

# Programming Fundamentals

Special class

# Programming Fundamentals

Instructor: Love Babbar

# Thought Process to solve a Problem

① Understand the Problem

② i/p values

③ Approach

$$\text{a} + \text{b}$$

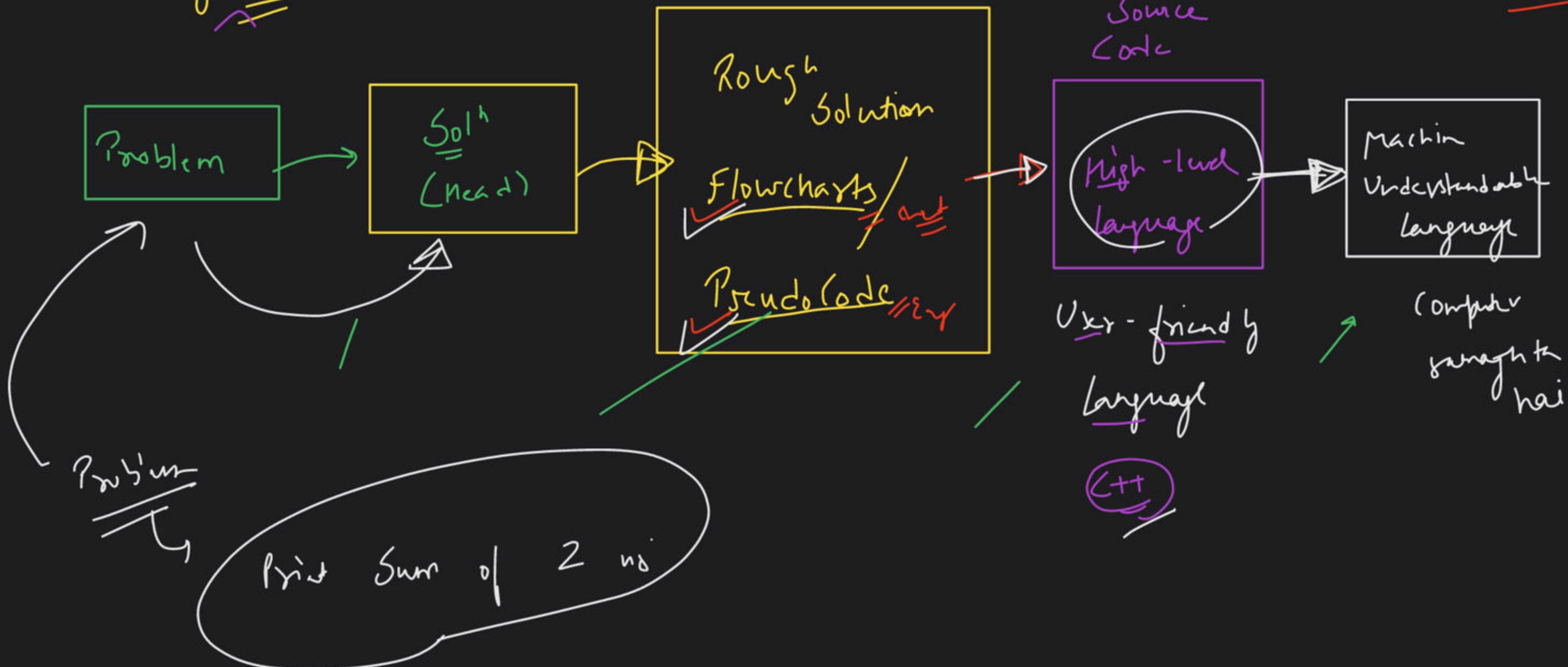
$$v^2 - u^2 = 2 \text{ as}$$

$$\rightarrow \text{avg} = \frac{a+b}{2}$$

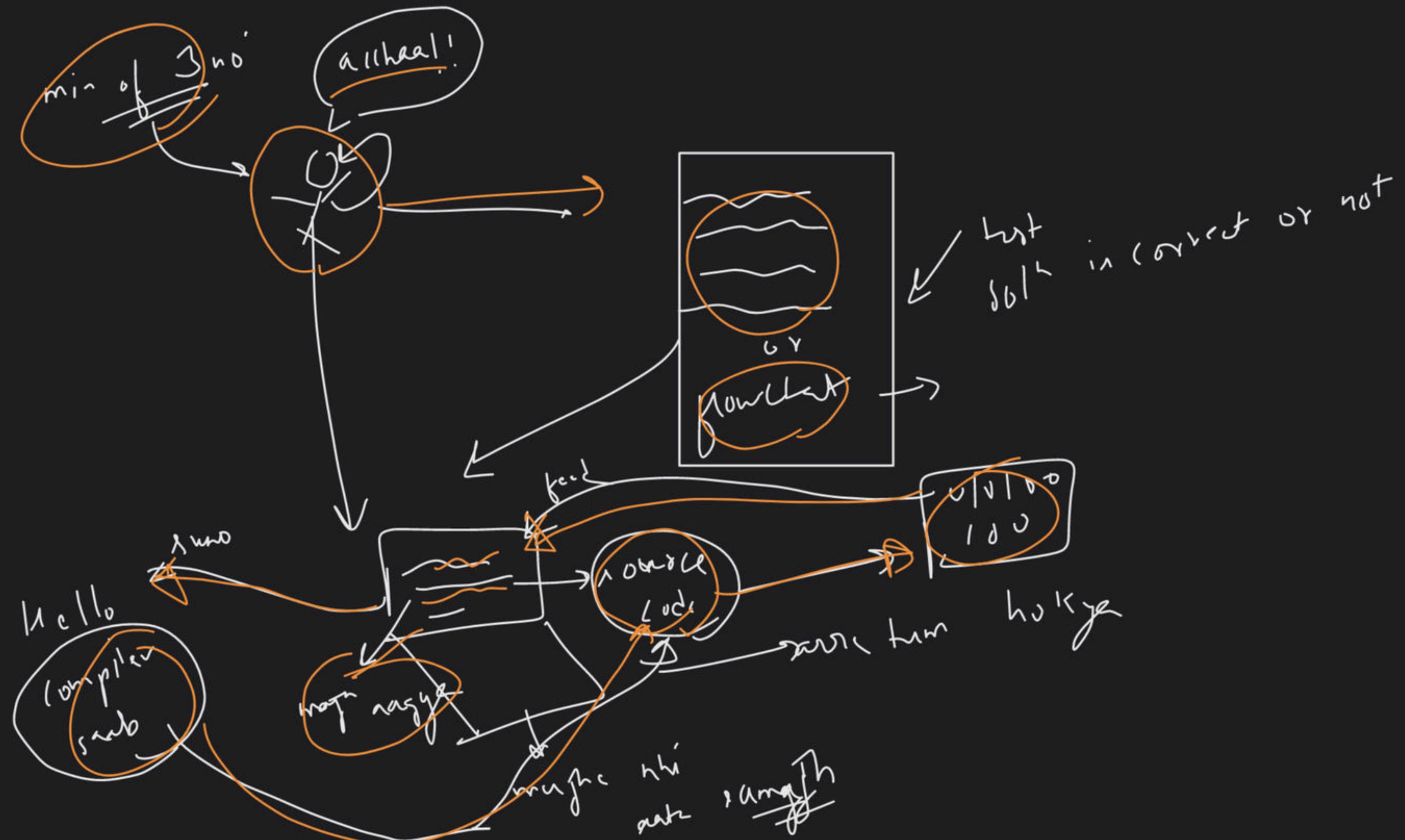
$$\text{i/p} \rightarrow a=3 \\ b=7$$

$$\begin{aligned} & \text{MCQ} \\ & \text{avg} = ? \\ & \text{avg} = \frac{7+2}{2} = 5 \\ & \text{units} \\ & \text{m/s} \\ & \text{km/h} \end{aligned}$$

Algorithm  
≡



→ O.I ↓



# Using Computer to solve a Problem:

13 →

$$13/2 \rightarrow \text{rem} \rightarrow 1$$

$$13/3 \rightarrow \text{rem} \rightarrow 1$$

$$13/4 \rightarrow \text{rem} \rightarrow 1$$

$$13/5 \rightarrow$$

⋮

$$13/12 \rightarrow 1$$

1013

Div 1 and 13

divide

Computer

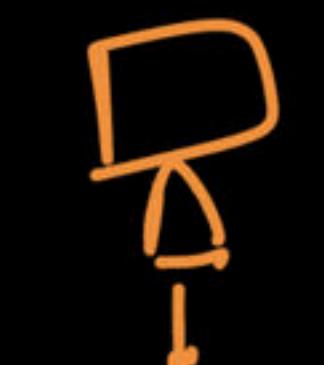
rem. 0  $\neq$   
not prime

13 is a  
prime no

$$\underline{n=13}$$

↓  
Prime or not

{辗转相除法  
或  
Pseudocode



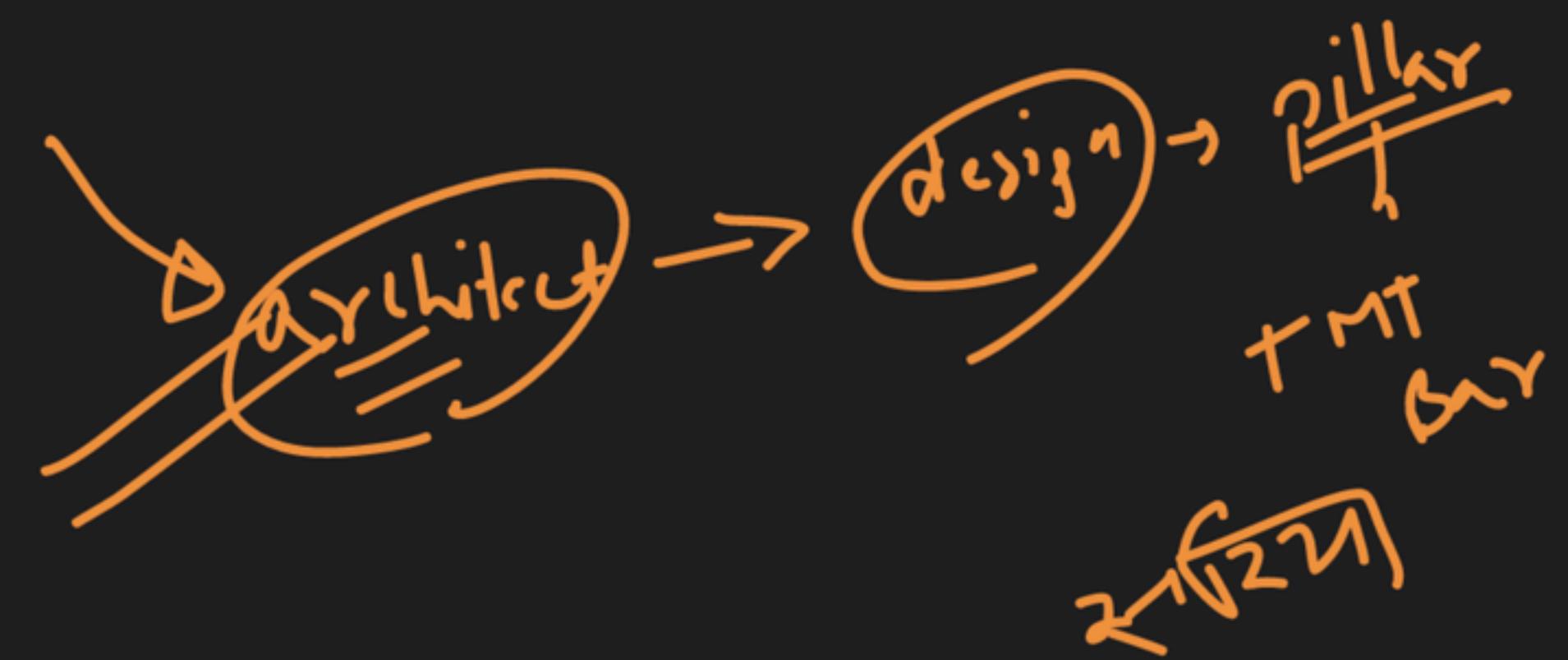
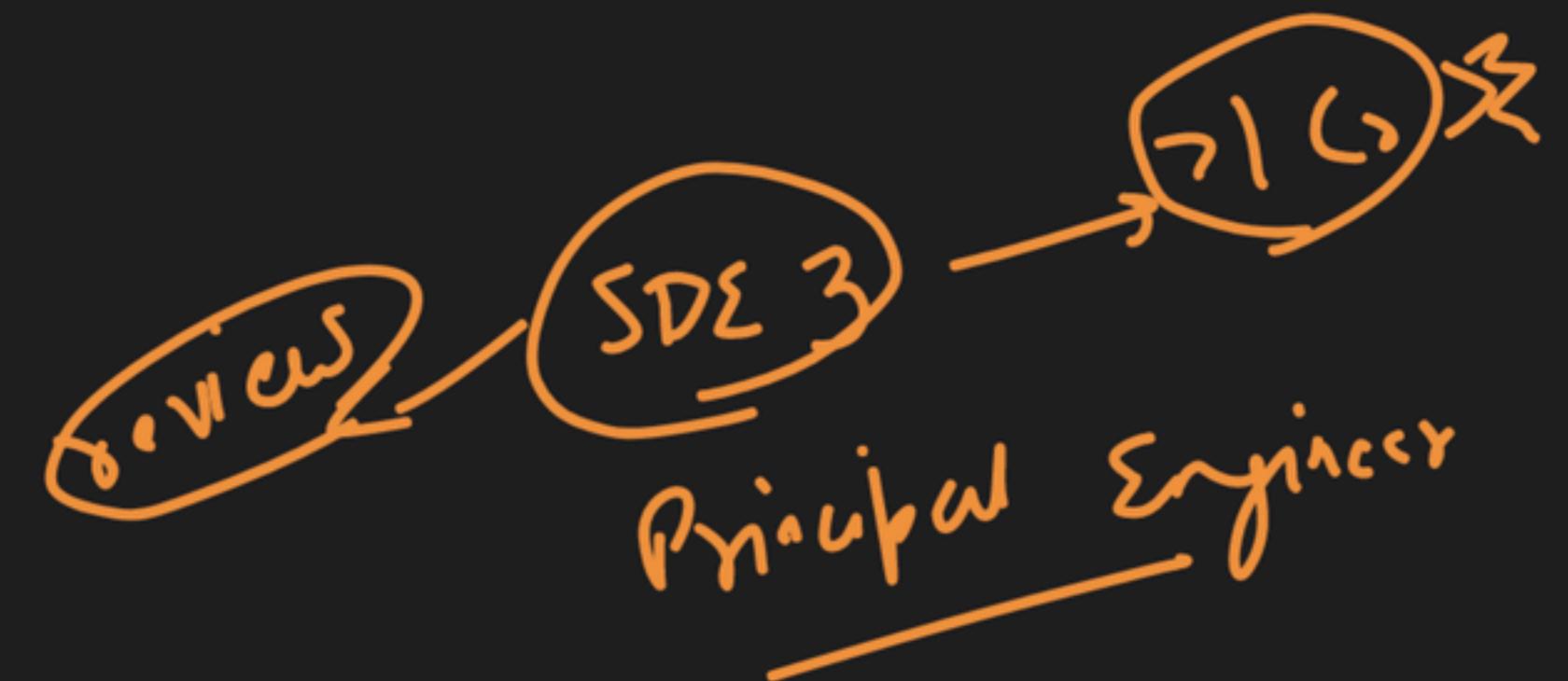
Binary

