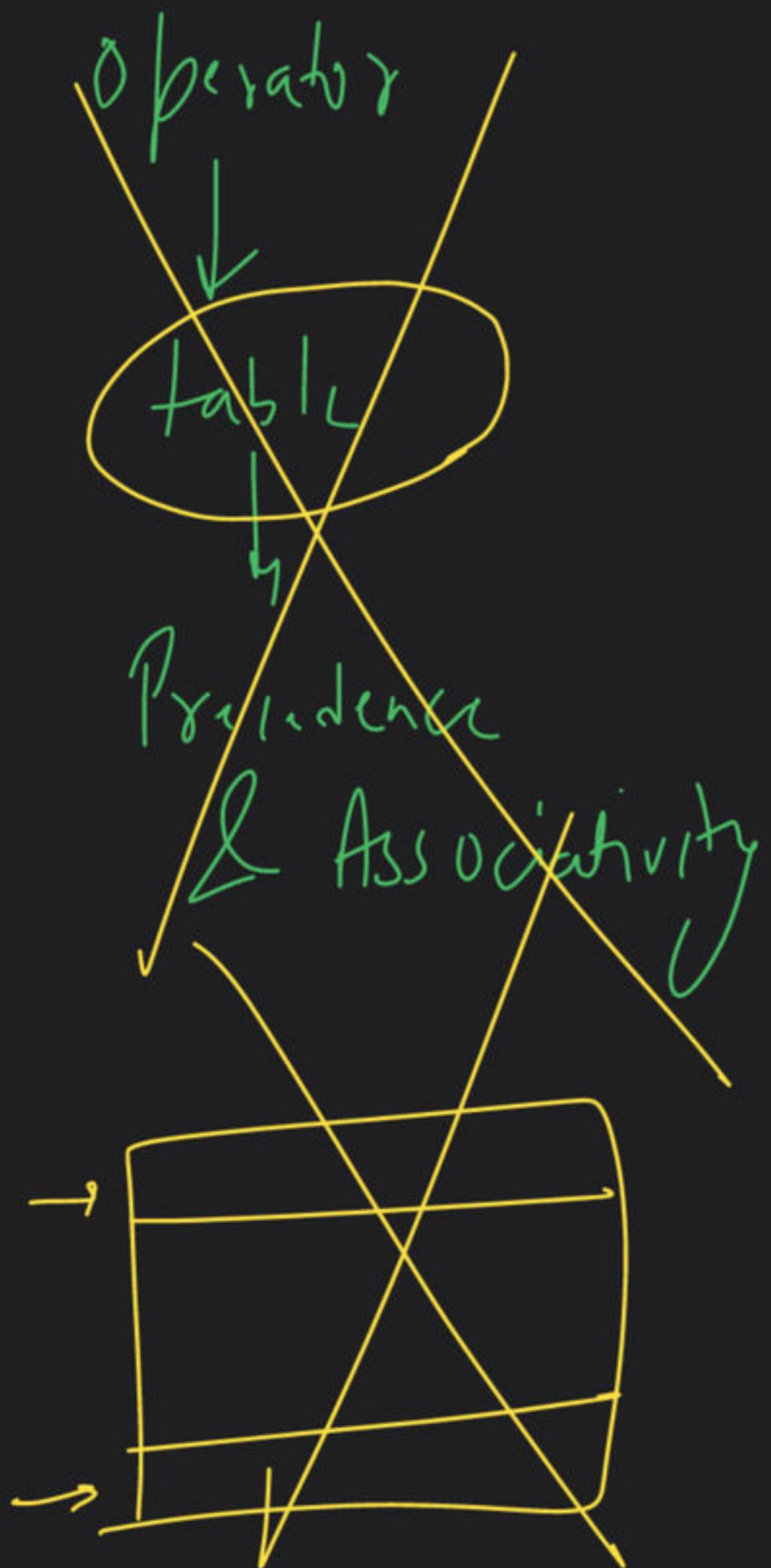
$\text{Val} = \underline{(a++) \otimes (++a)}$

$\underline{a++ \otimes ++a}$

$$\left[2 + (3 \times 4) \right] + 5$$

exp?

Use Brackets



ζ

$$a = \cancel{y}, b = \cancel{x}^f$$

$$(- - a) \star (\underline{++b}) \star 5$$

$$a = \cancel{x}, b = \cancel{y}$$

$$y \times \delta \times r$$

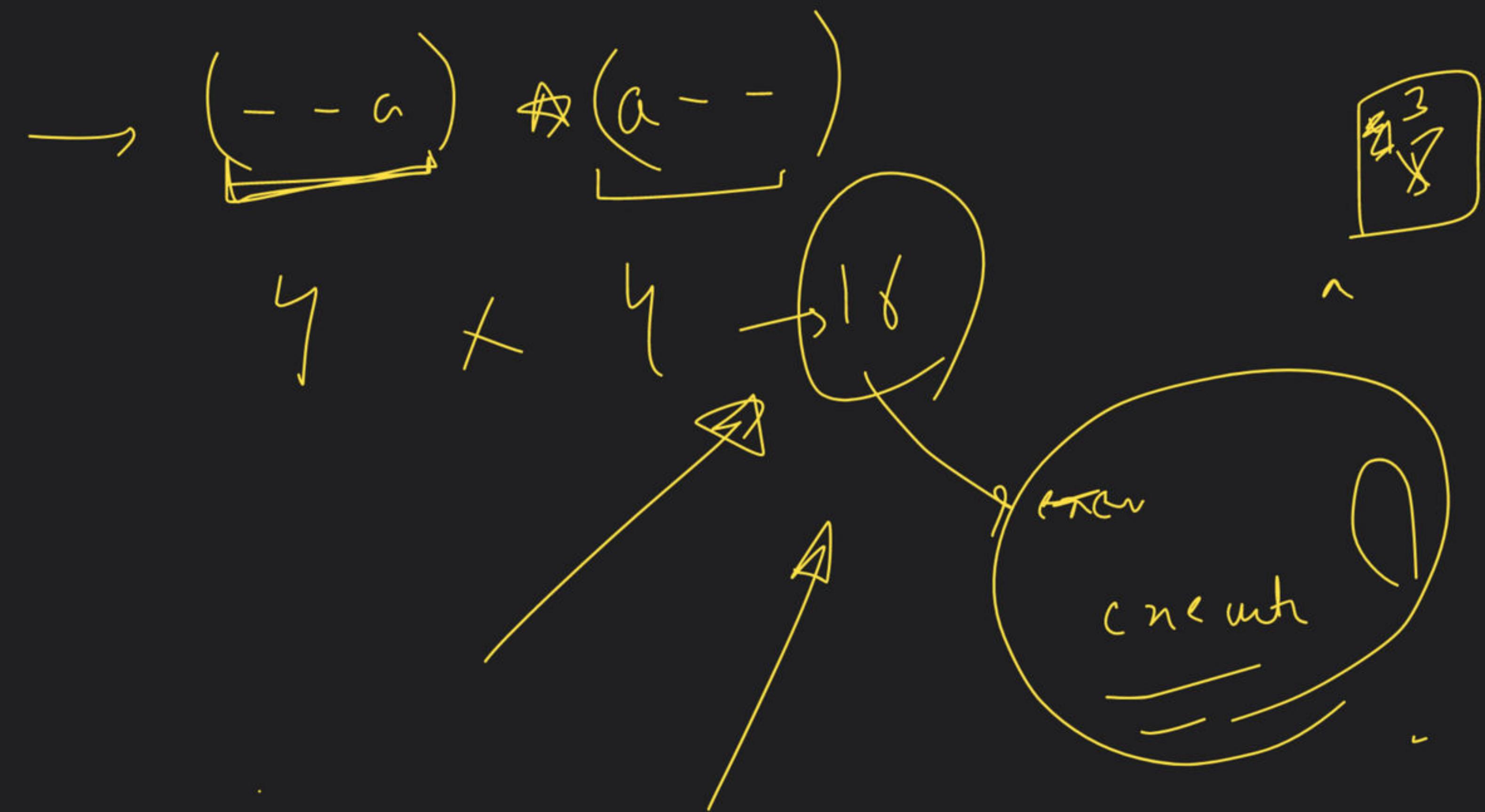
$$y \times y_0$$

$$160$$

$$(- - a) \star (\underline{b++})$$

$$24$$

$n=5$



g 9 9 9 9 - - - - g

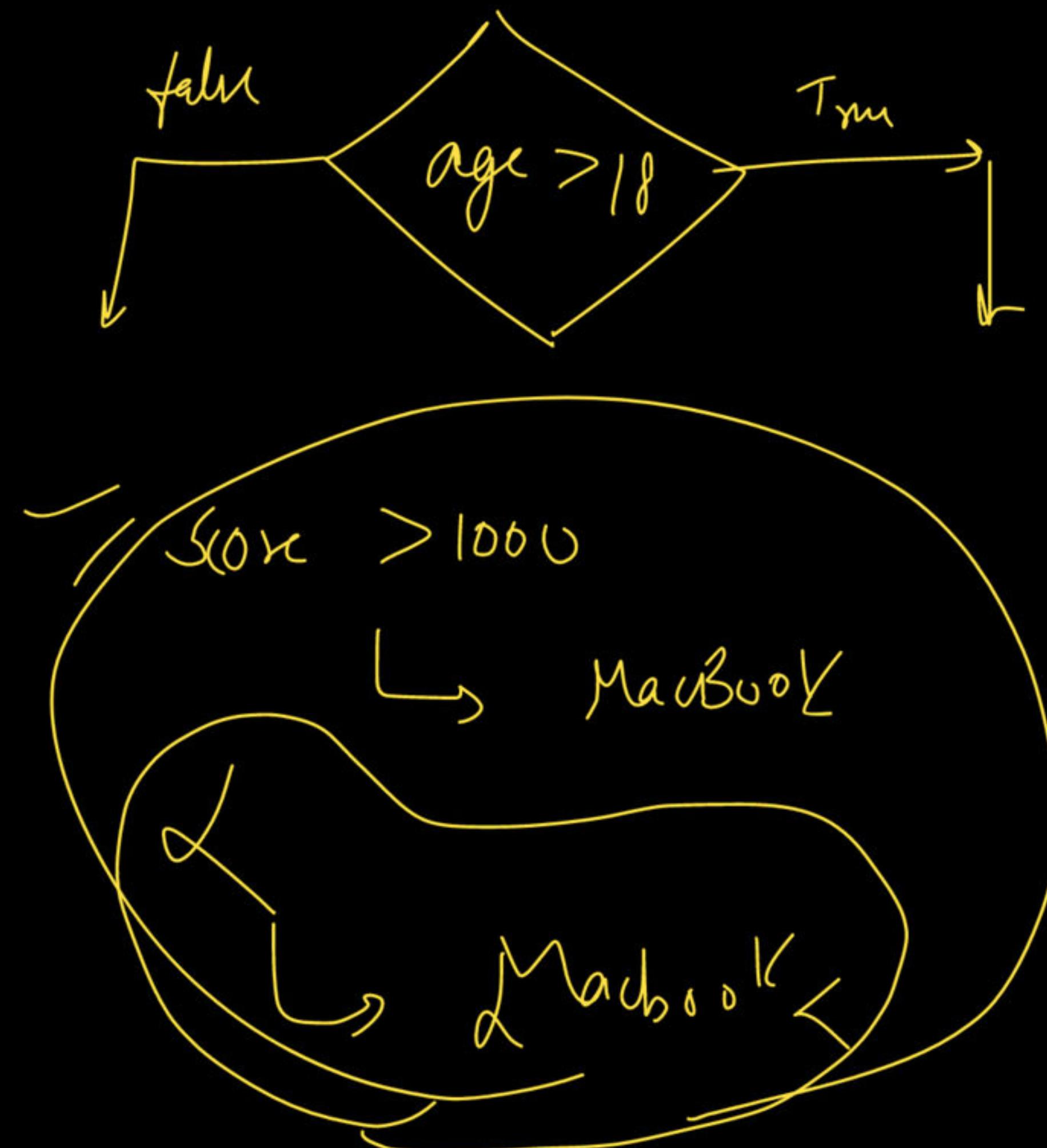


| 3



Conditionals:

- Basic if block
- If-else block
- If-elseif-else block
- Nested if else
- Switch case
- Ternary Operator



-if.usr block

if Condition →
{

====

}

usr

{====

}

Switch

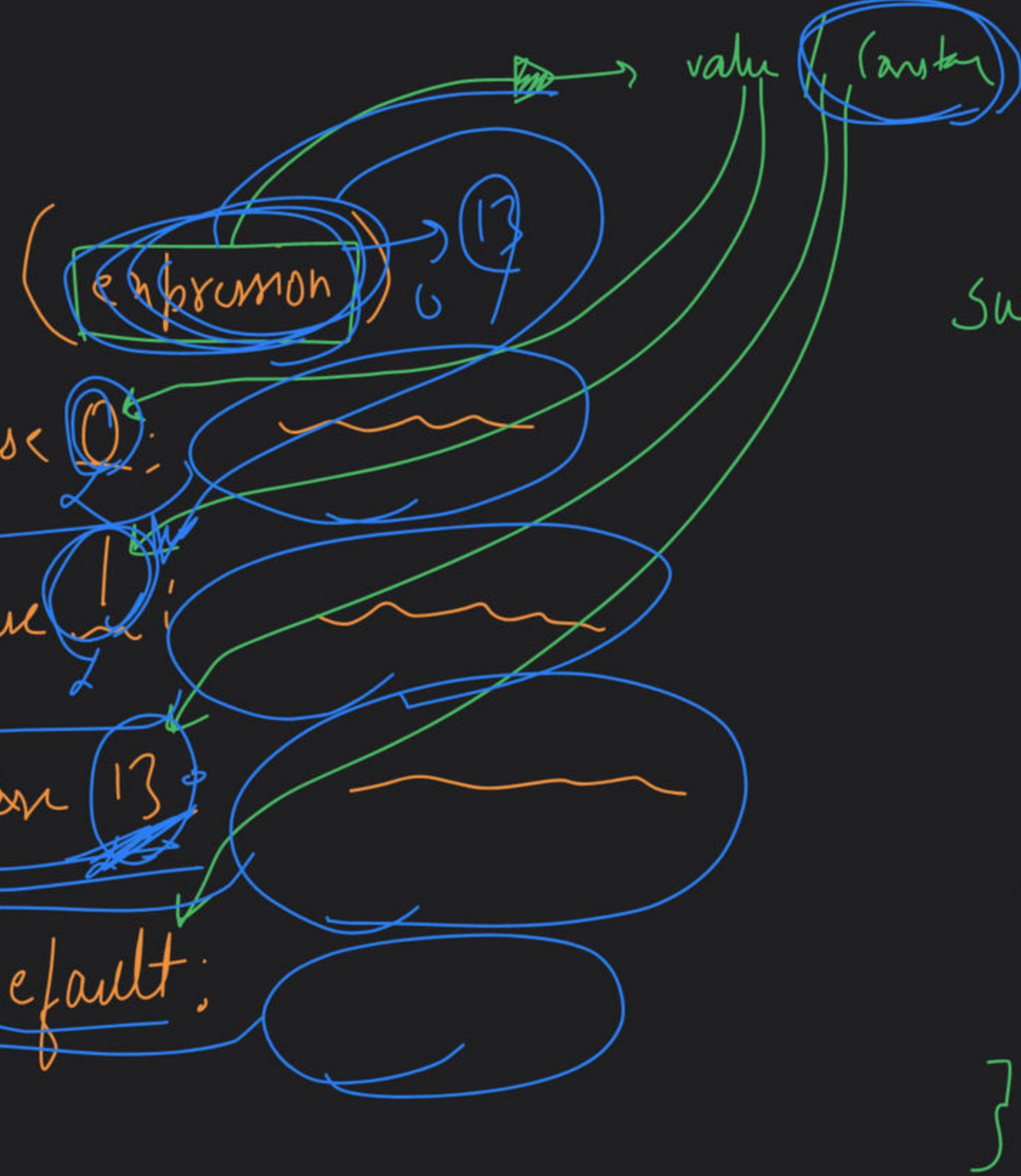
switch

{ case

case 0:

case 1:

default:



1, 2, 2, 9, 15, 6, 7

switch (index)

case 0:

cout << "Monday"

case 1:

cout << "Tuesday"

default:

cout << "Sunday"

short circuiting

if ($C_1^T \neq \infty$ & $C_2 \neq \infty$ & $C_3 \neq \infty$ & $C_4 \neq \infty$)

{
 $C_{out} \ll \text{ "low" } \theta_j$ 
}

Switch (index)

{

$\text{case } 0 : \{ \text{int } a = 15;$

}

}

Switch (expression) \rightarrow Kya Kya \rightarrow Rules



Rules

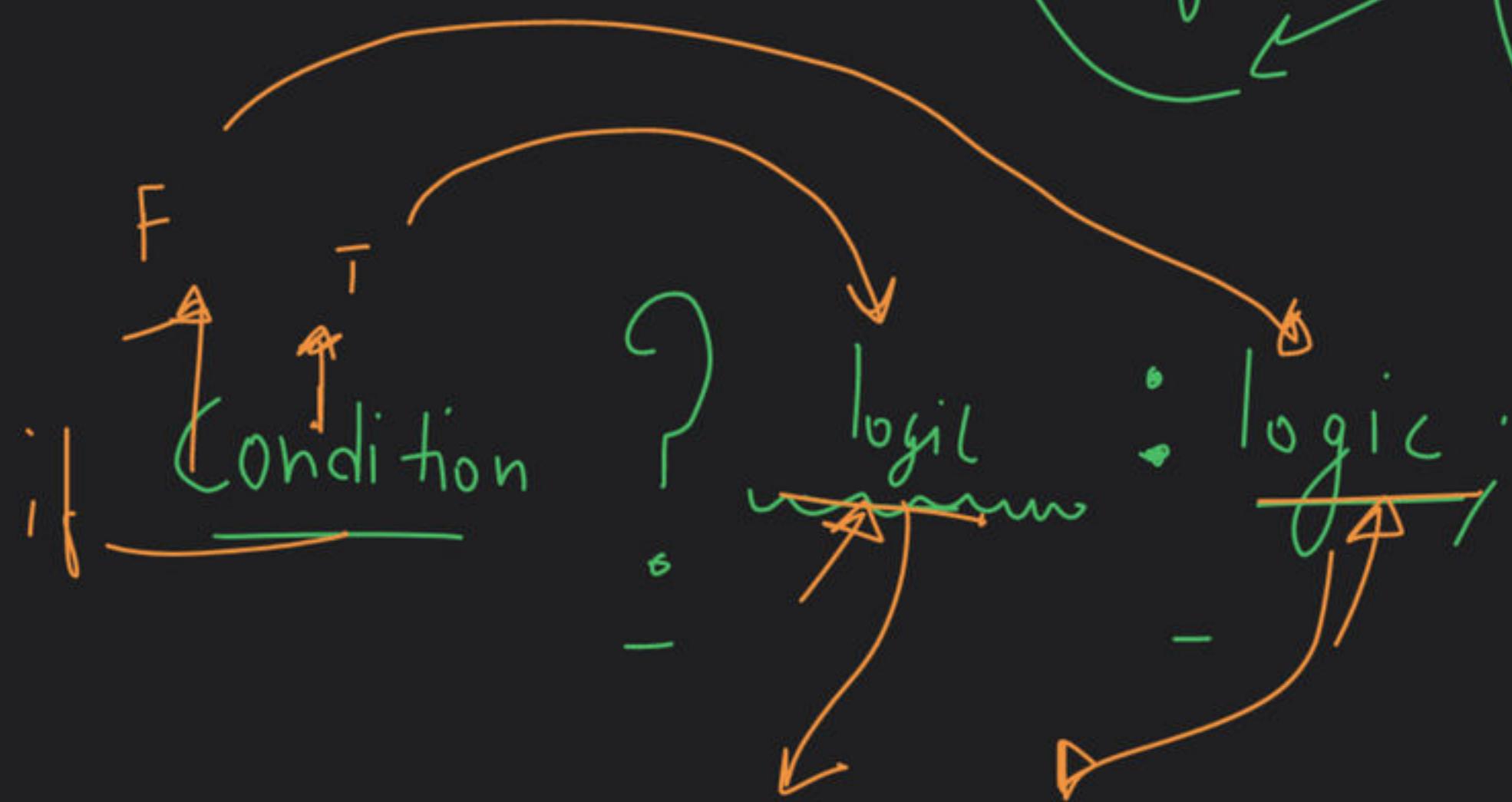
Video \leftarrow Switch
Case

Case
 $C + L$

Ternary

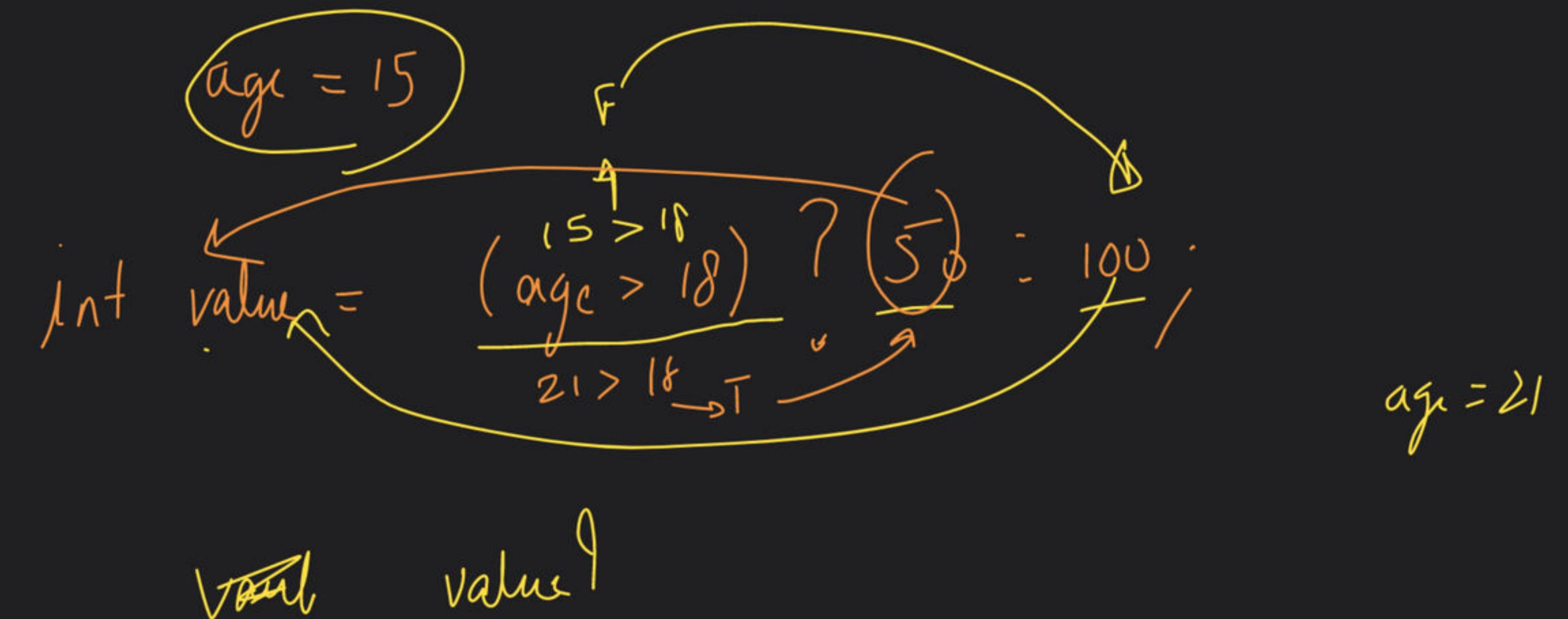
Operator

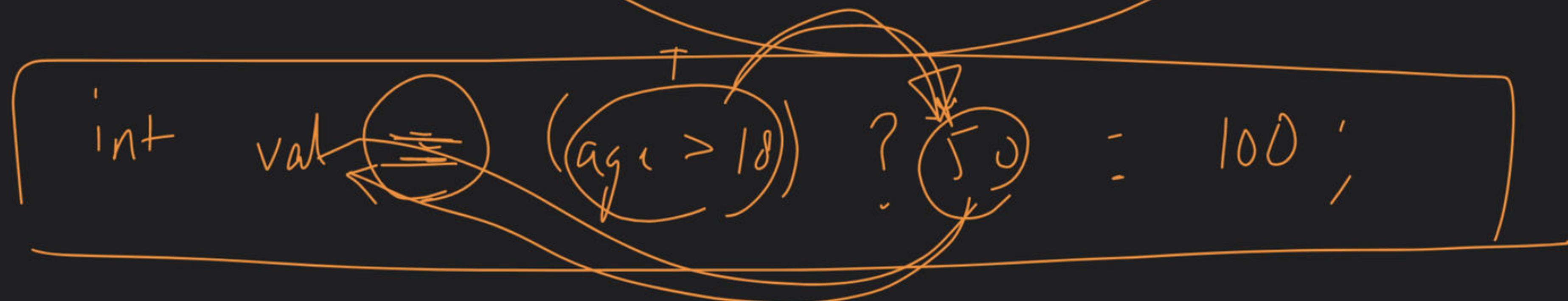
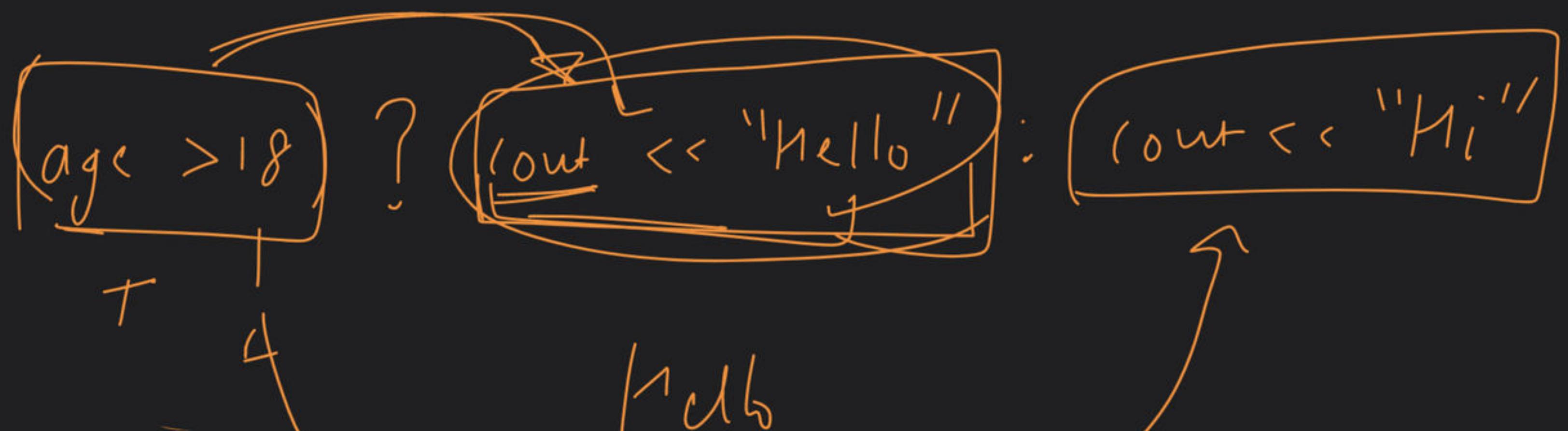
Syntax



if
 $\{ \underline{\text{sign} > 1000} \}$
 $\text{cout} \ll \text{"MB"};$

else
 $\text{cout} \ll \text{"No MB"};$





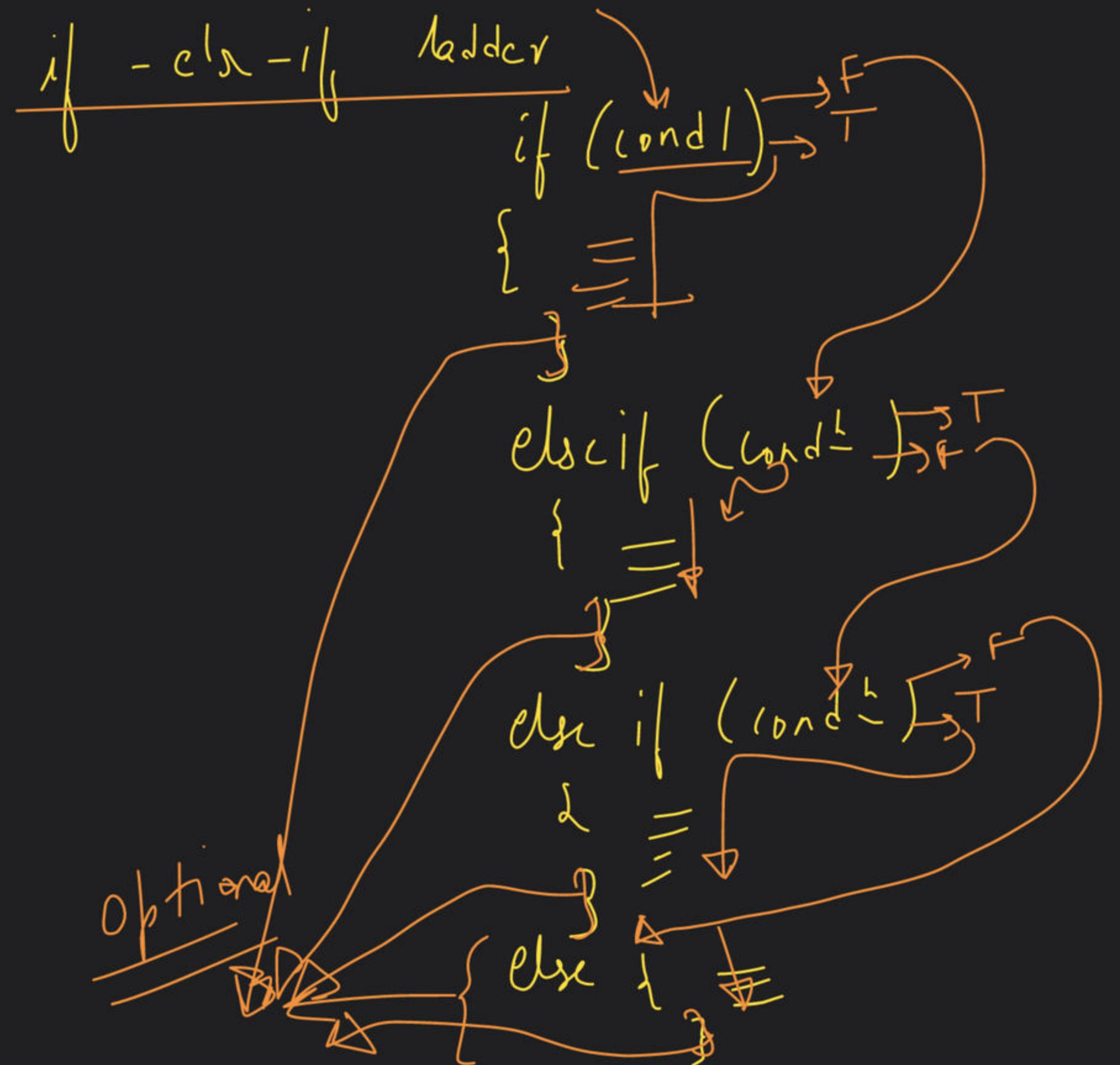
`cout << val;` → 50

Val



```
if ( )  
{  
    if ( )  
    {  
        if ( )  
    }  
}
```

```
if ( )  
{  
    if ( )  
    {  
        if ( )  
    }  
}  
else  
{  
    if ( )  
    {  
        if ( )  
    }  
}
```