011

no +

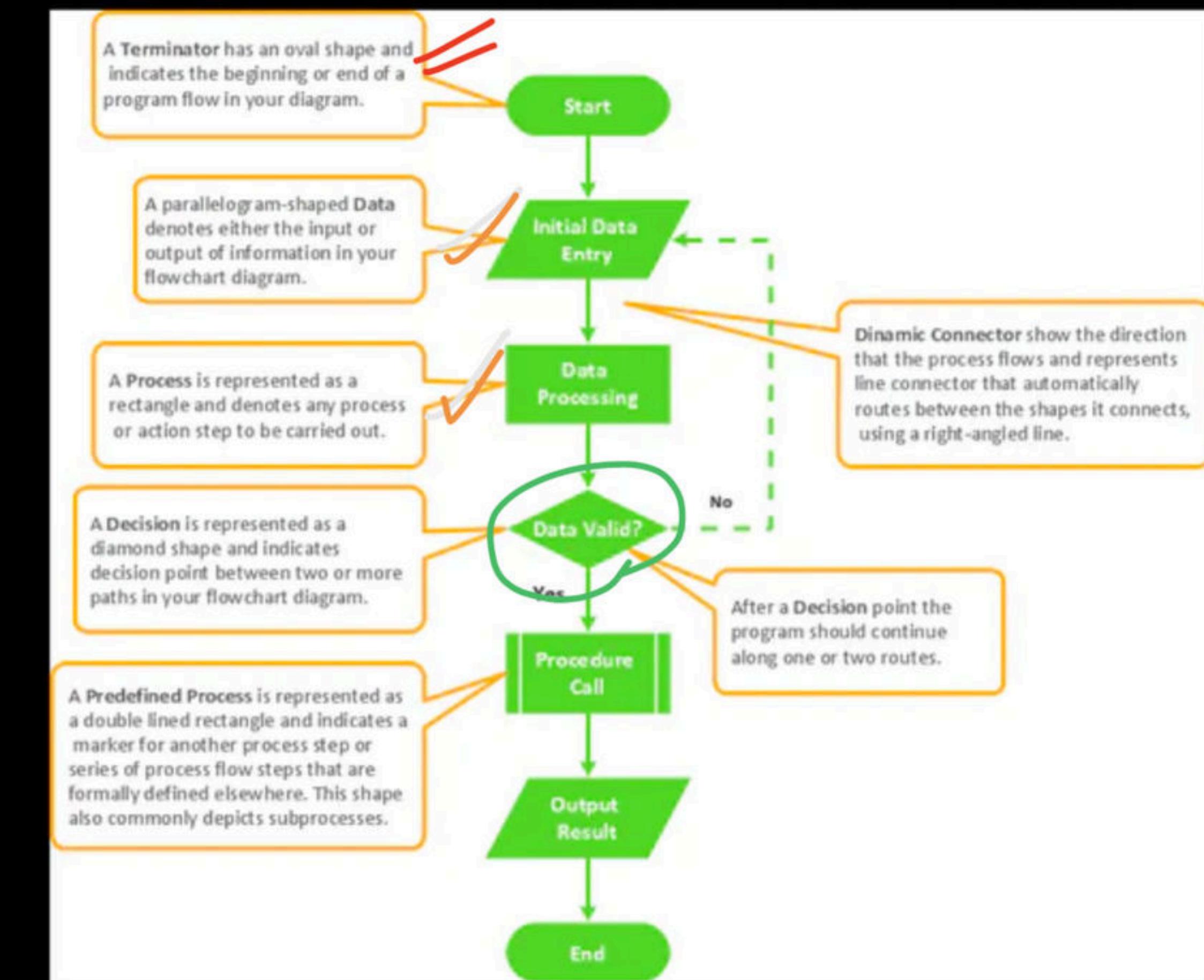
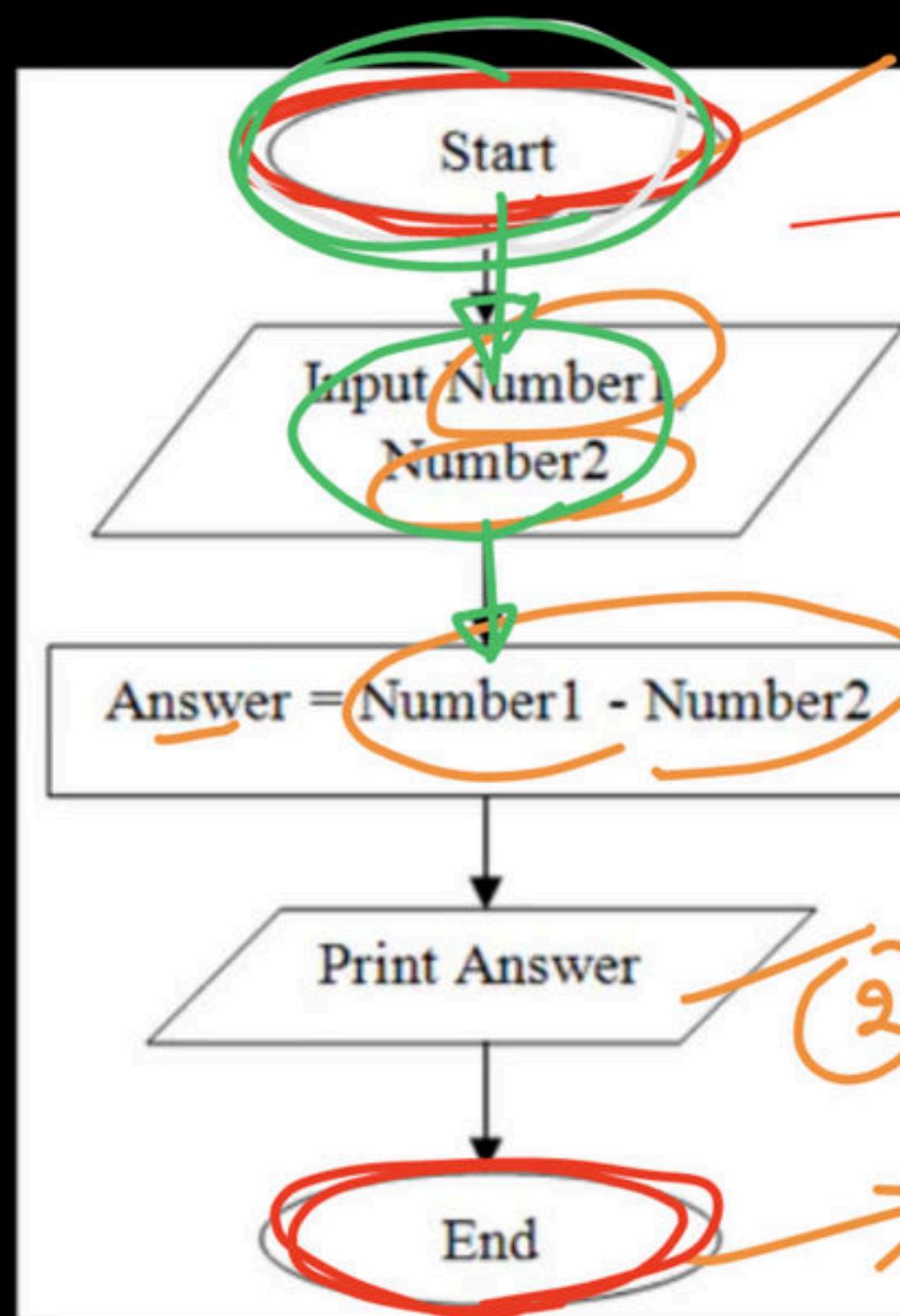
HLL

prime  $\rightarrow$  L  
 $\rightarrow$  prime not  
Yrem = 0



# FlowCharts:

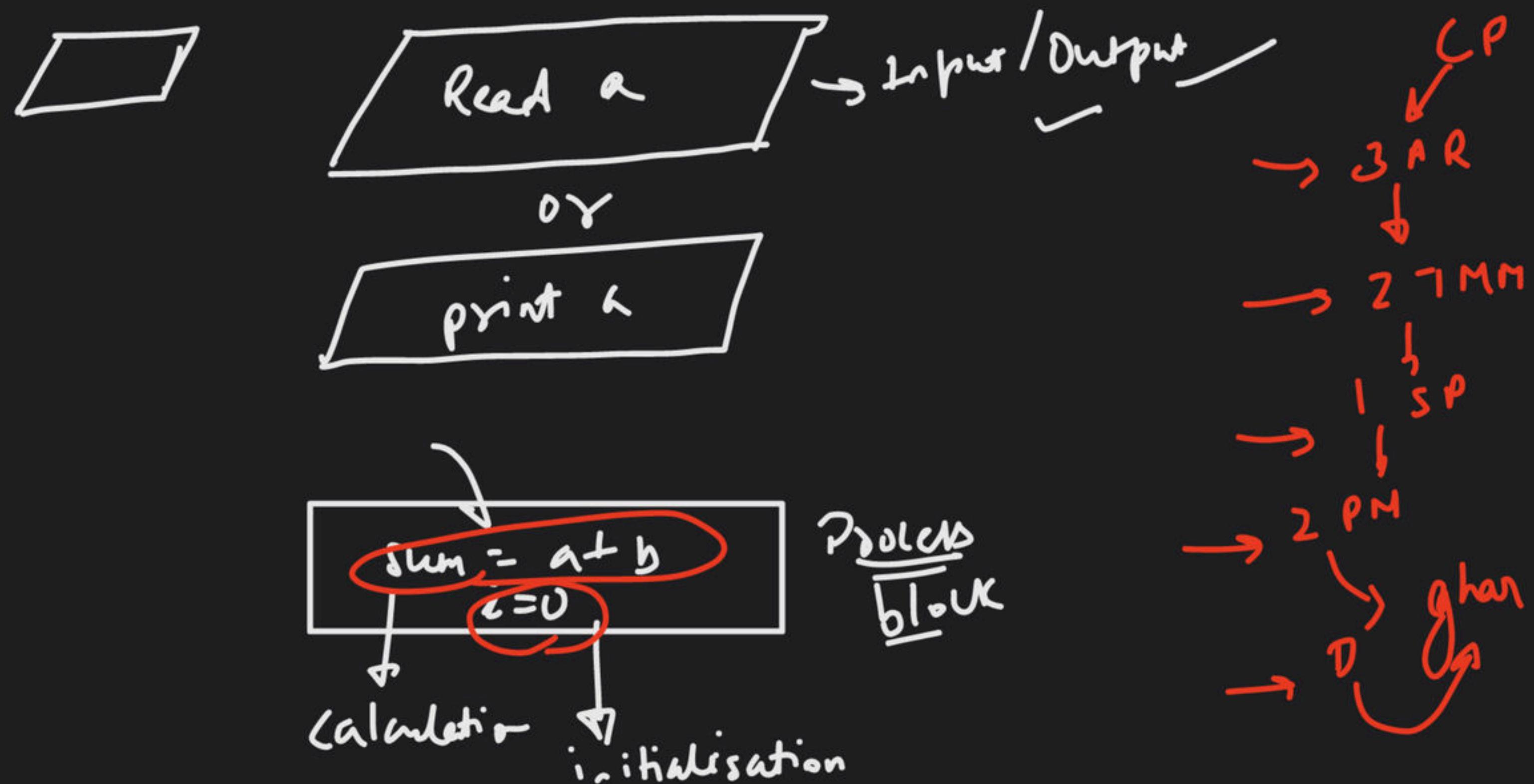
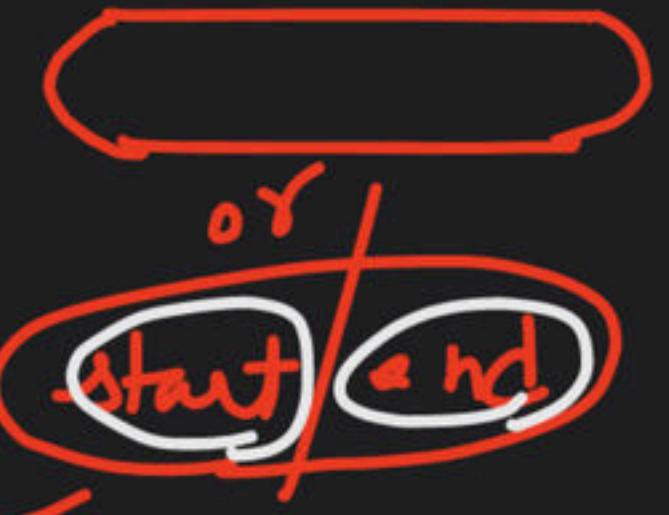
Flowchart is a graphical representation of an algorithm. Programmers often use it as a program-planning tool to solve a problem. It makes use of symbols which are connected among them to indicate the flow of information and processing. The process of drawing a flowchart for an algorithm is known as "flowcharting".