short - circuiting

if

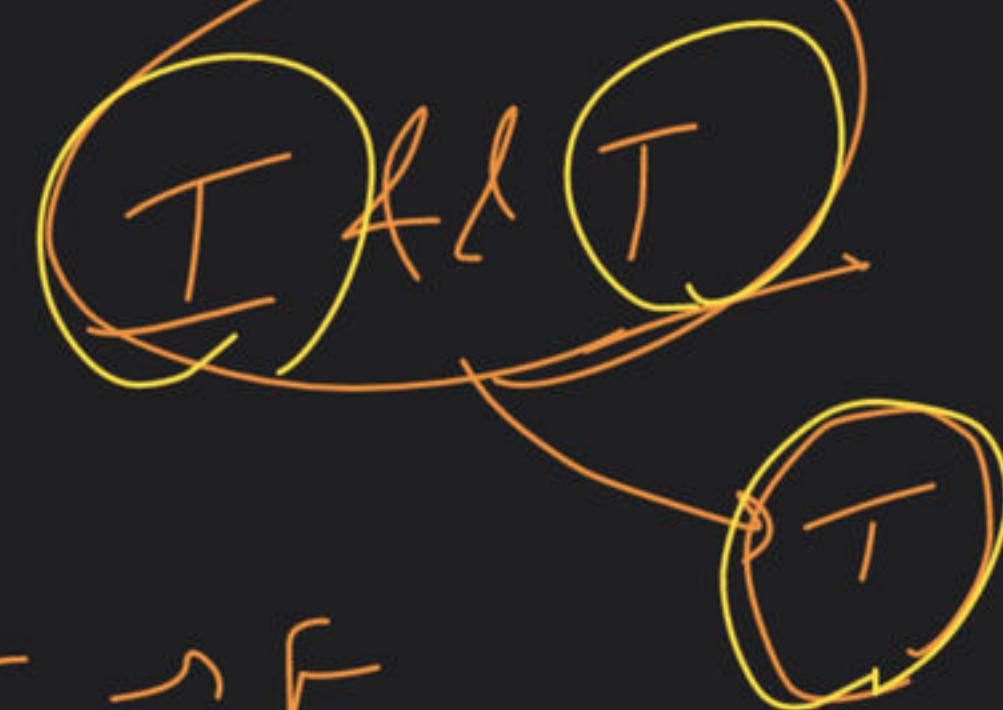
{

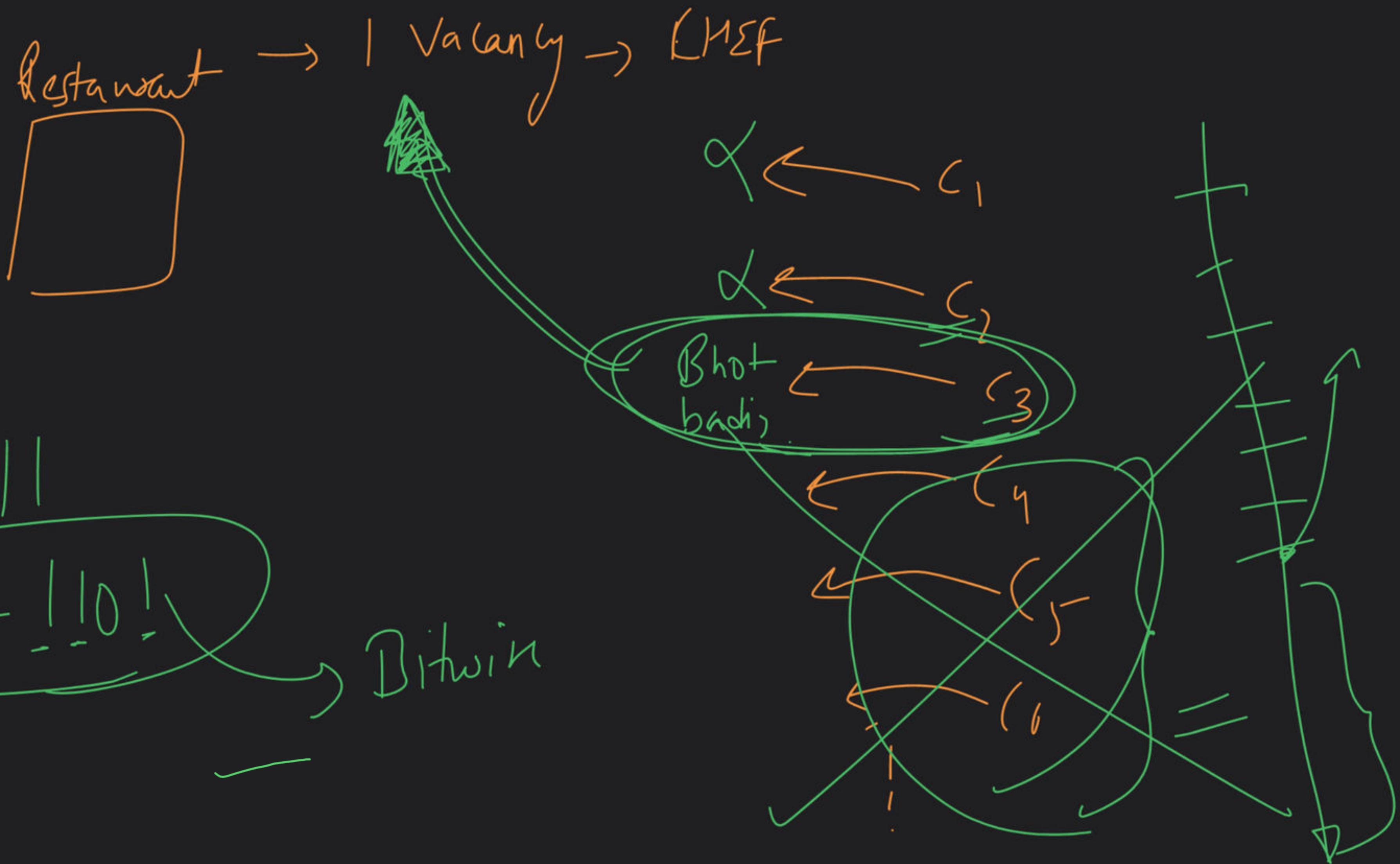
}

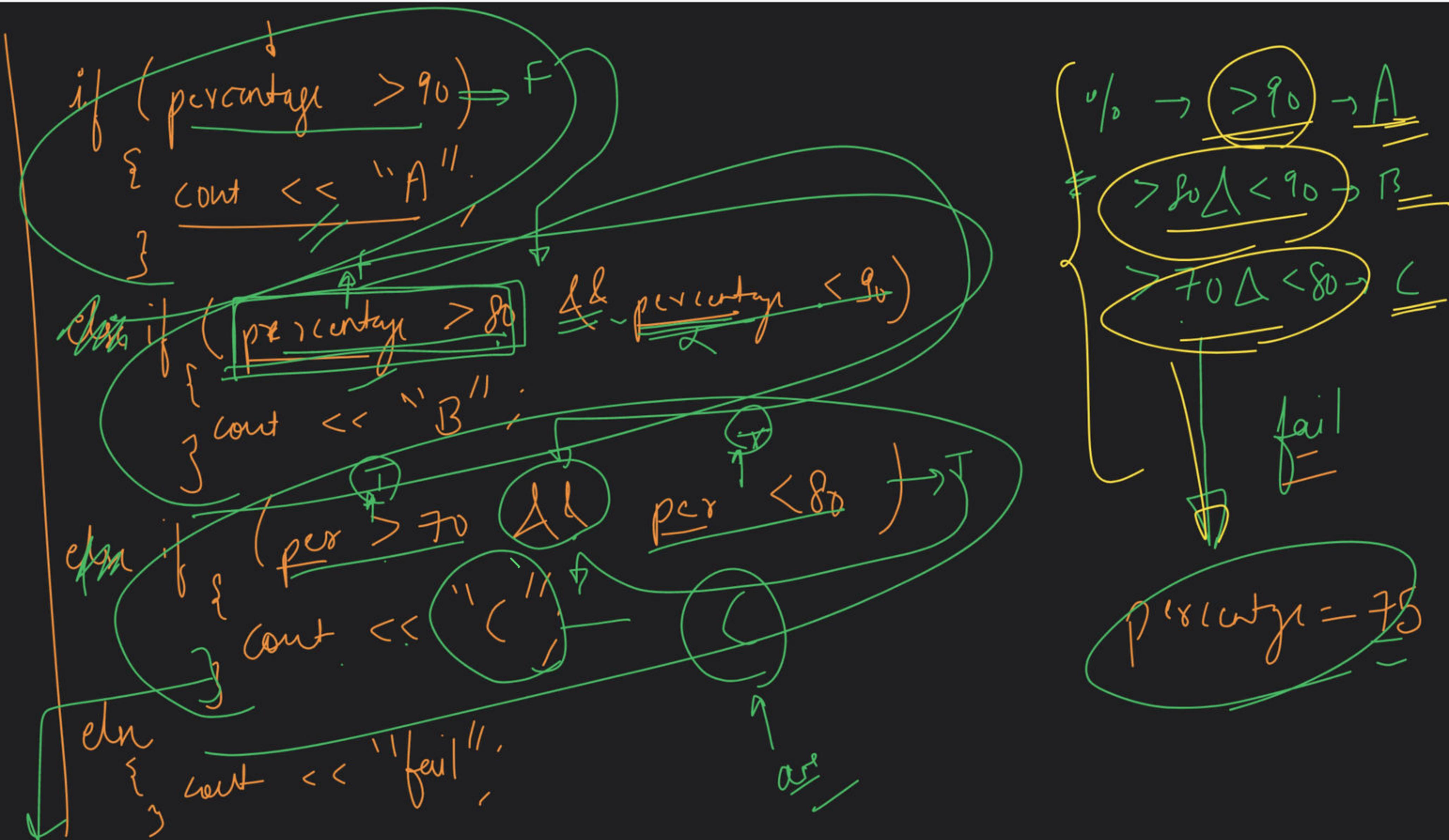
(out << "Love")

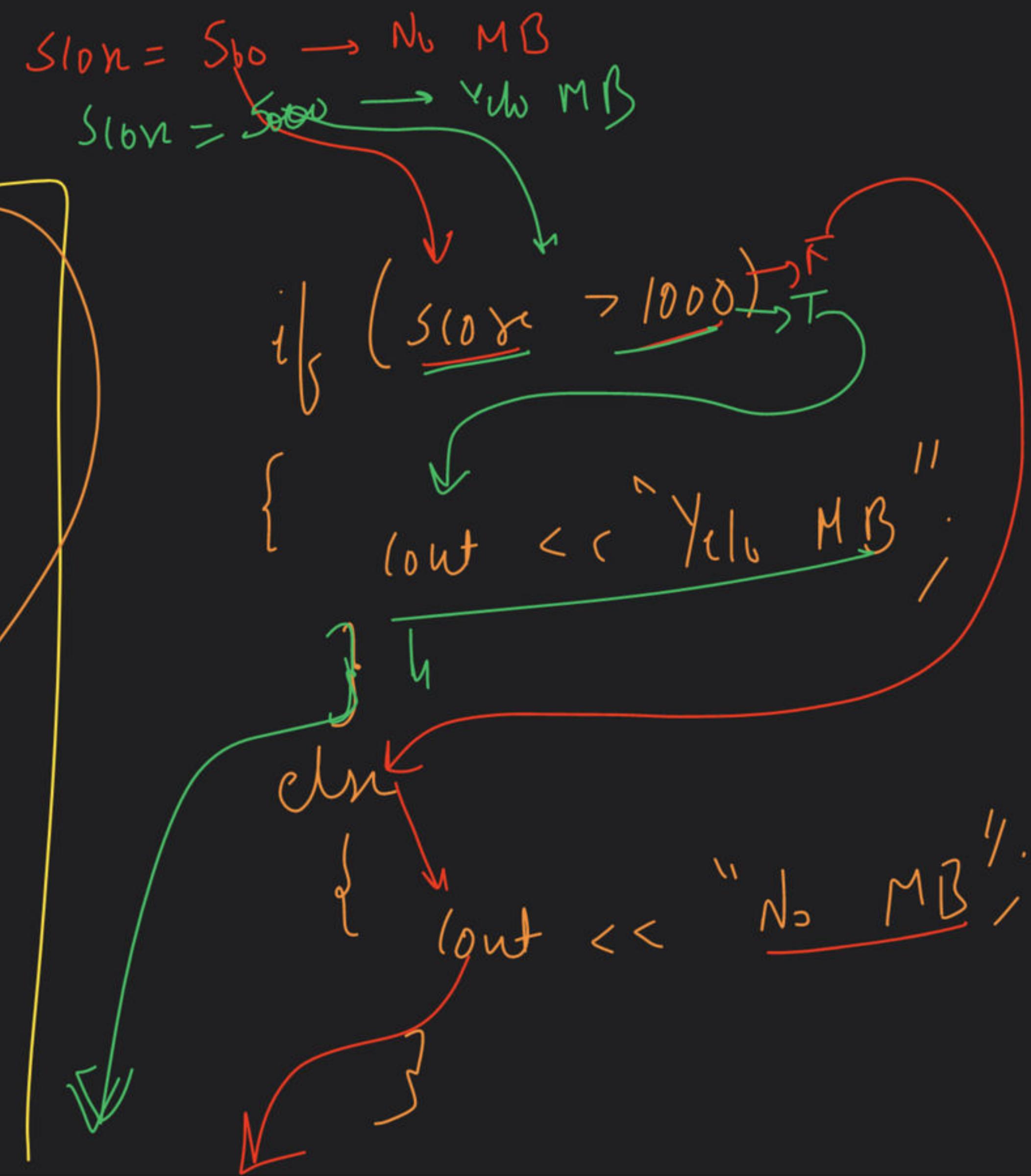
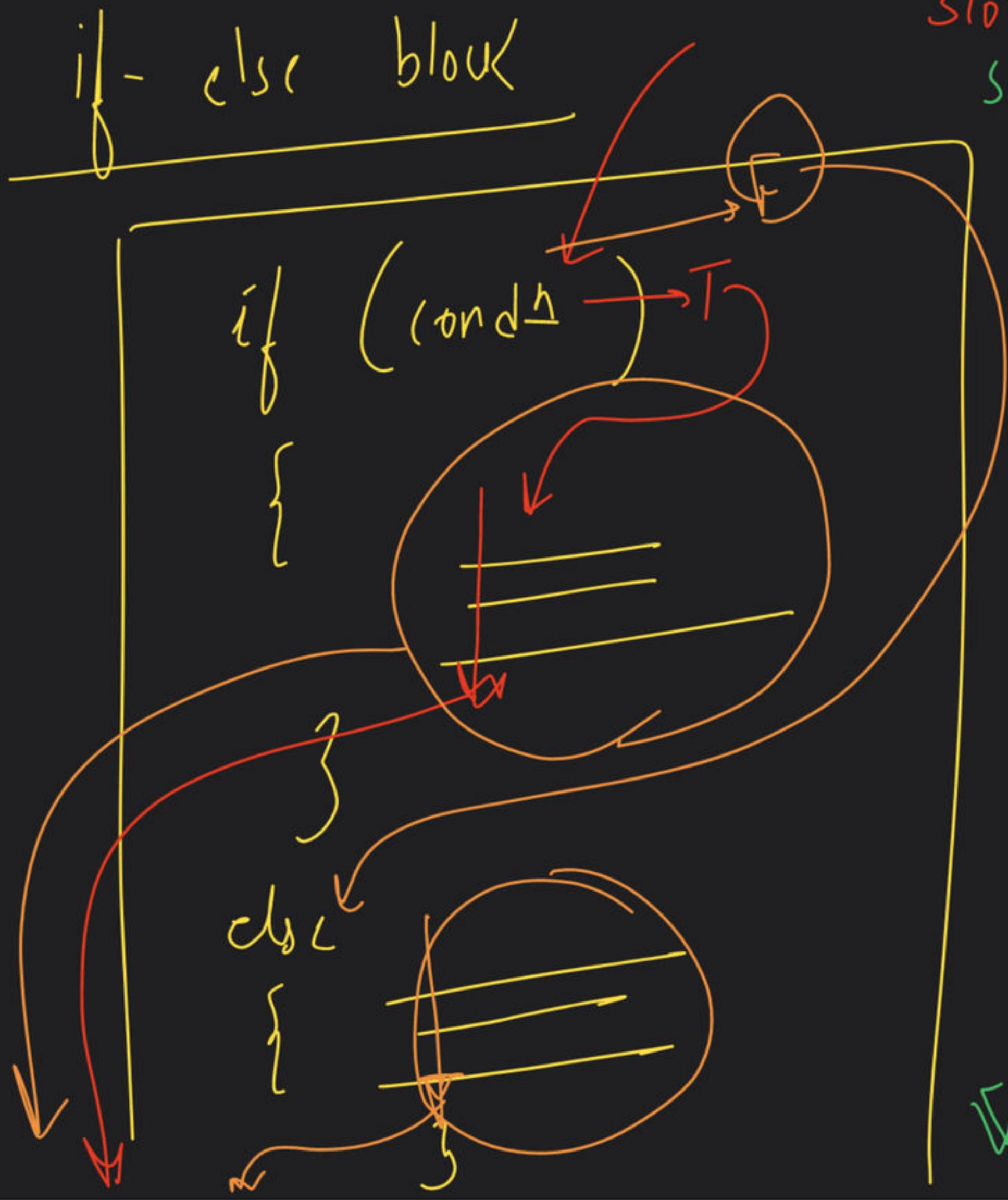
F&T>F

F&F>F T&F>F









```
if (age >= 15)
{
    cout << "You look "
}
```

```
if (slope > 1000)
{
    cout << "Yellow MacBook"
}
```

```
if ( age >= 18 & & scon > 1000 )  
{  
    cout << "Bike";  
}
```

لطفاً

Bitwise AND op

The logo of the University of Göttingen, featuring a circular emblem with three interlocking rings and the letters 'U' and 'G'.

logical
AND op by

2113

T & J

203

203

t

1

Basic - if Block

Syntax →

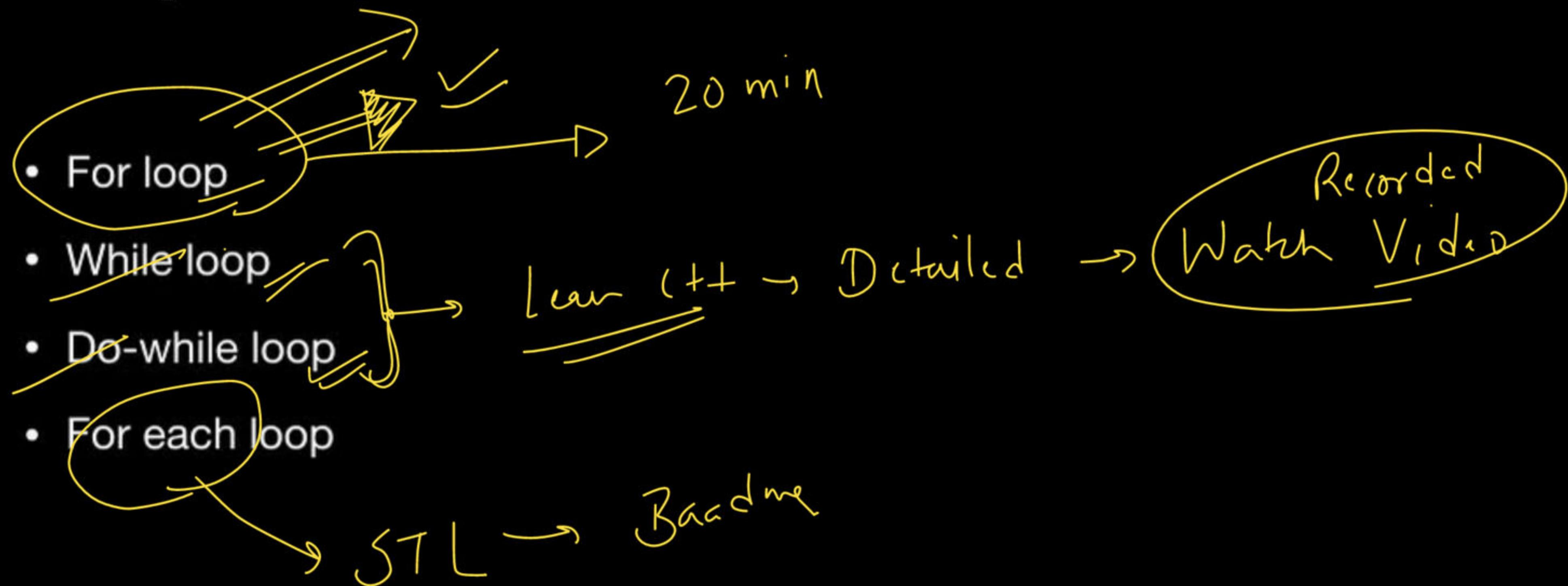
if (Condition)
multiple / join -> operation