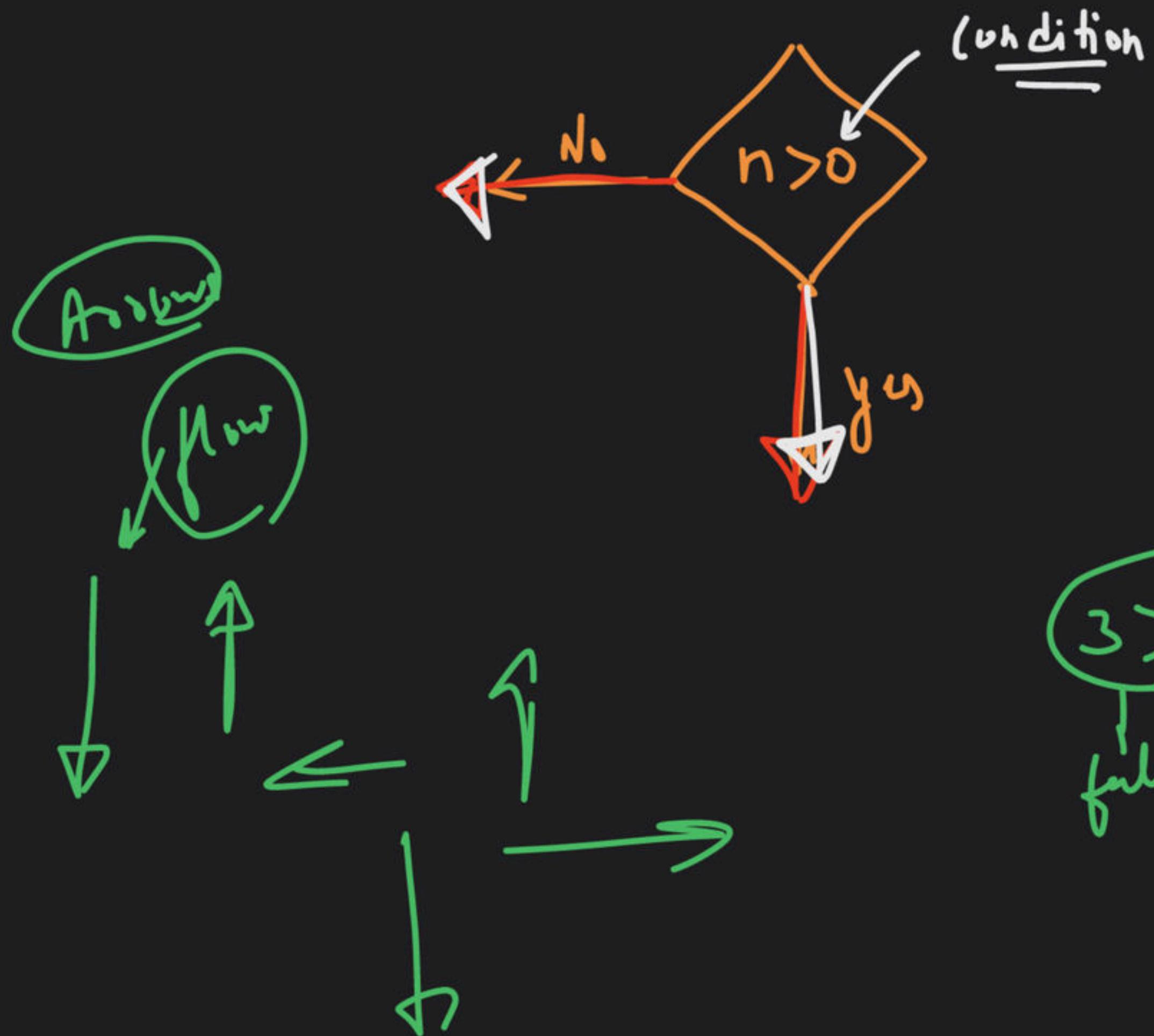


number 1 = 5  
number 2 = 3  
 $Answer = 5 - 3 = 2$

algorithm

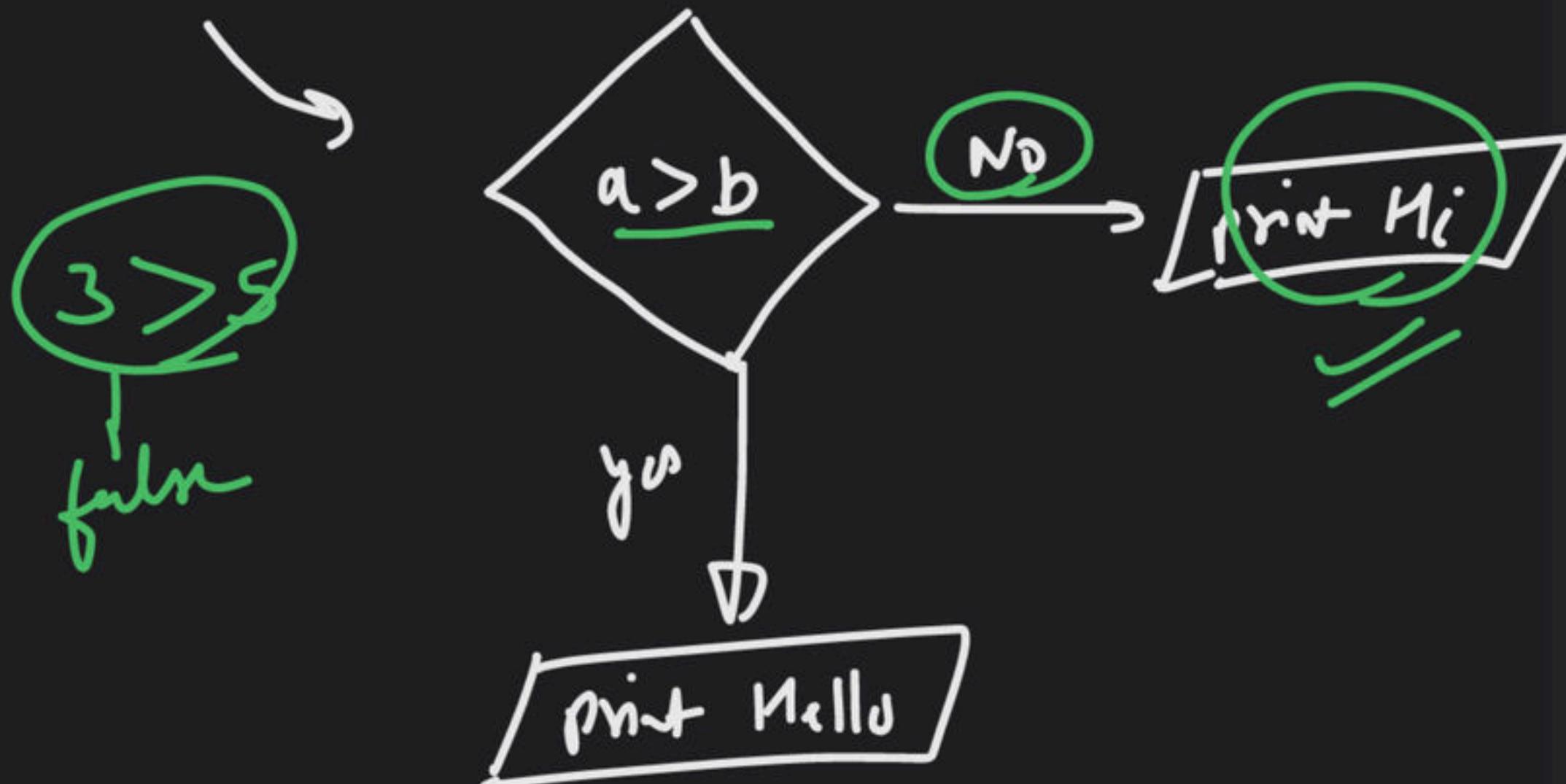
# Components

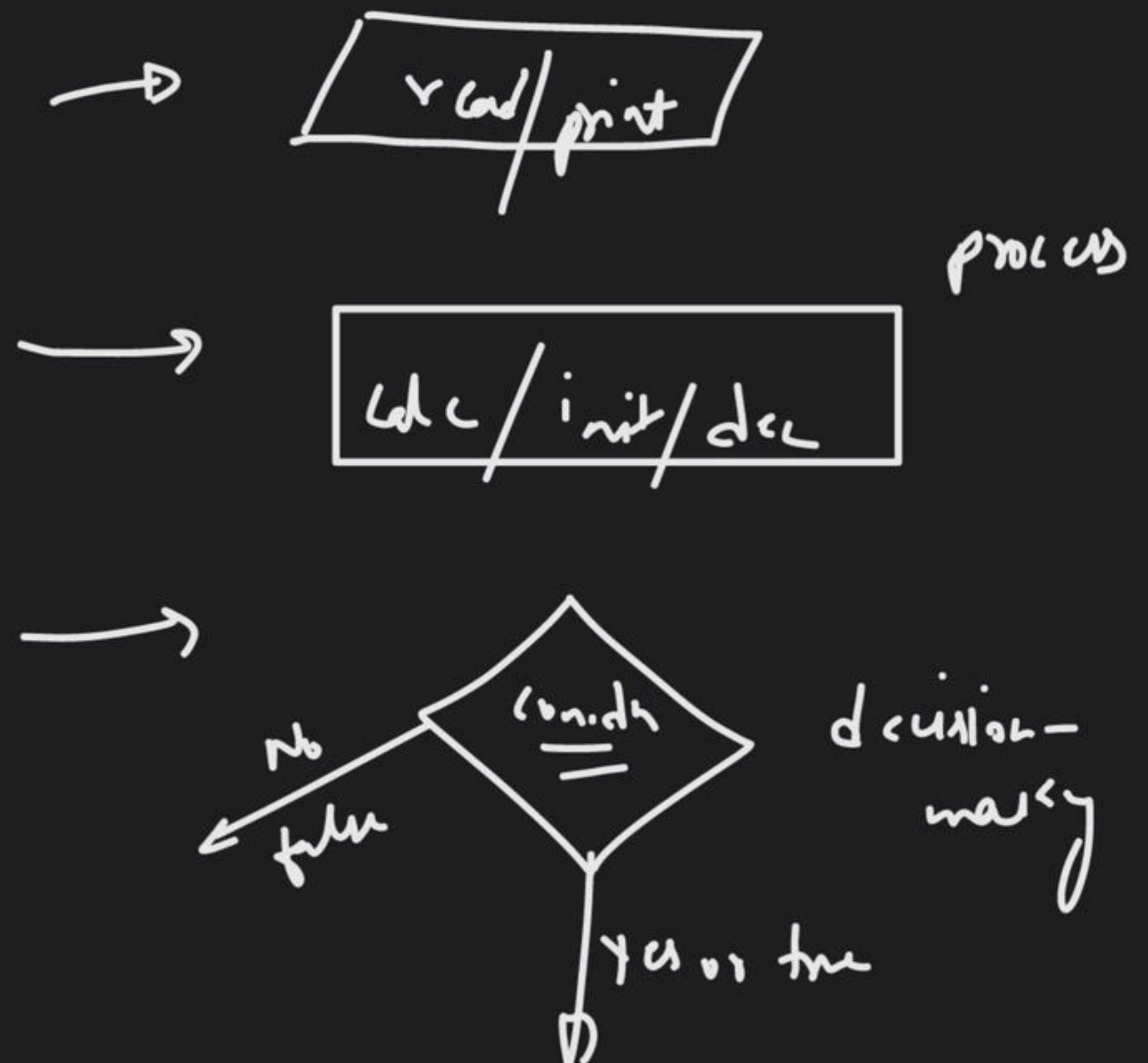




Decision-making  
Block

$$\begin{aligned} a &= 3 \\ b &= 5 \end{aligned}$$





Connector :-> (skip)

A

function?

start / end  
terminated

arrows

# FlowChart Components:

# PseudoCode:

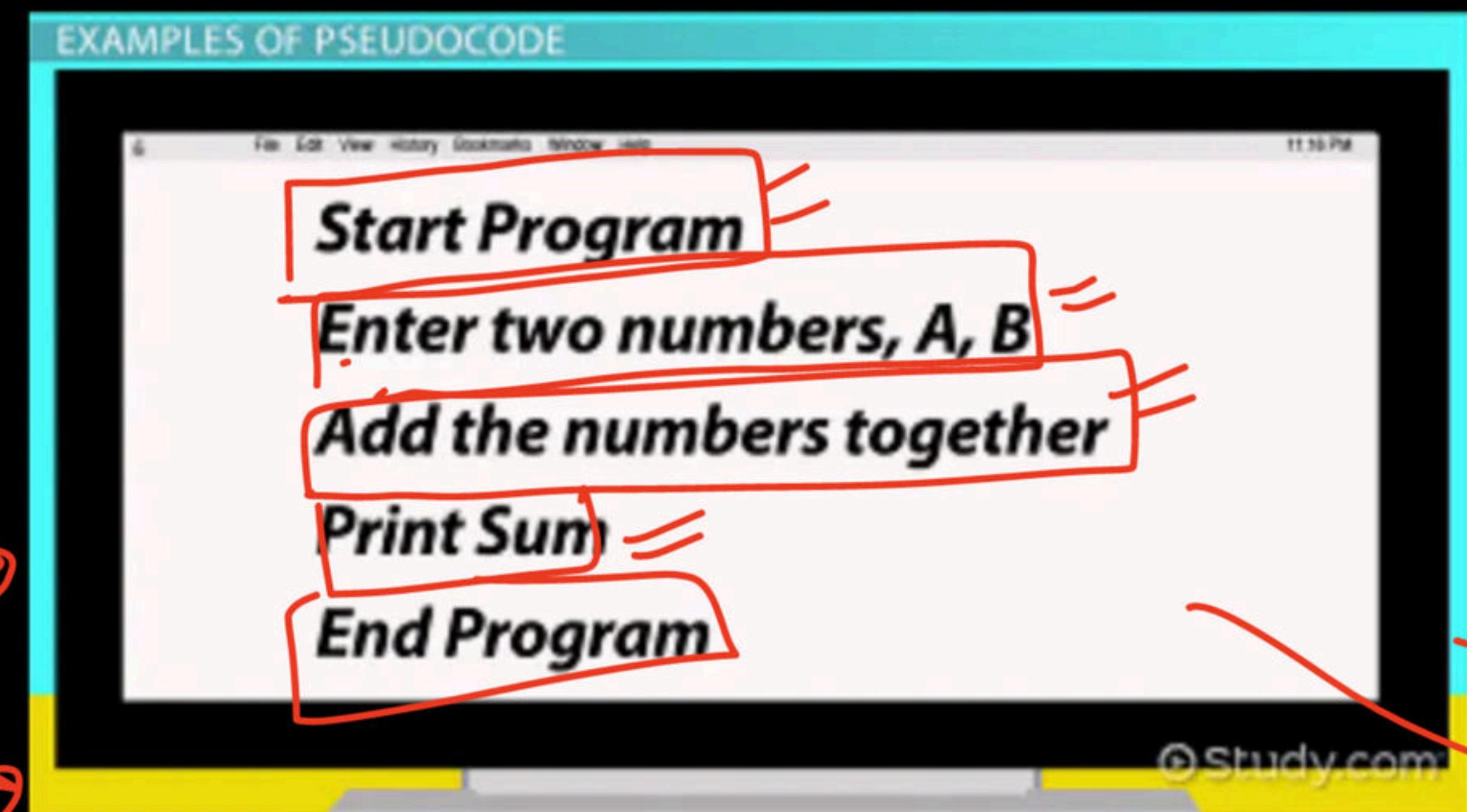
Nakli code

Pseudocode is a detailed yet readable description of what a computer program or algorithm must do, expressed in a formally-styled natural language rather than in a programming language. Pseudocode is sometimes used as a detailed step in the process of developing a program.

generic way  
of representing  
your

algo in textual  
form

informal  
English



I answer

give me difference of  
2 no's a & b

print

a - b

→ Solution :-

- ① Read a & b
  - ② difference =  $a - b$
  - ③ print difference
- Read c
- Read b
- print( $a - b$ )