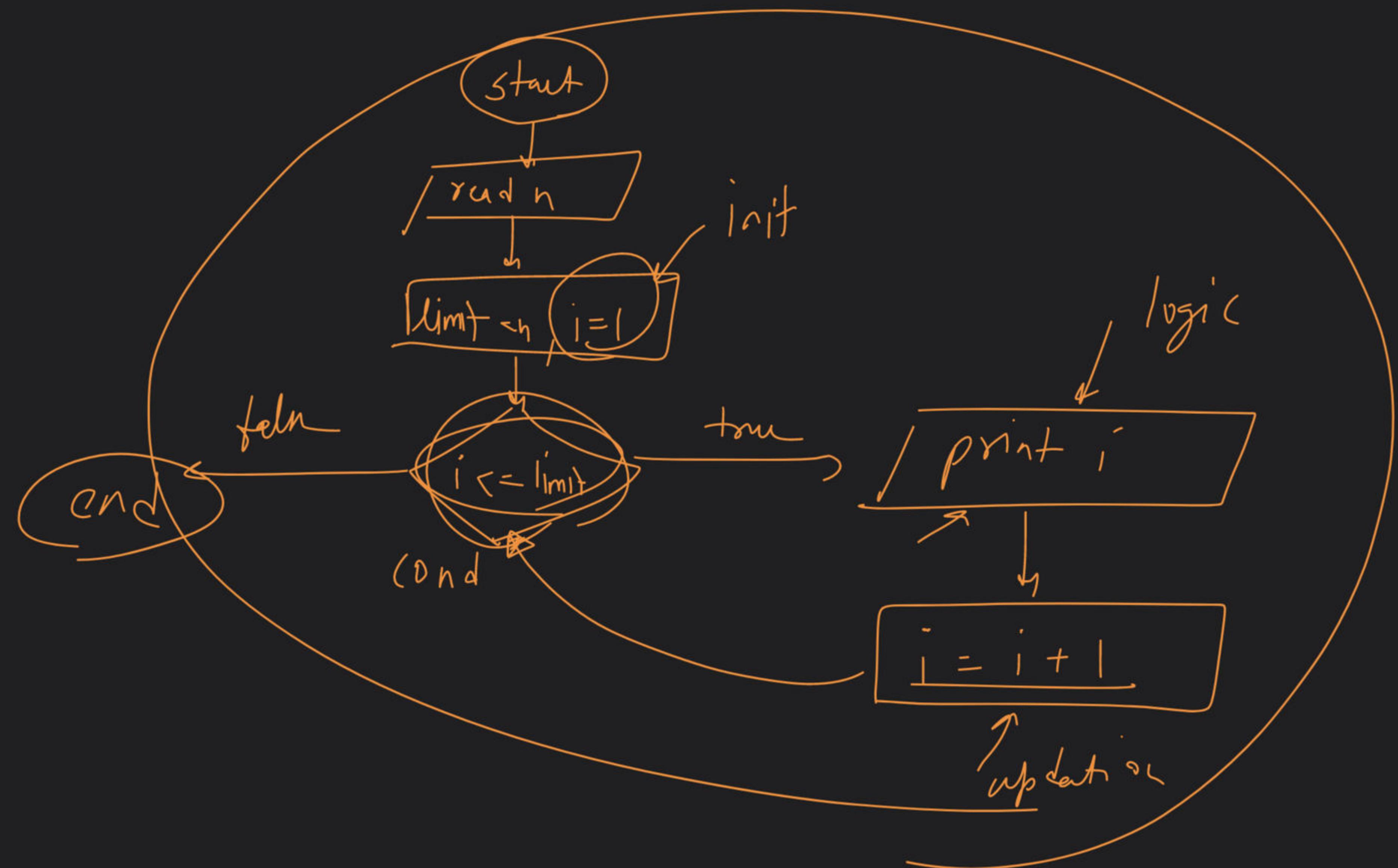


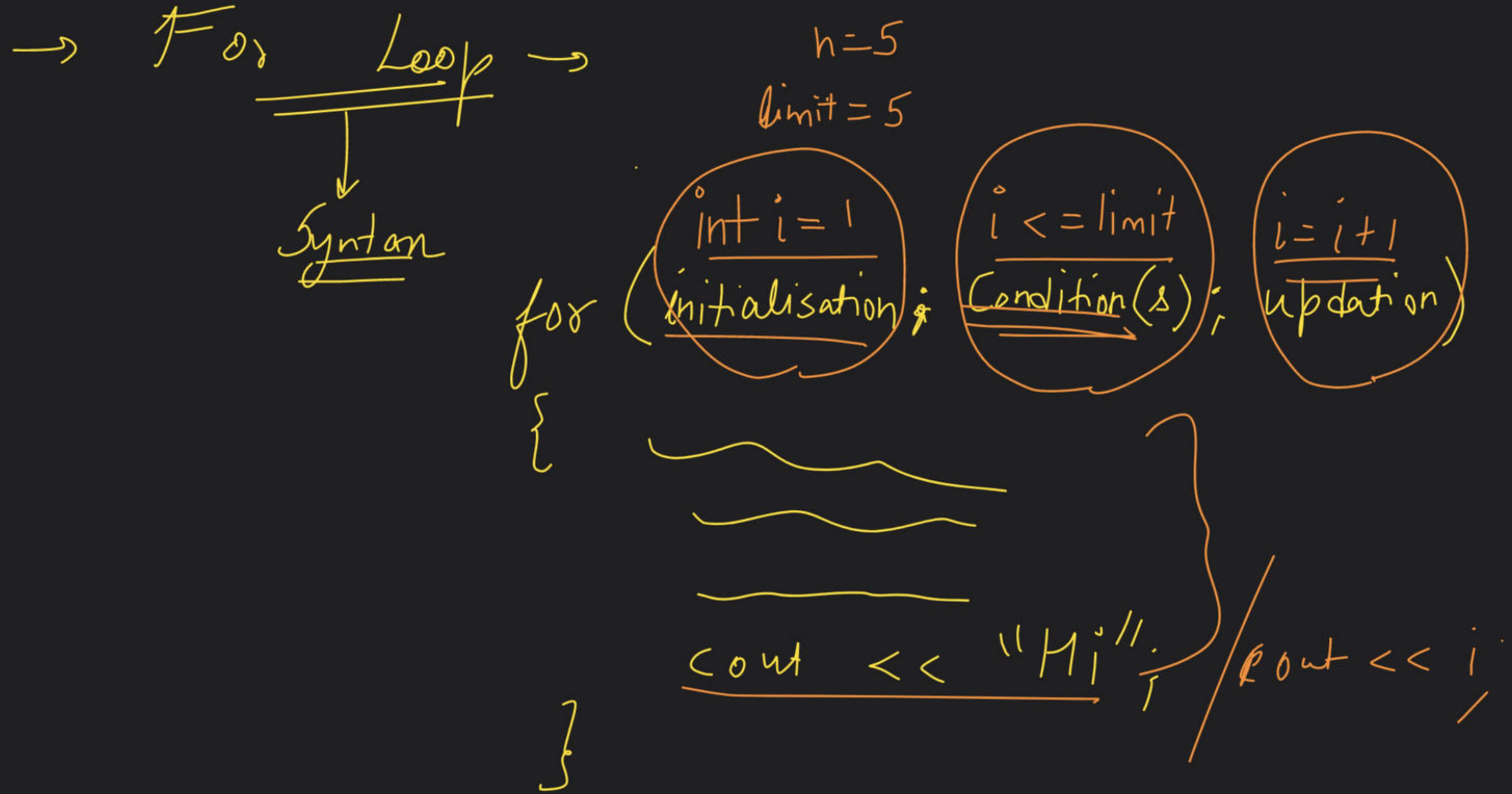
Variable Scoping

~~functions~~

Loops:







for (int i=1; i <= 5; i = i+1)

{

cout << i;