Point the multiplication/

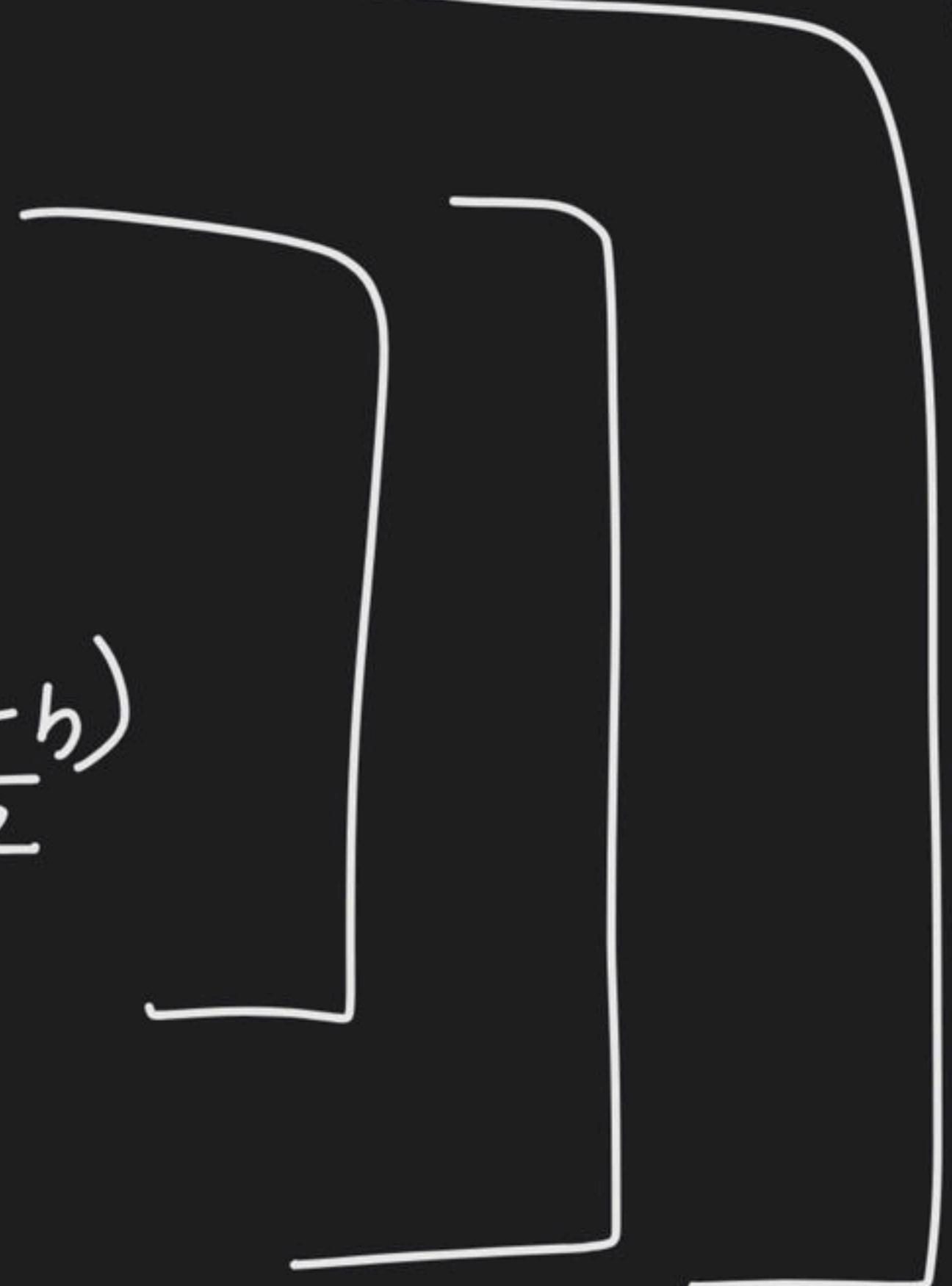
product  
of  $a \& b$

- ① Read a
  - ② Read b
  - ③ product =  $a \times b$
  - ④ print product
- 

Print the avg of 2 no

adb

- ① Start
- ② Read a
- ③ Read b
- ④  $\text{avg} = \frac{(a+b)}{2}$
- ⑤ Print avg
- ⑥ Exit



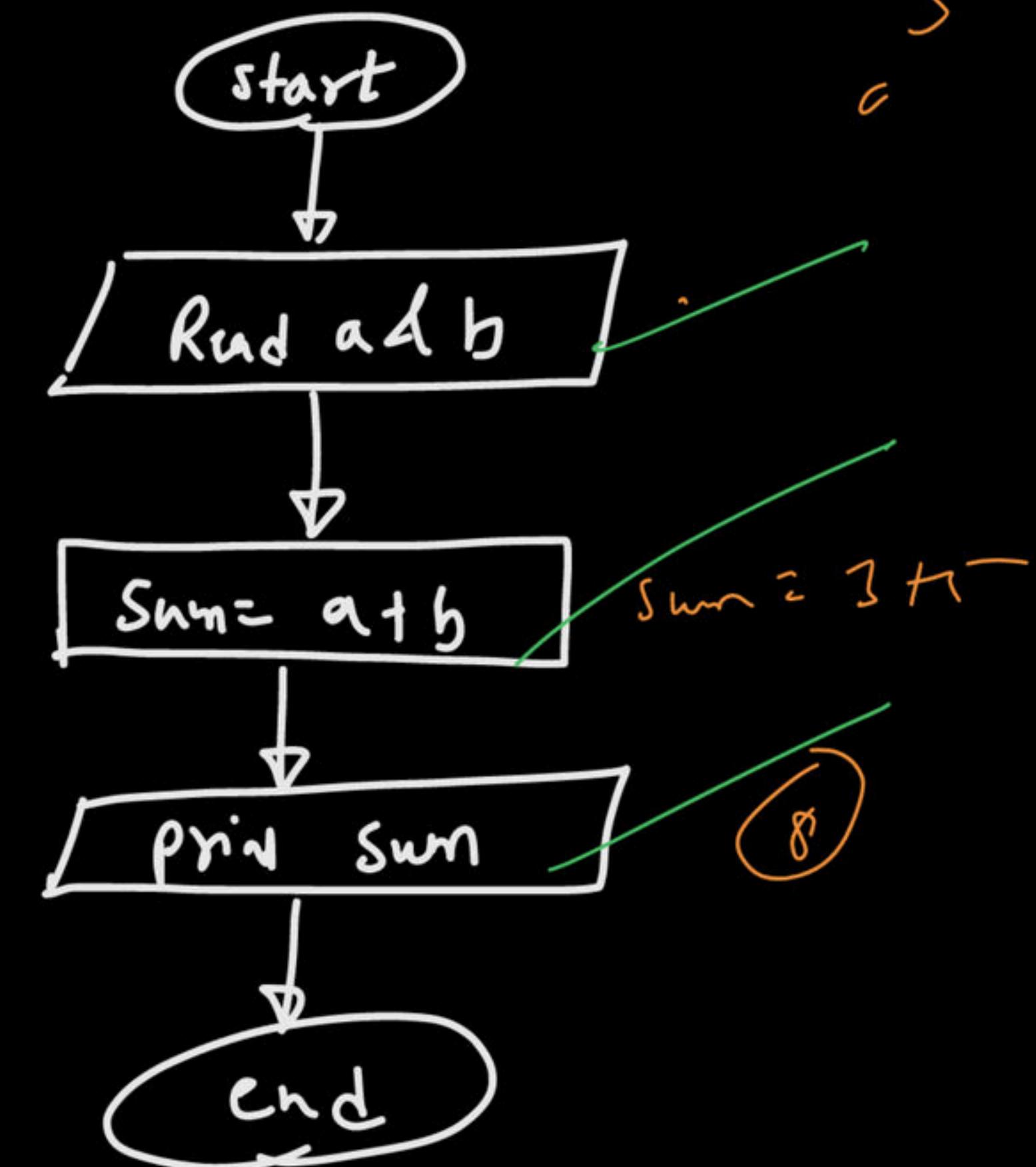
**Time to Practice starts :-)**

# Add 2 numbers by taking Input

① flowchart

Pseudocode -

- ① Read a & b
- ② sum = a + b
- ③ print sum



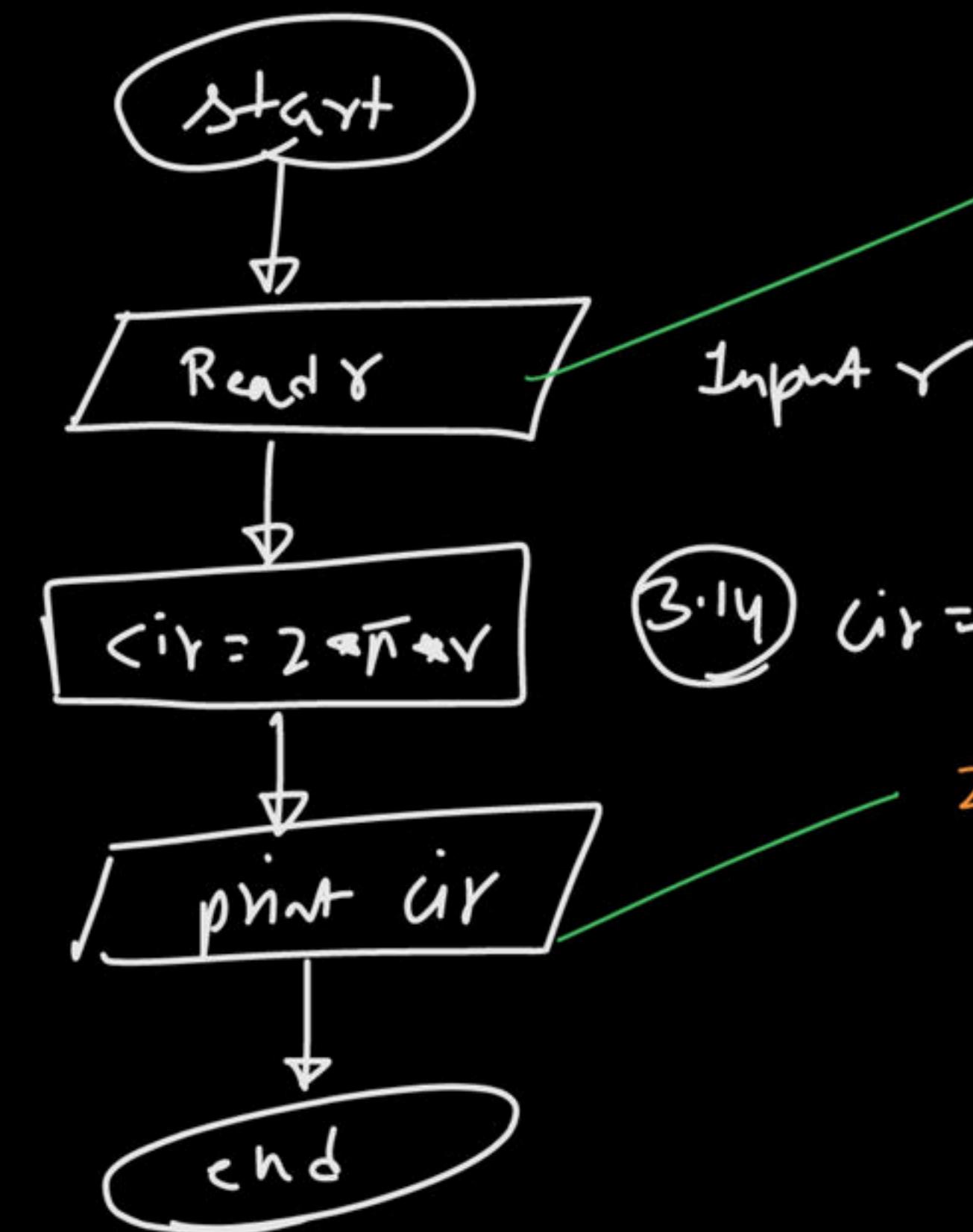
$$\begin{array}{c} a \\ \downarrow \\ 3 \\ + \\ b \\ \downarrow \\ 5 \\ \hline c \end{array}$$

# Find Circumference of a Circle

$$\underline{\underline{C = 2\pi r}}$$

$$\text{Circumference} = 2\pi r$$

F.C.



$$r = 2$$

Input r

③  $3.14 \quad \text{Cir} = 2 \times 3.14 \times r$

$$\begin{aligned} & 2 \times 3.14 \times 2 \\ & = 6.28 \times 2 \\ & = 12.56 \end{aligned}$$

④ 12

Pseudocode

① Start  
② Read r

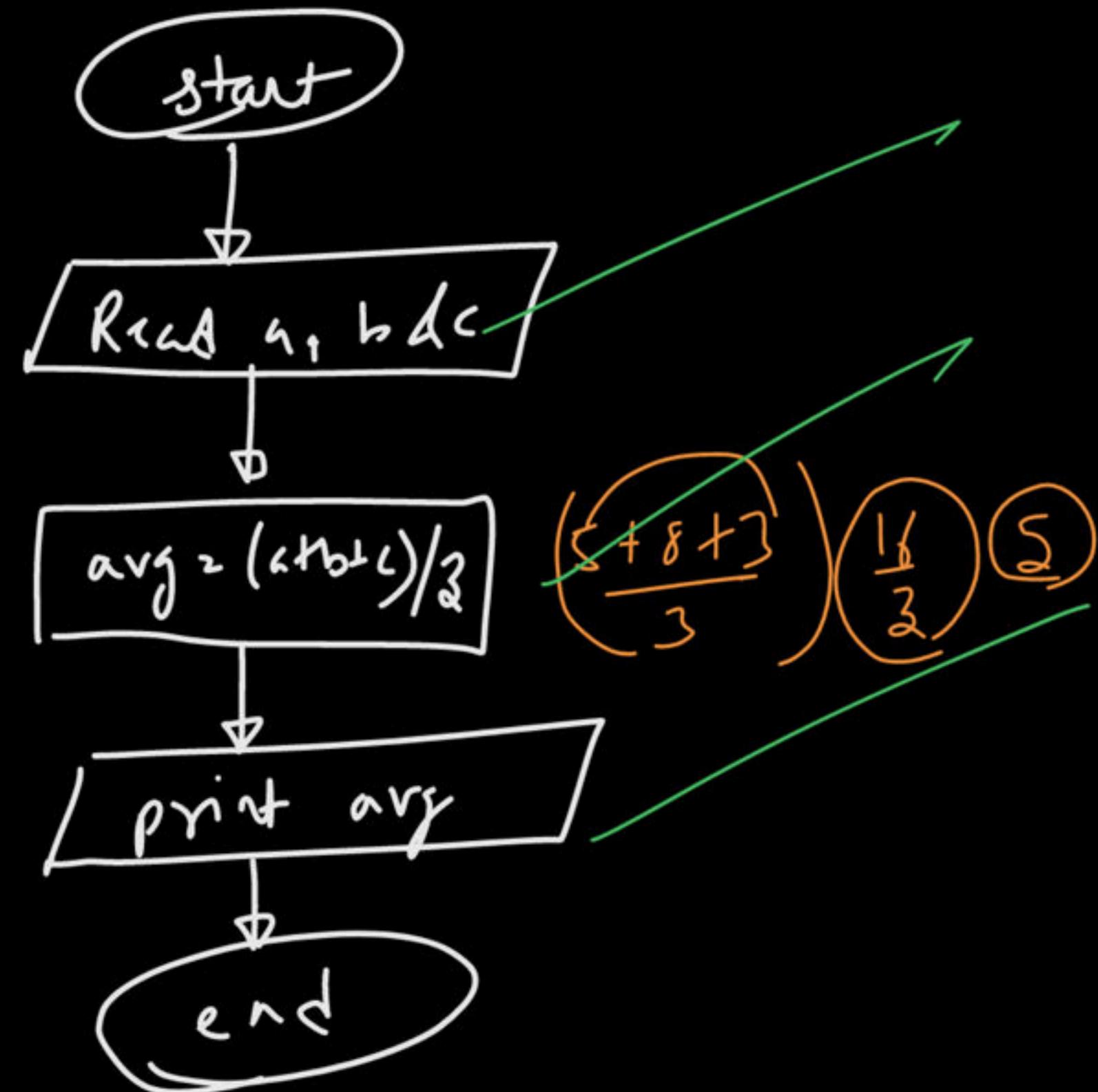
③ Cir = 2 \* 3.14 \* r

④ print Cir

Exit

# Average of 3 numbers

F = c



$$\begin{array}{c} a \quad b \quad c \\ \hline 5 \quad 3 \quad 8 \end{array}$$

$$\text{avg} = \frac{a+b+c}{3}$$

10.5cc

Pseudocode

start  
→ Read a, b, c  
→ avg = (a+b+c)/3  
→ Print avg  
exit

# Check number is **ODD** or **EVEN**

$$n \% 2 = 0$$

$$\frac{n}{2} \rightarrow \text{sum} \rightarrow 0$$

$\frac{n}{2} \rightarrow \text{sum}$   
 $n \% 2 = 0$   
2 के ताले में आते हैं

2 के ताले में आते हैं

$$\frac{n}{2} \rightarrow \text{remainder} \rightarrow 0$$

$$\frac{n}{2} \rightarrow \text{sum} - 1$$

$$a \% b$$

$$a \% b = 3$$

$$\frac{a}{b} \rightarrow \text{sum} = 3$$

$$\frac{14}{2} \rightarrow \text{sum} = 0$$
  
$$\frac{16}{2} \rightarrow \text{sum} = 4$$
  
$$\frac{18}{2} \rightarrow \text{sum} = 9$$
  
$$\frac{20}{2} \rightarrow \text{sum} = 12$$

$$\frac{n}{y} = 5$$

Condition

$$\sqrt{\frac{n}{y}} \rightarrow \text{rem} \rightarrow 5$$

$$n \cdot 2 \rightarrow 0 \leftarrow \text{even}$$

$$n \cdot 2 \rightarrow 1 \leftarrow \text{odd}$$

$$n \cdot 2 = 0 \rightarrow \text{even}$$

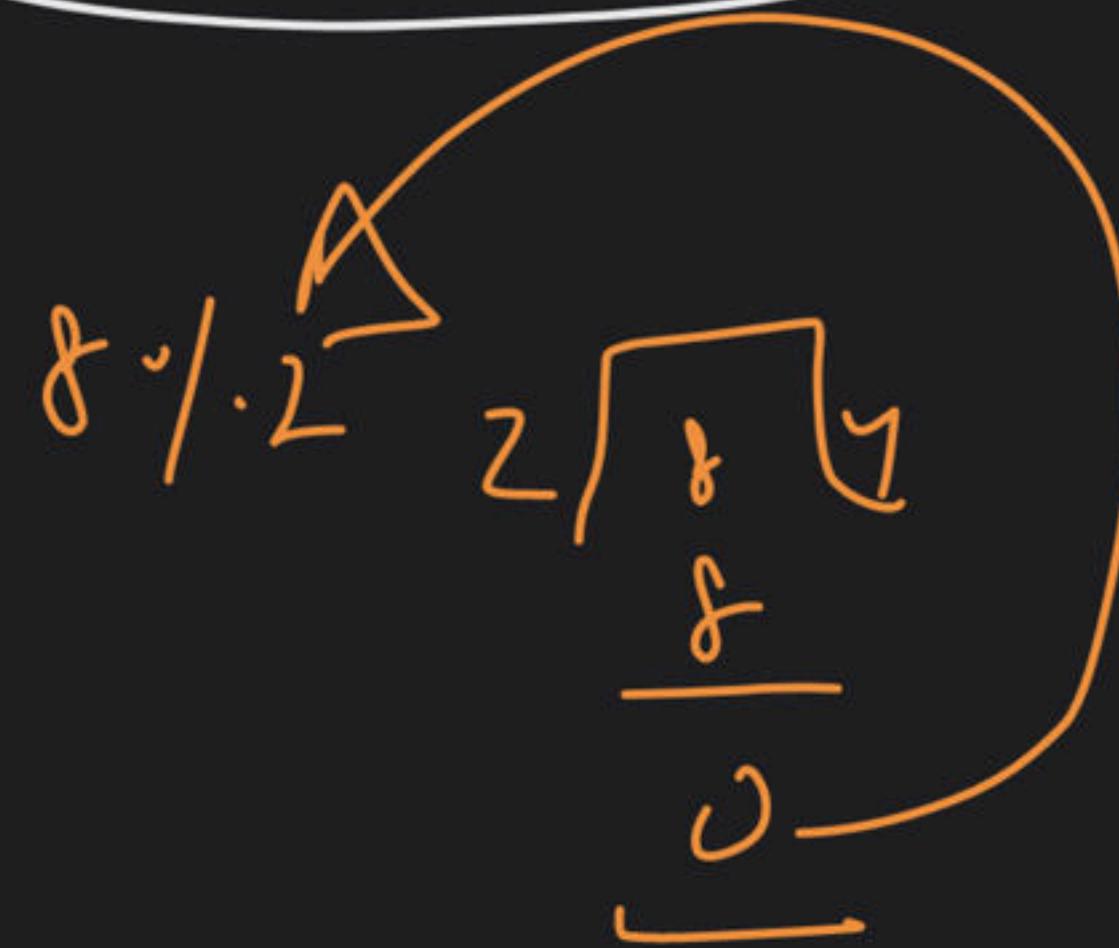
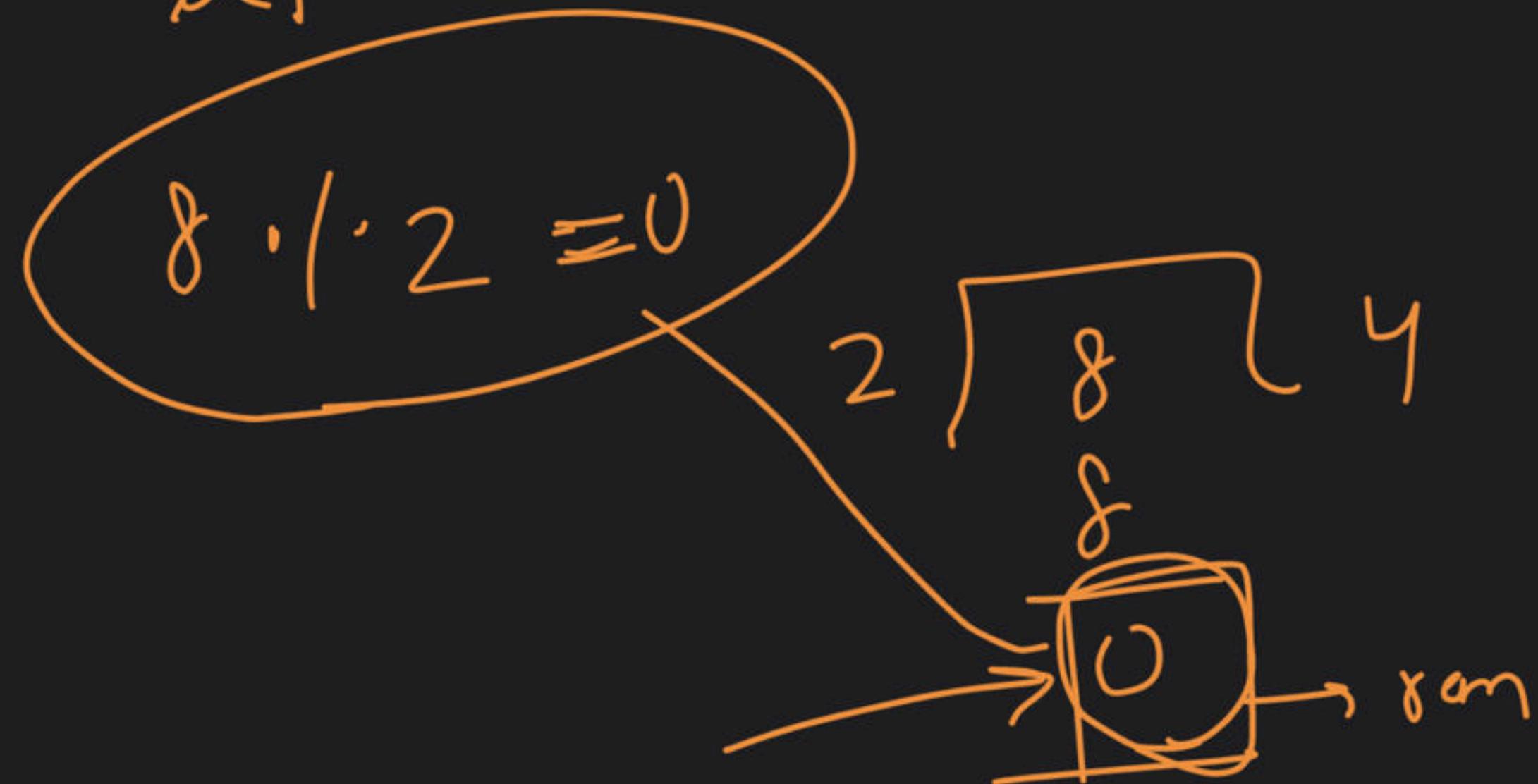
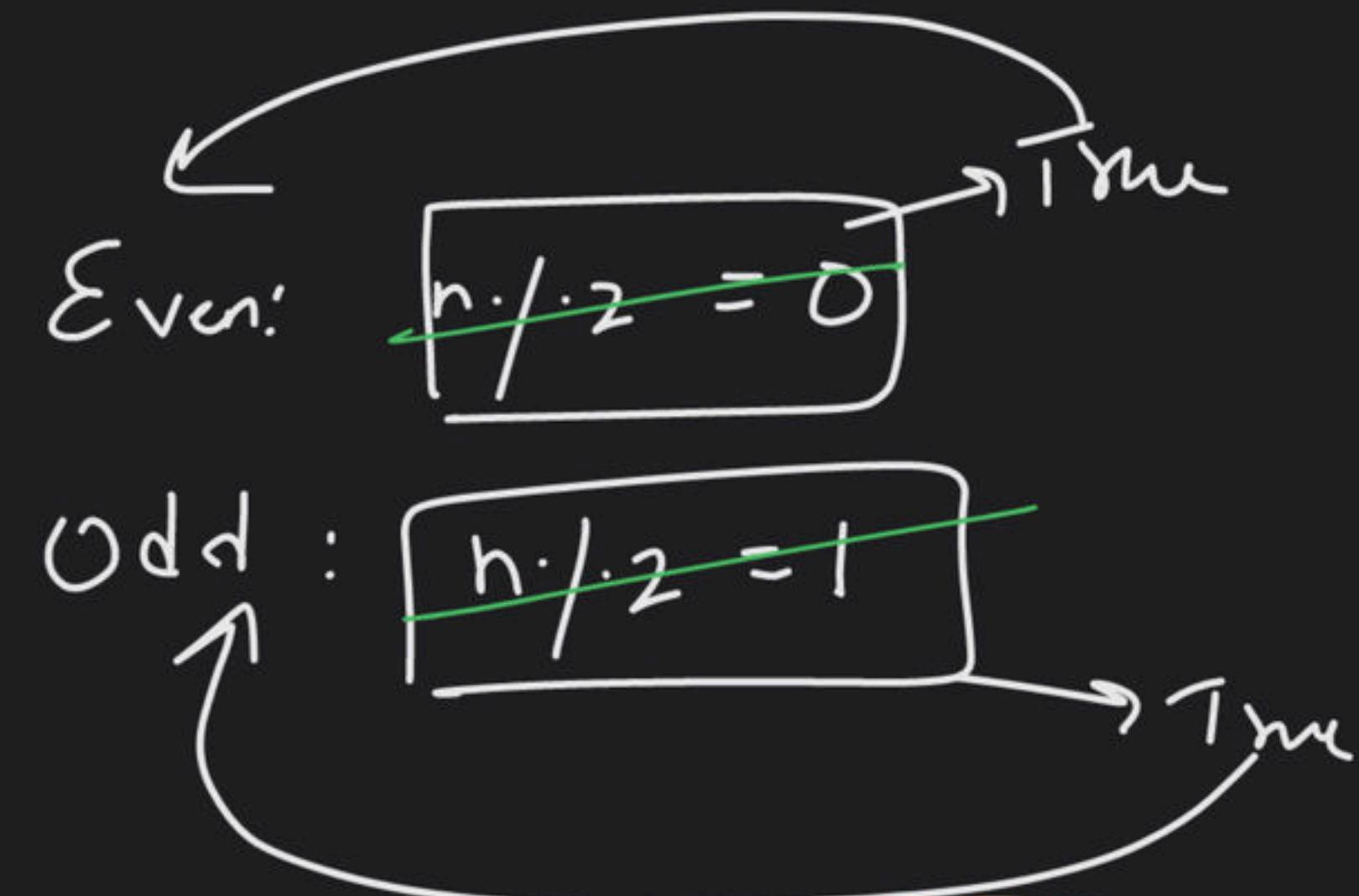
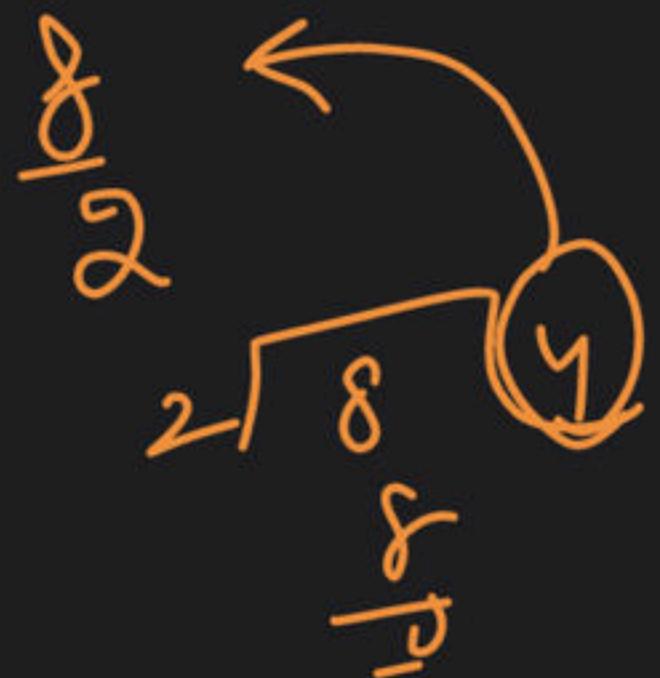
$$' = n \cdot 2 = ' \rightarrow \text{odd}$$