i = i+1

or
++i ✓

or
i++ ✓

$$i = 1$$

$$1 \leq 5$$

T

Print 1

$$i = 1 + 1 = 2$$

$$2 \leq 5$$

T

Print 2

$$i = 2 + 1 = 3$$

$$3 \leq 5$$

T

Print 3

$$i = 3 + 1 = 4$$

$$4 \leq 5$$

T

Print 4

$$i = 4 + 1 = 5$$

$$5 \leq 5$$

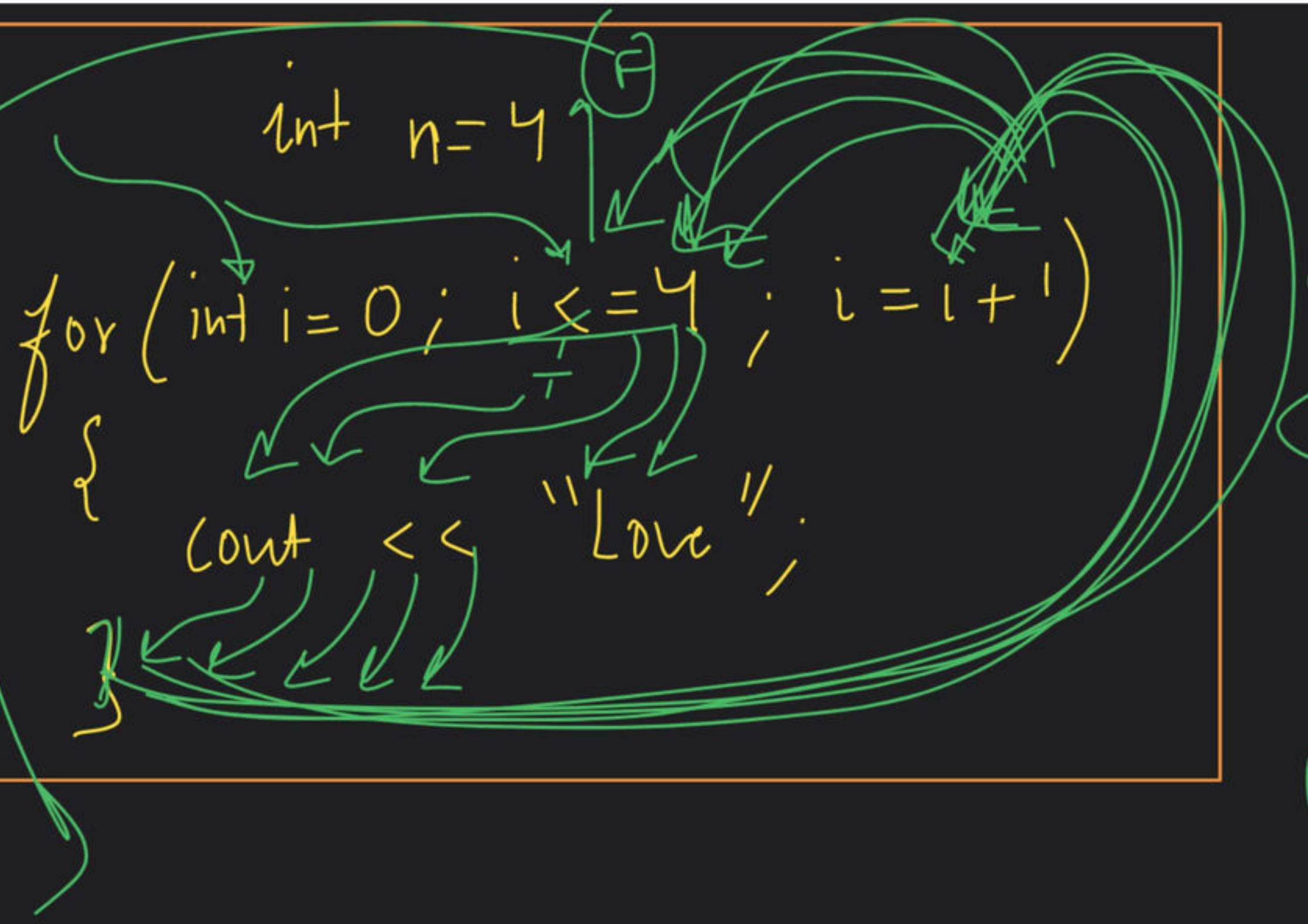
T

Print 5

$$i = 5 + 1 = 6$$

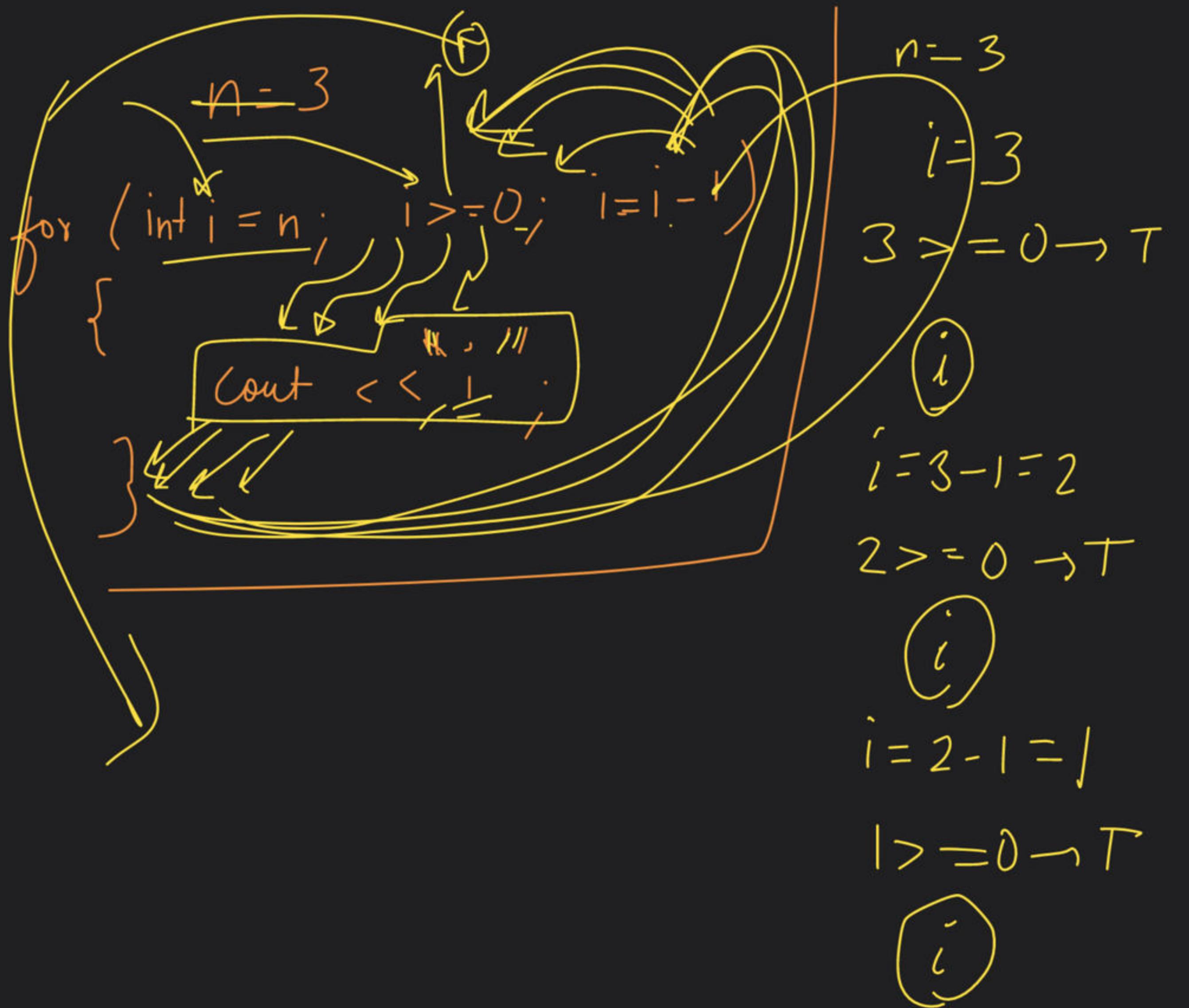
$$6 \leq 5$$

T

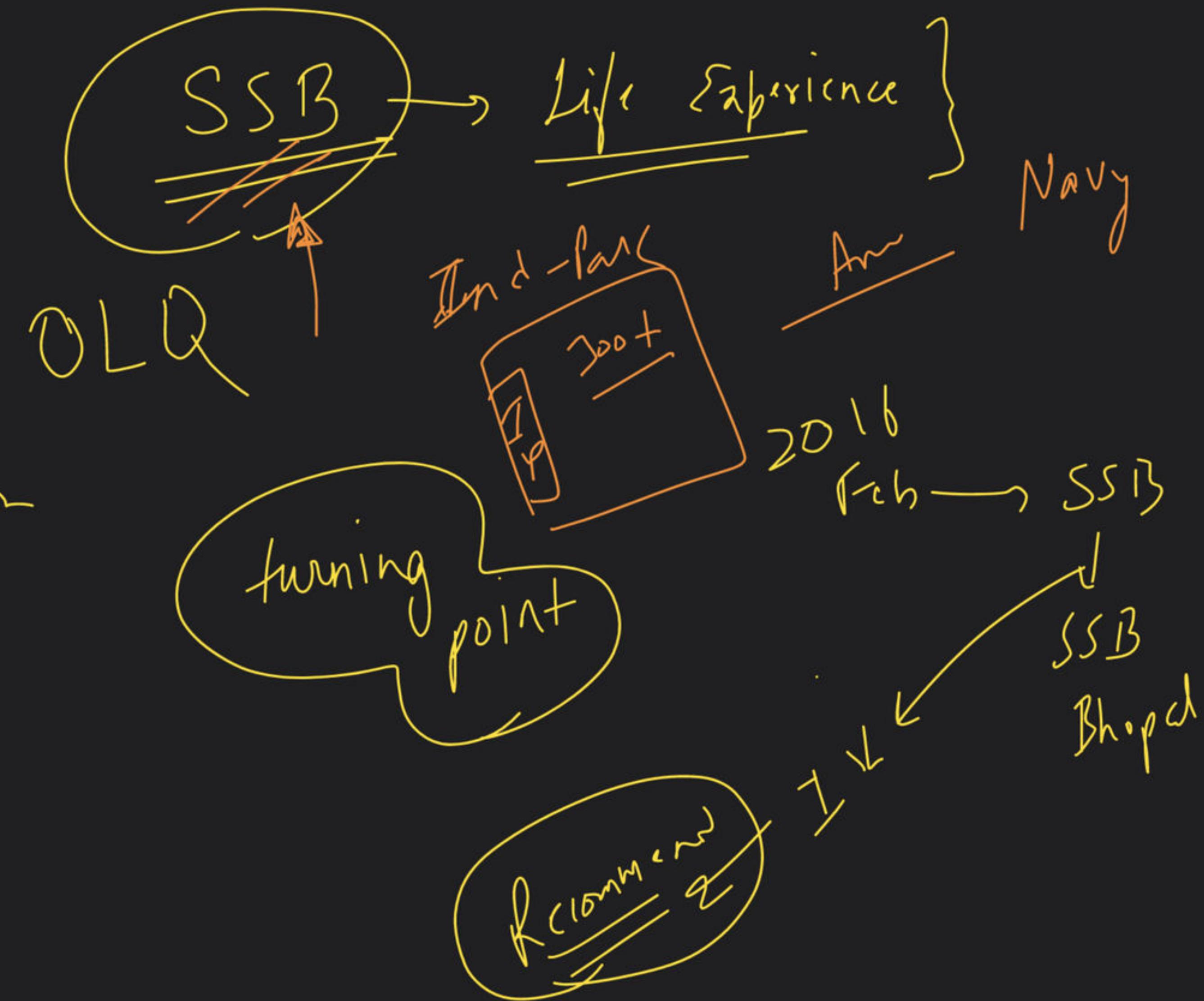
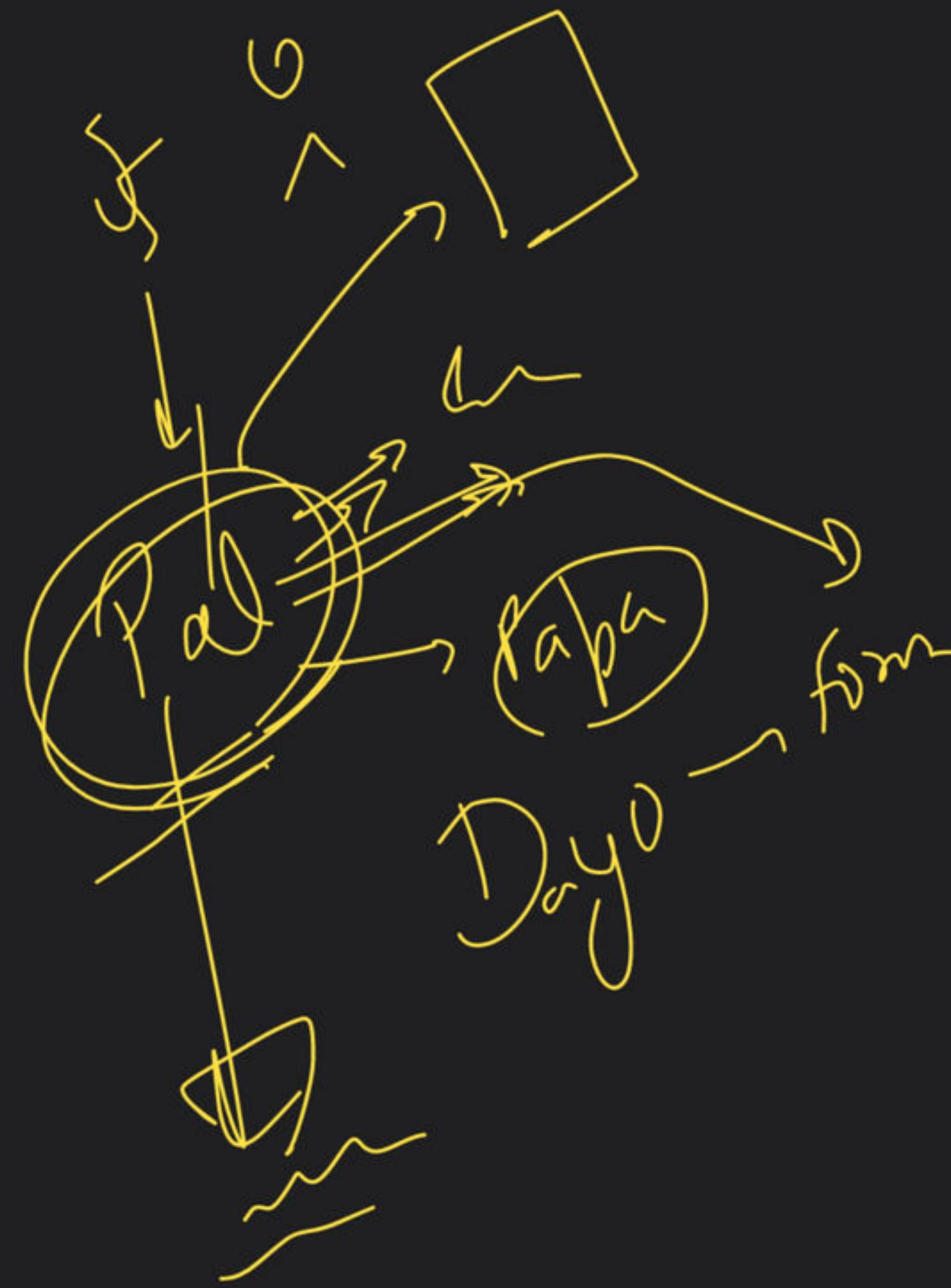