Sounds of

Reload /

Reposi  
t

$$\frac{8}{2} = 4$$



## flowchart

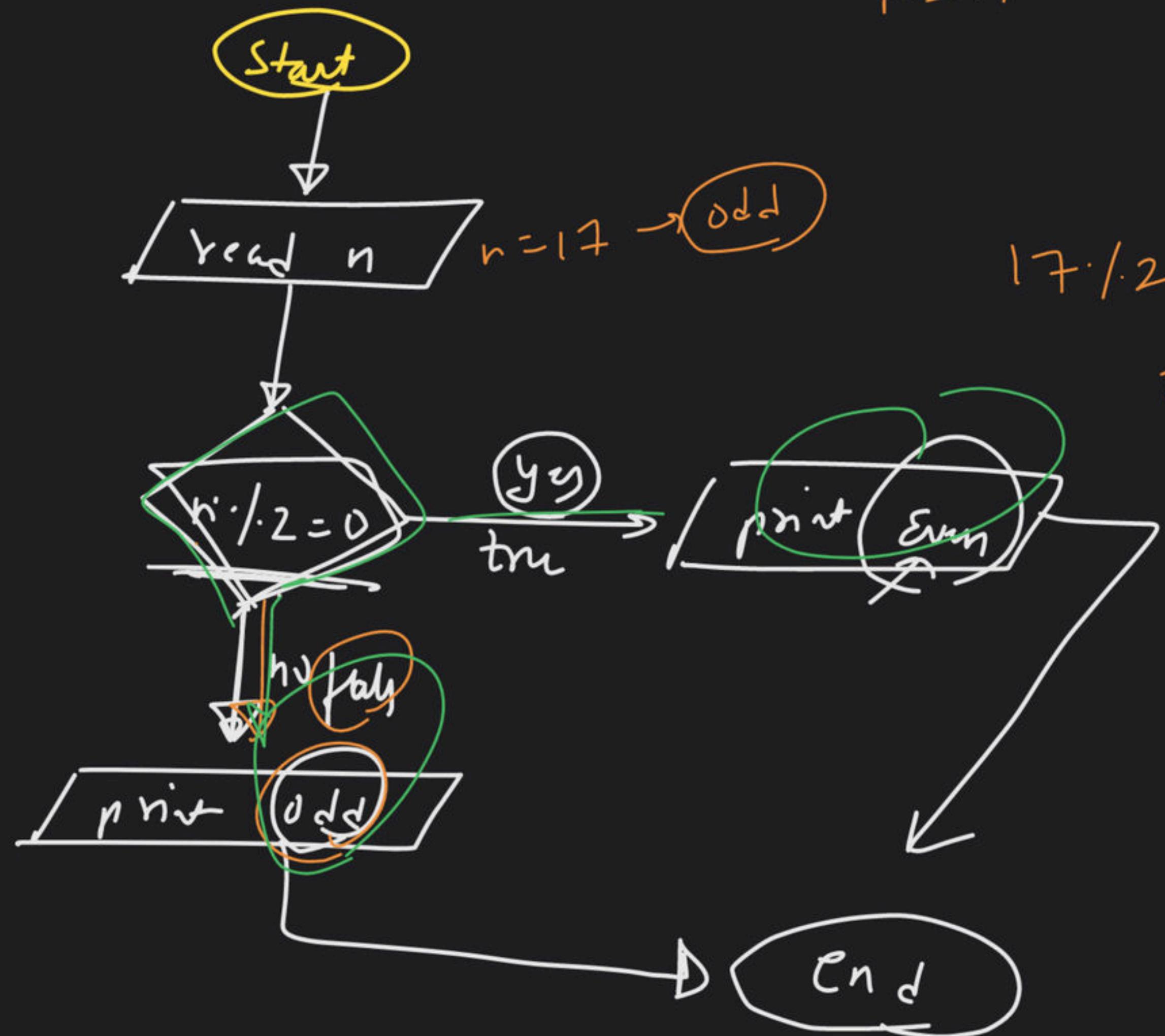
Read n

if ( $n \cdot 1.2 = 0$ )