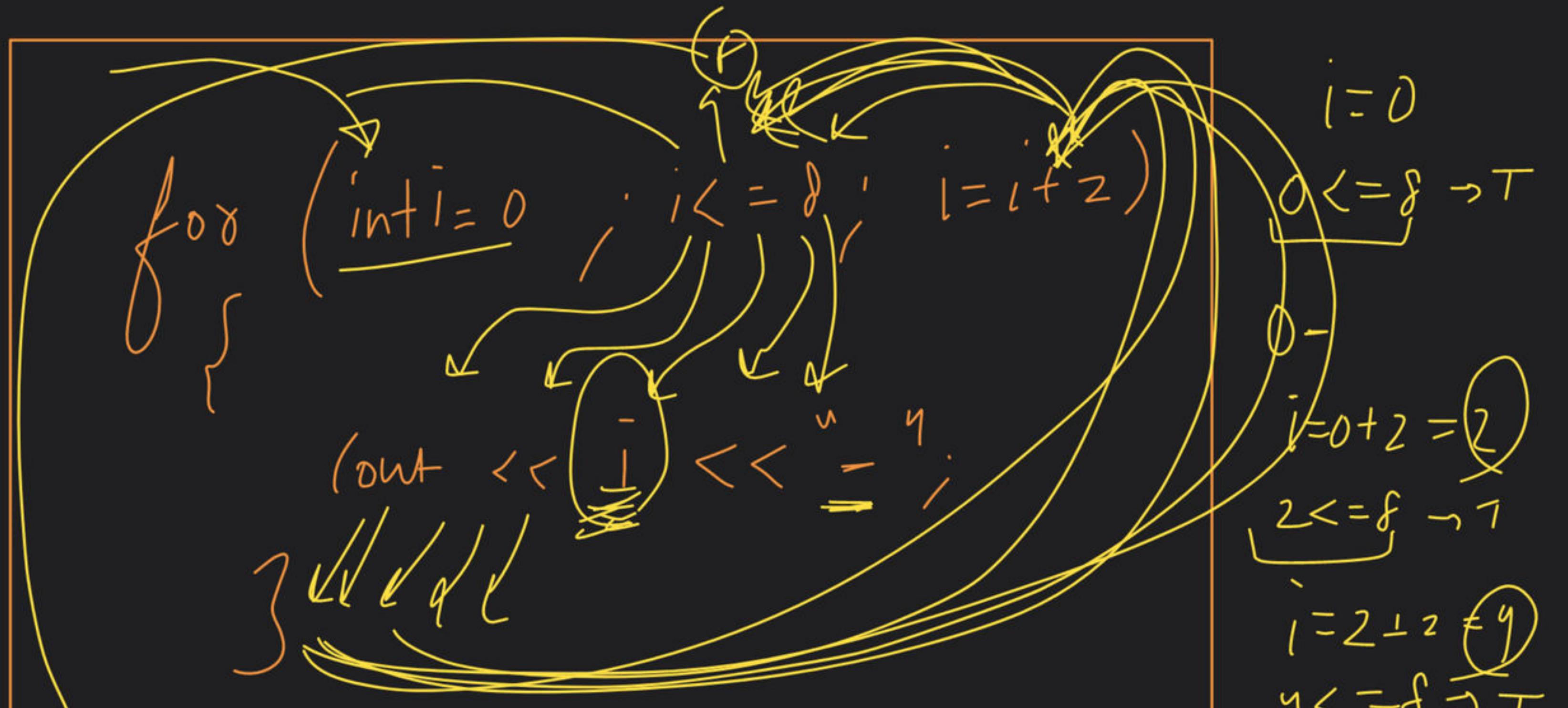
$n = 4$
 $i = 0$
 $0 \leq 4 \rightarrow T$
 Love
 $i = 0 + 1 = 1$
 $1 \leq 4 \rightarrow T$
 Love
 $i = 1 + 1 = 2$
 $2 \leq 4 \rightarrow T$
 Love

$i = 2 + 1 = 3$
 $3 \leq 4 \rightarrow T$
 Love
 $i = 3 + 1 = 4$
 $4 \leq 4 \rightarrow T$
 Love
 $i = 4 + 1 = 5$
 $5 \leq 4 \rightarrow F$



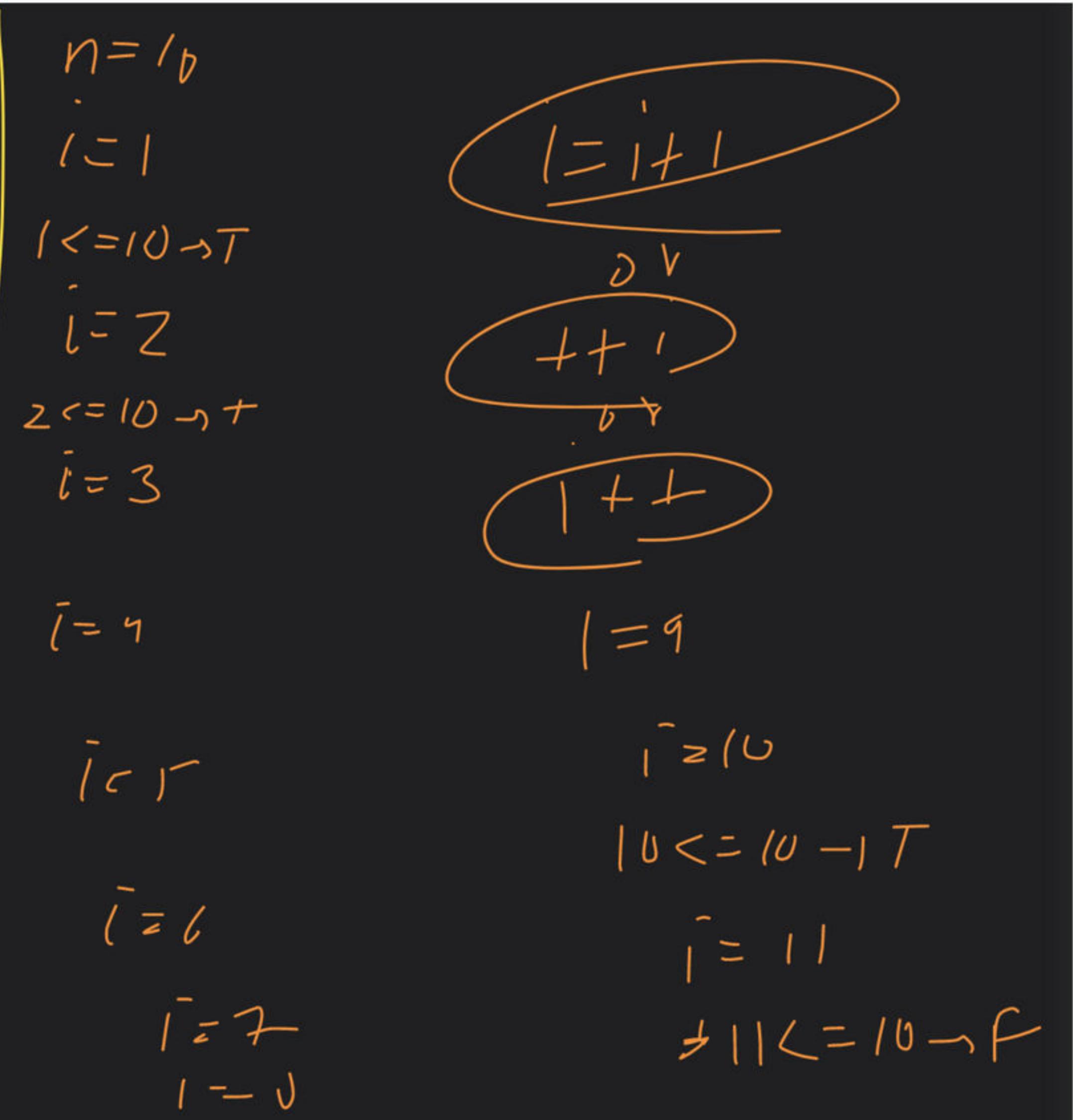
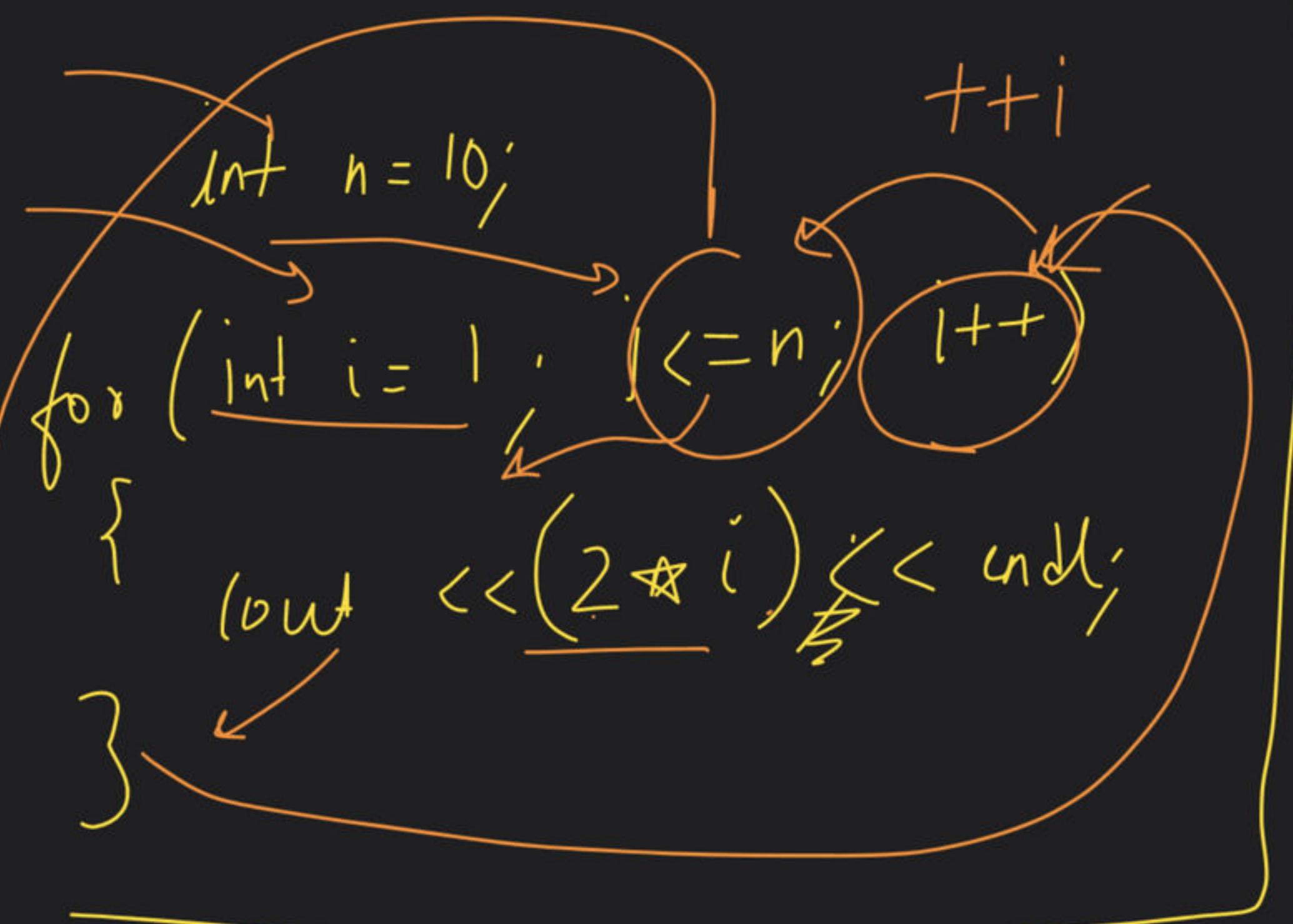
$i : i - 1 = 0$
 $0 >= 0 \rightarrow +$
 i
 $i = 0 - 1 = -1$
 $-1 >= 0 \rightarrow F$





$0 / p \rightarrow 0 - 2 - 4 - 6 - 8 -$

$$\begin{aligned}
 i &= 8 + 2 \\
 &= 10 \\
 8 &<= 8 \\
 &\quad \downarrow \\
 i - 8 + 2 &= 10 \\
 10 &<= 8 \\
 &\quad \downarrow \\
 &\quad \downarrow \\
 &\quad \downarrow
 \end{aligned}$$



Pathway