then print even

else

print odd



$$n = 17$$

$$\frac{n}{1.2} = \underline{\quad}$$

$17 \cdot 1.2 = 0$

$$2 \sqrt{17} = 8$$

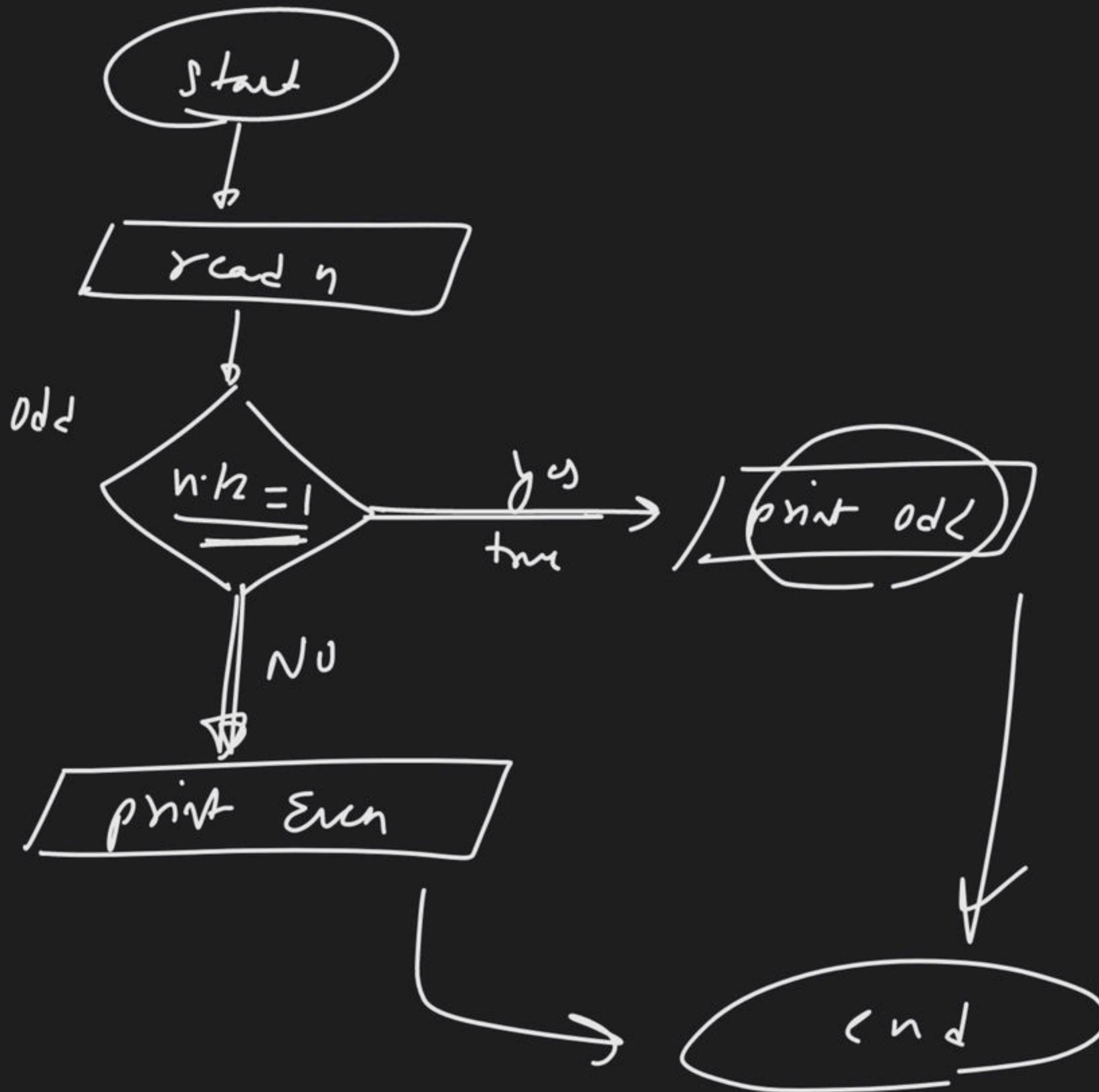
↓

1

Pseudocode:-

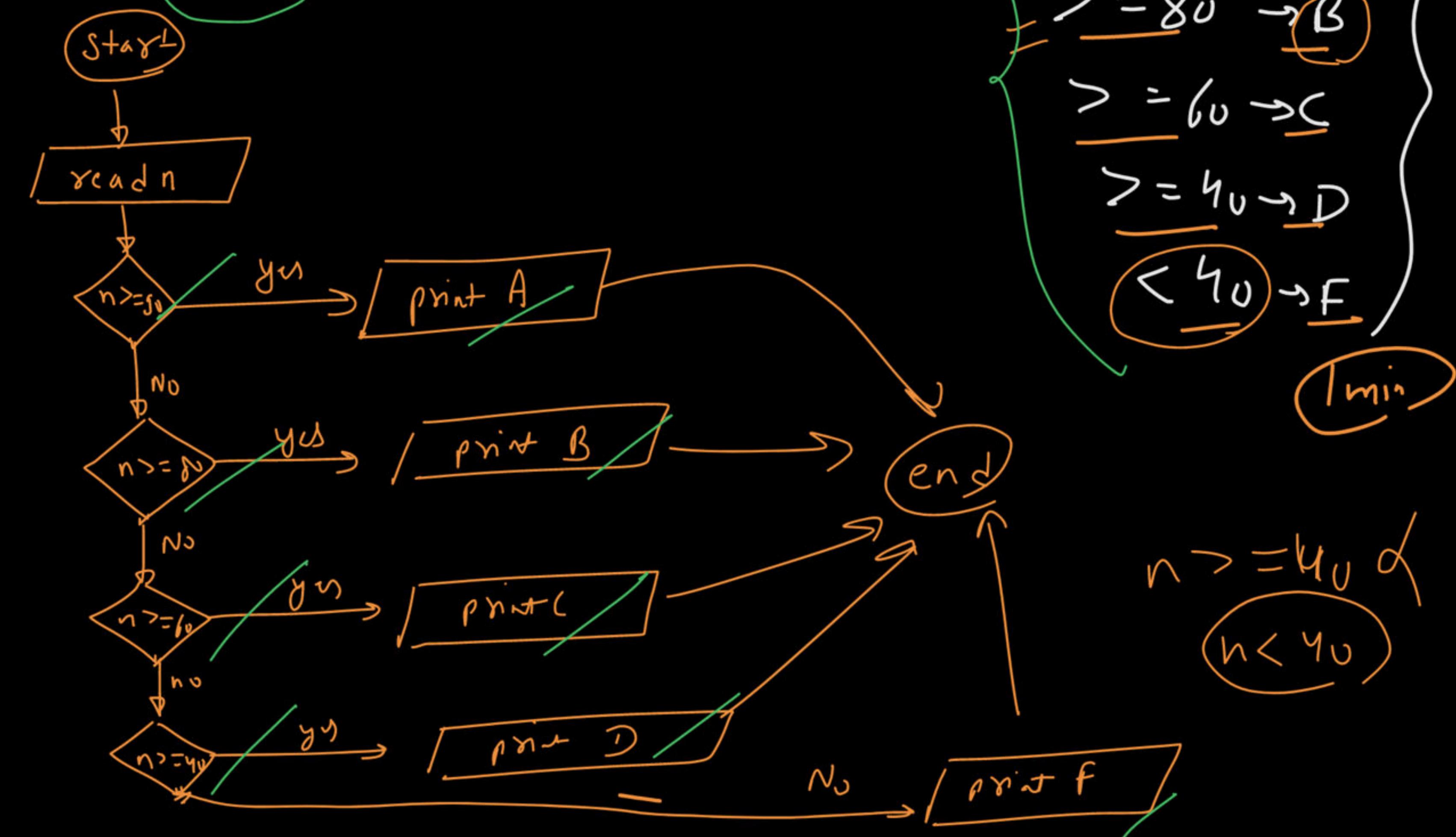
① Read n

② if  $n \cdot h = 1$  , then print odd  
else  
    print Even



# Students & Grade Flowchart

FC



Pseudocode:-

→ read m

→ if  $m \geq 90$   
    then print A

else if  $m \geq 80$   
    then print B

else if  $m \geq 60$   
    then print C

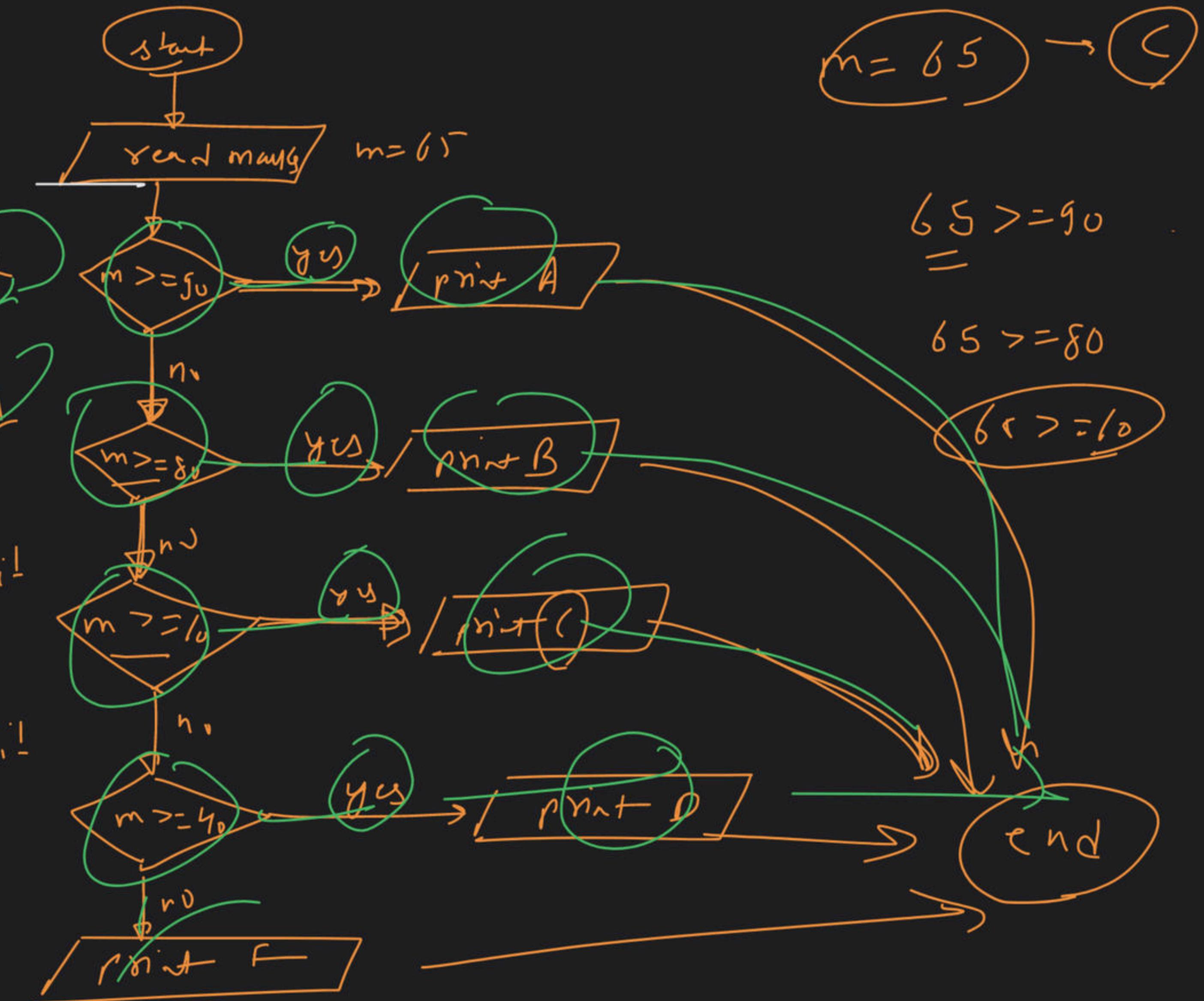
else if  $m \geq 40$

    print D

else  
    print F

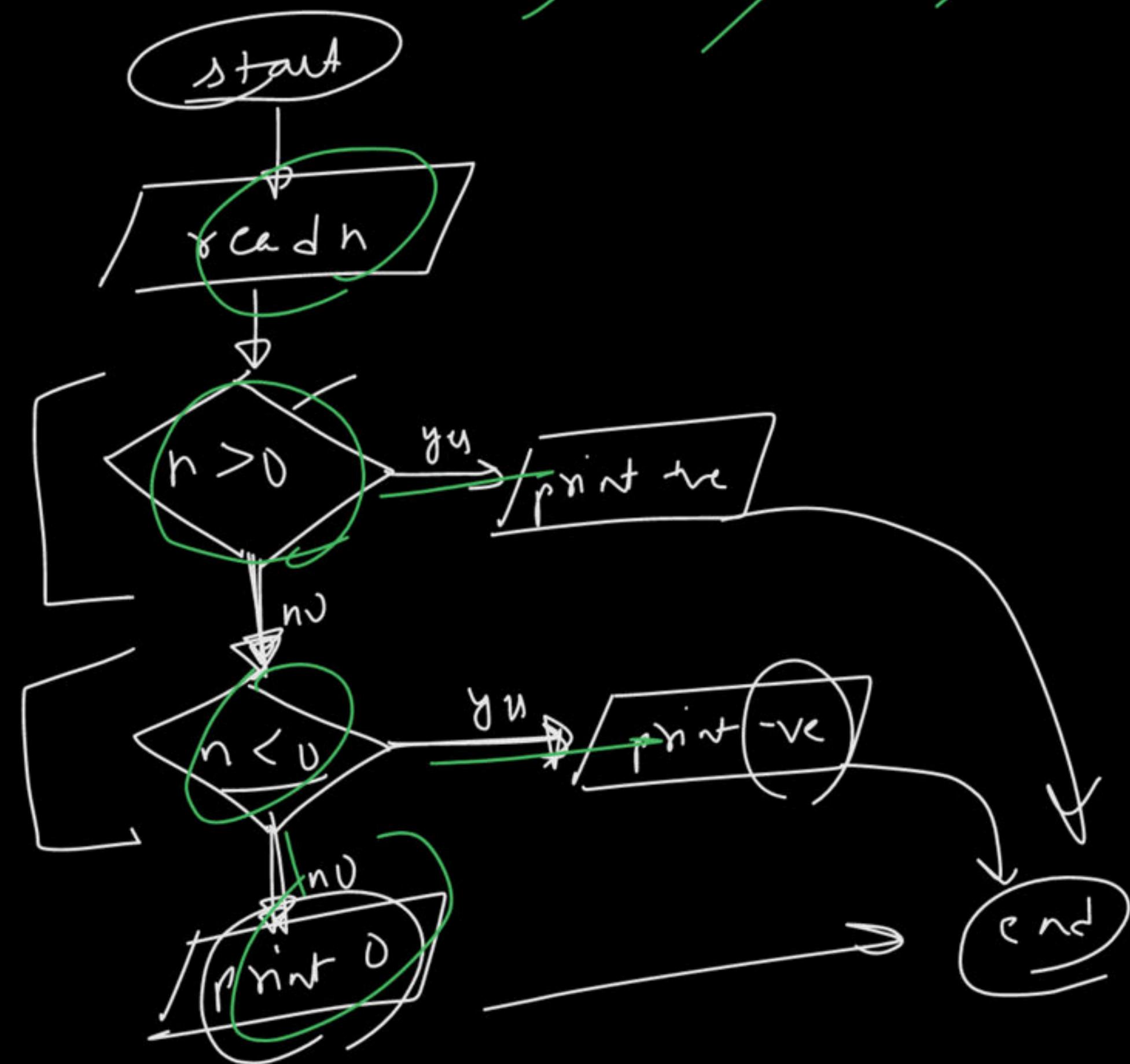
exit

GT



# Check number is +ve, -ve or 0.

flowchart



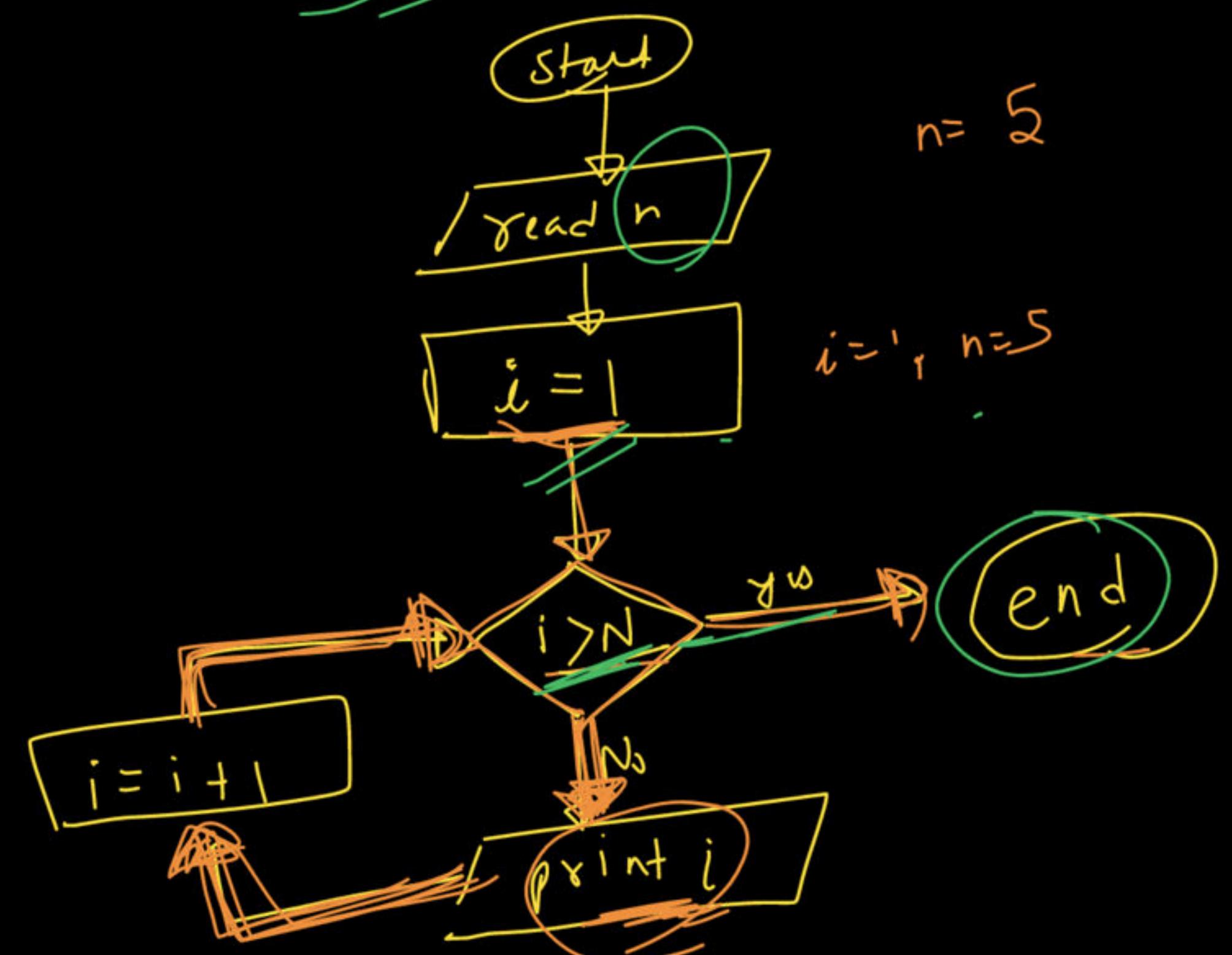
Pseudocode

```
→ Read n  
→ if  $n > 0$ , then print +ve  
else if  $n < 0$ , then print -ve  
else print 0
```

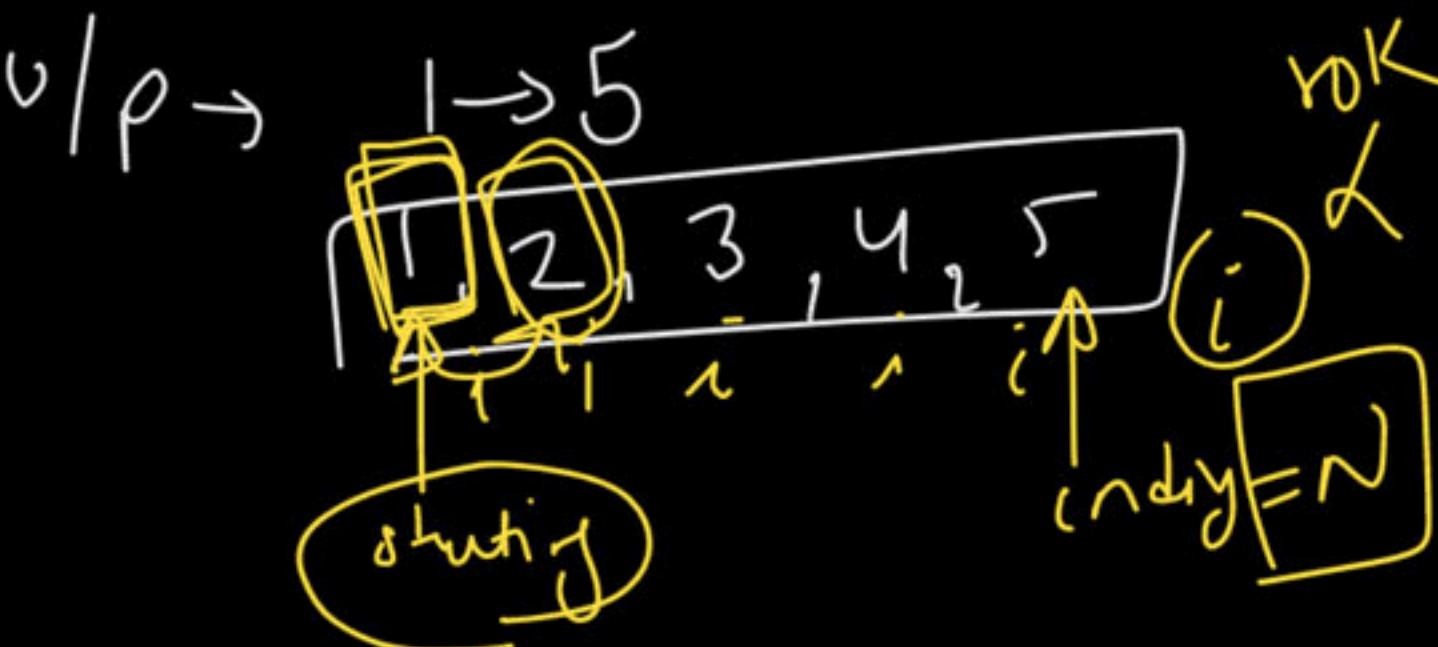
Bo ark

# Print Counting from 1 to N

E'



n = 5



i > 5 → no

print 1

i = 1 + 1 = 2

2 > 5 → no

print 2

i = 2 + 1 = 3

3 > 5 → no

print 3

i = 3 + 1 = 4

i > 5 → no

print 4

i = 4 + 1 = 5

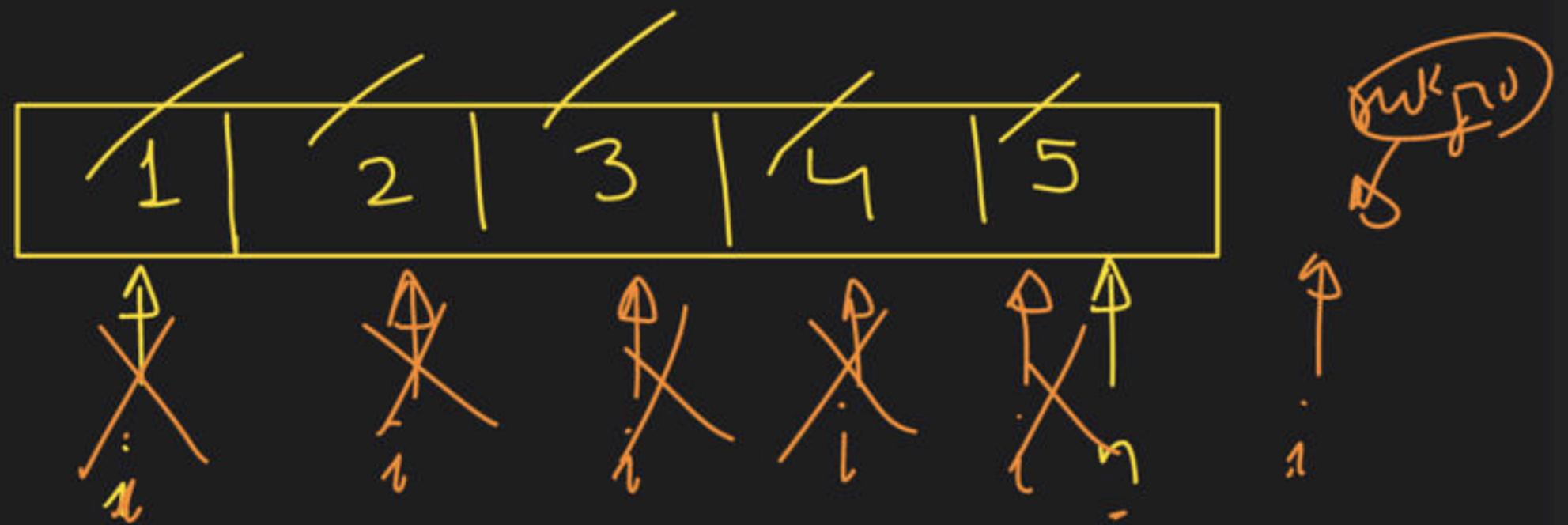
5 > 5 → no

print 5

i = 5 + 1 = 6

6 > 5 → yes

1 loop



$$\phi_p \rightarrow 1, 2, 3, 4, 5$$

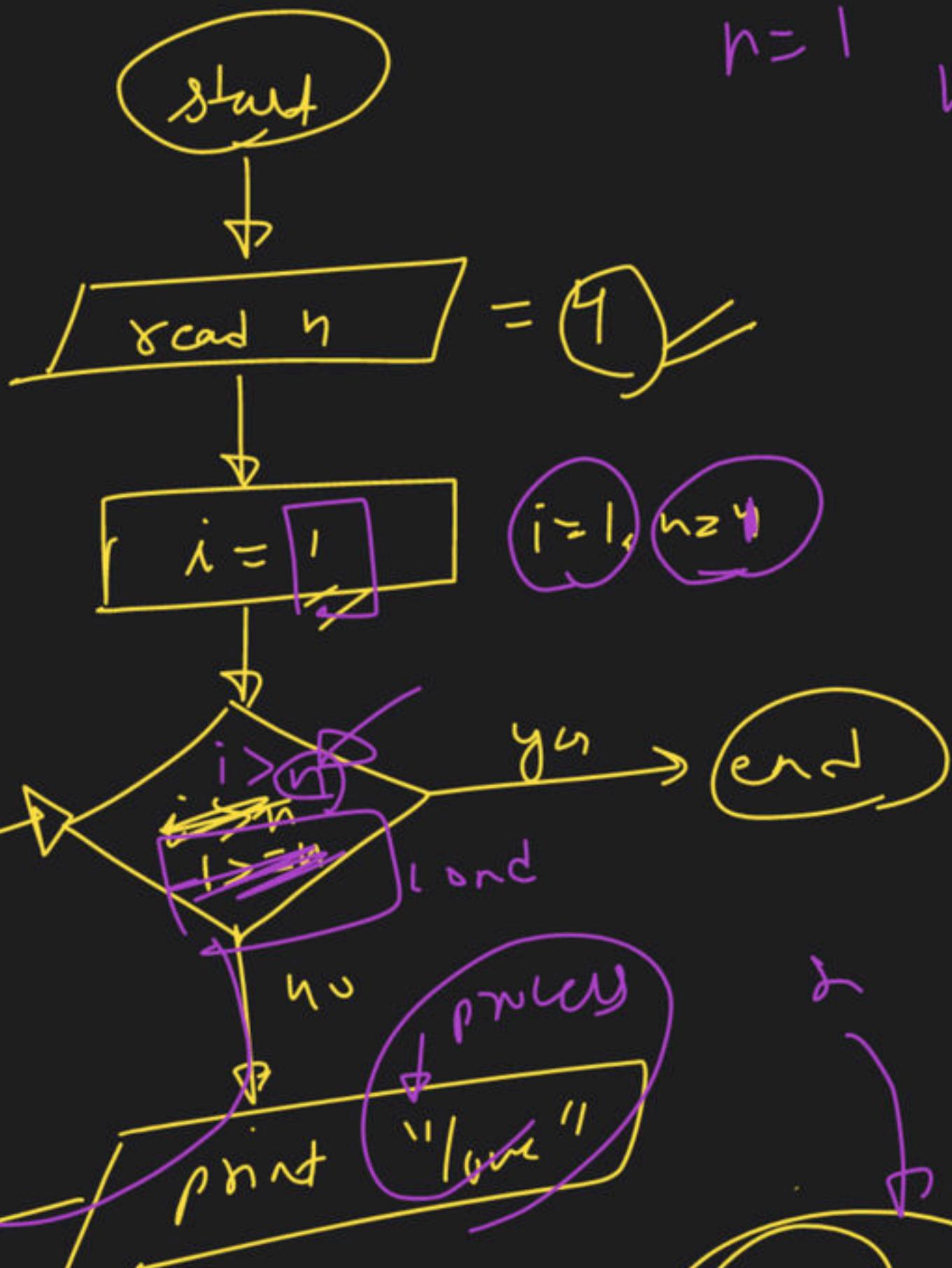
galoah ?

$i = 1$   
 $n = 1$

Updation  
increment

$i > n - 1$   
 $i >= n - 1$

$i = 0$   
 $i > z_n \rightarrow \text{no}$



$n = 1$   
 $i > 1 \rightarrow \text{no}$

$i = 1$   
 $i > r \rightarrow \text{end}$

$i = 0$

$i = 1$

$i = 0$

love  
 $i > n \rightarrow \text{no}$

$n \neq n$   
print

$i > n \rightarrow \text{no}$

$i = 1$

$i > n \rightarrow \text{no}$

$i = 2$

$i > n \rightarrow \text{no}$

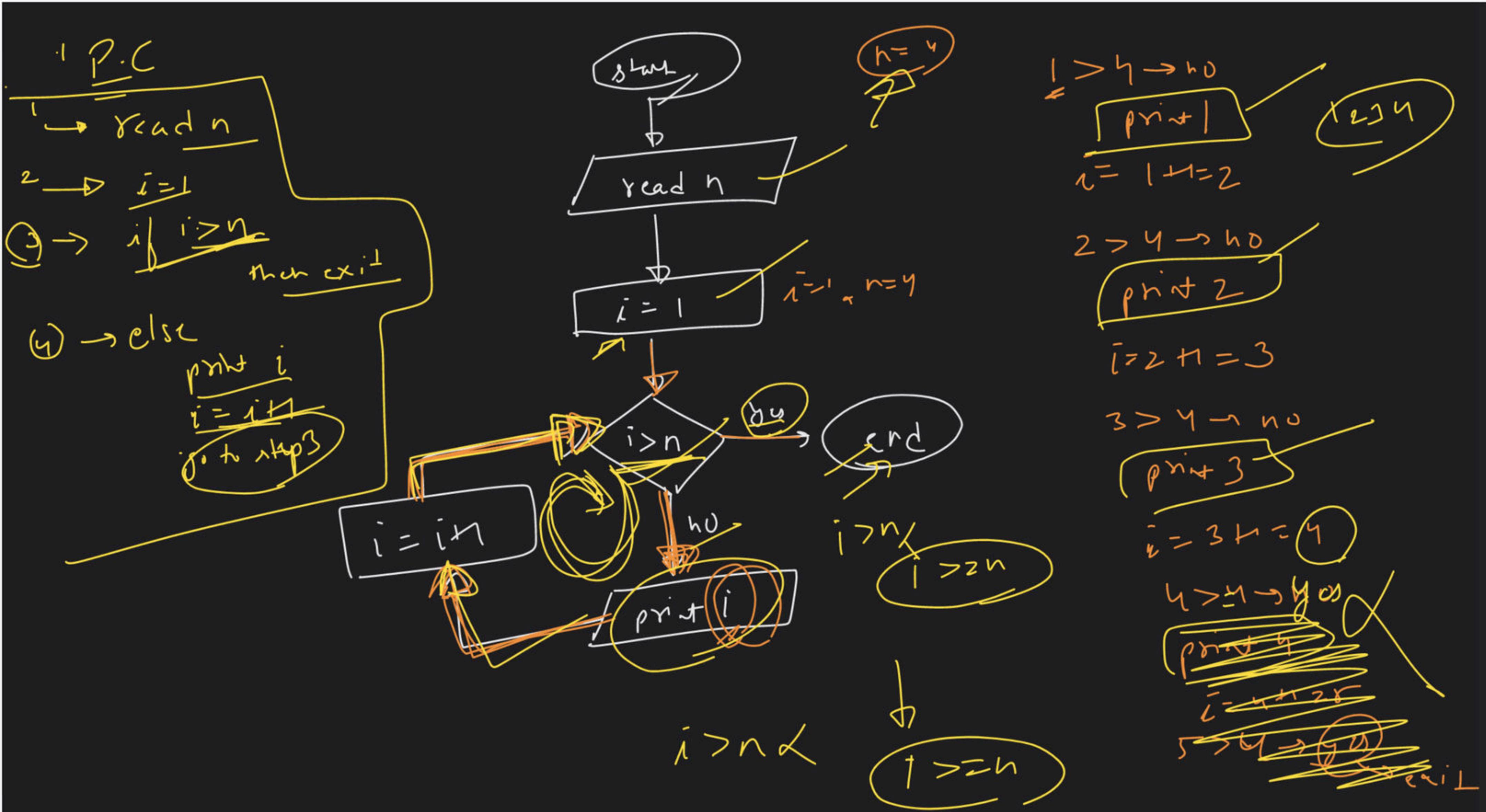
$i = 3$

$i > n \rightarrow \text{no}$

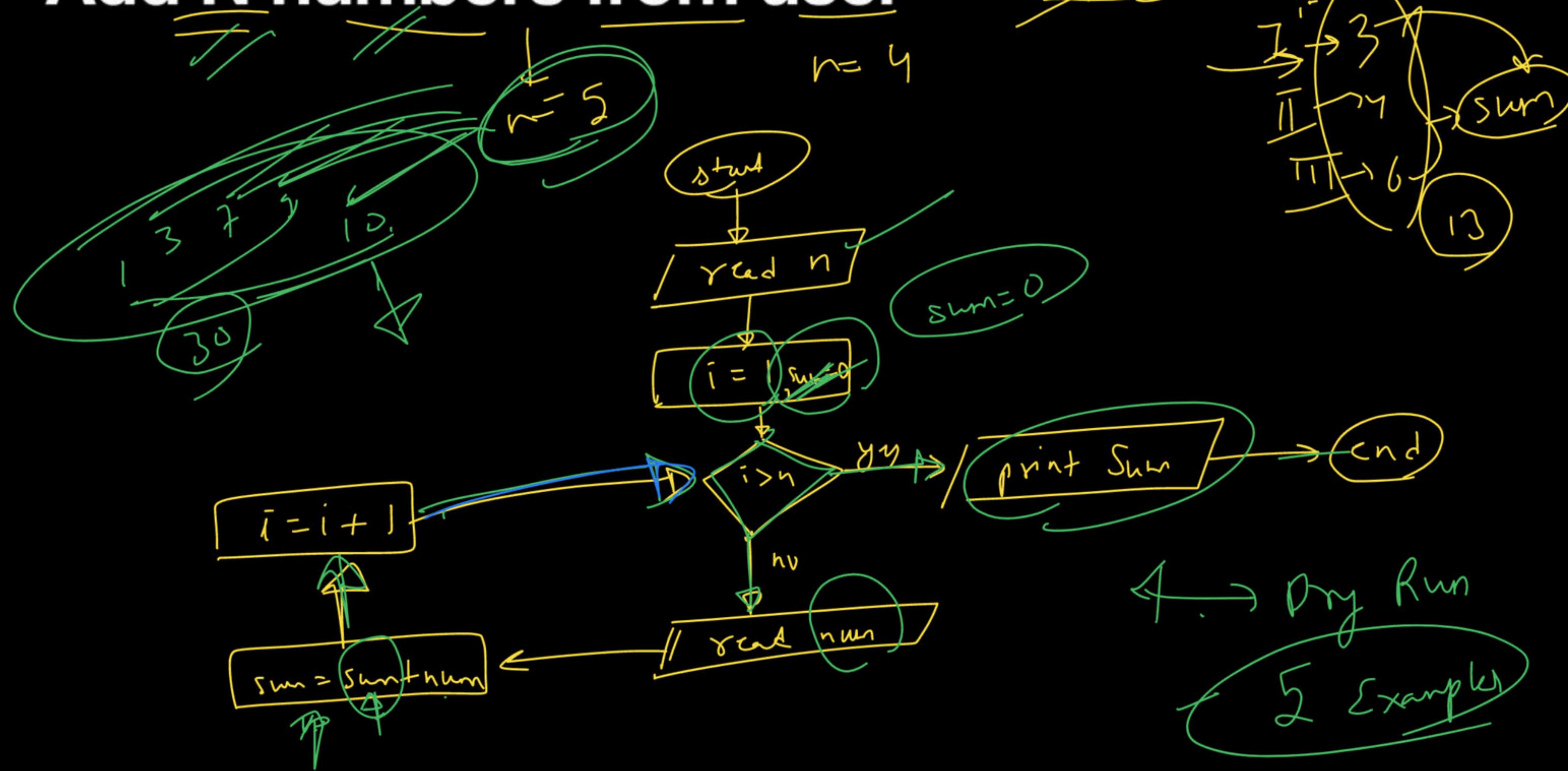
$i = 4$

$i > n \rightarrow \text{no}$

$i = 5$



# Add N numbers from user



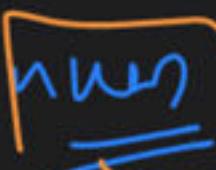
P.C

1 → read  how many numbers you are getting in input to create sum

2 →  $i = 1$

3 →  $sum = 0$

4 → if  $i > n$  then point sum ↗ exit

5 → else read  a number that should be included into sum

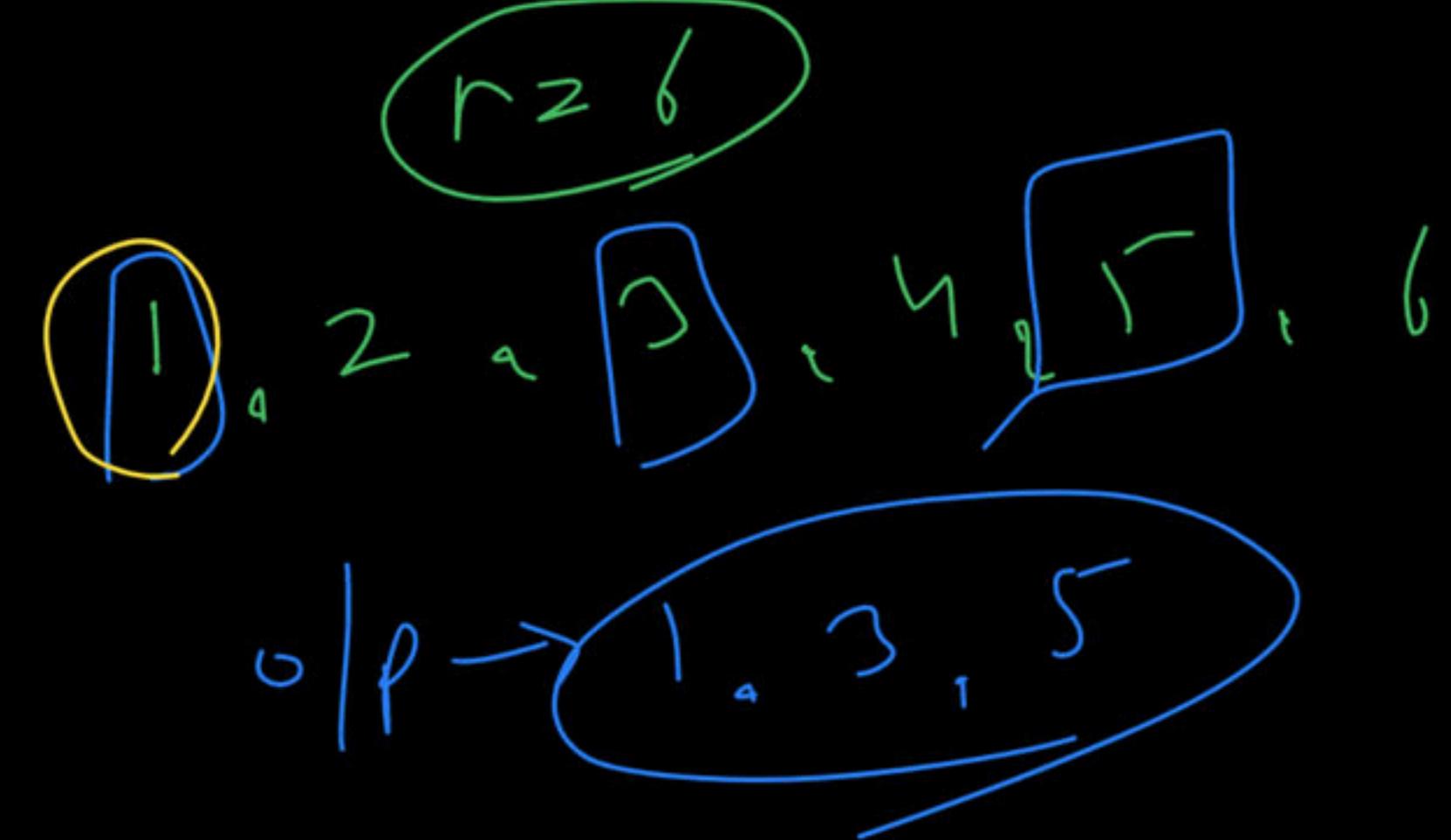
$sum = sum + num$

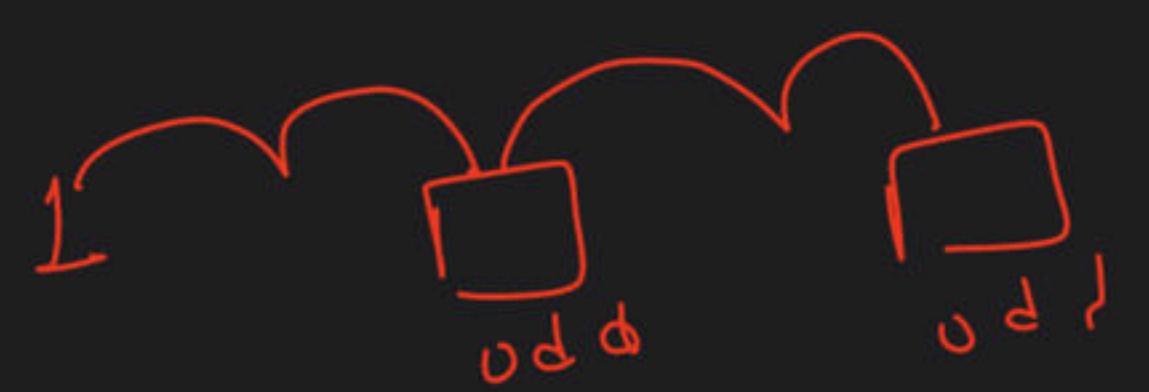
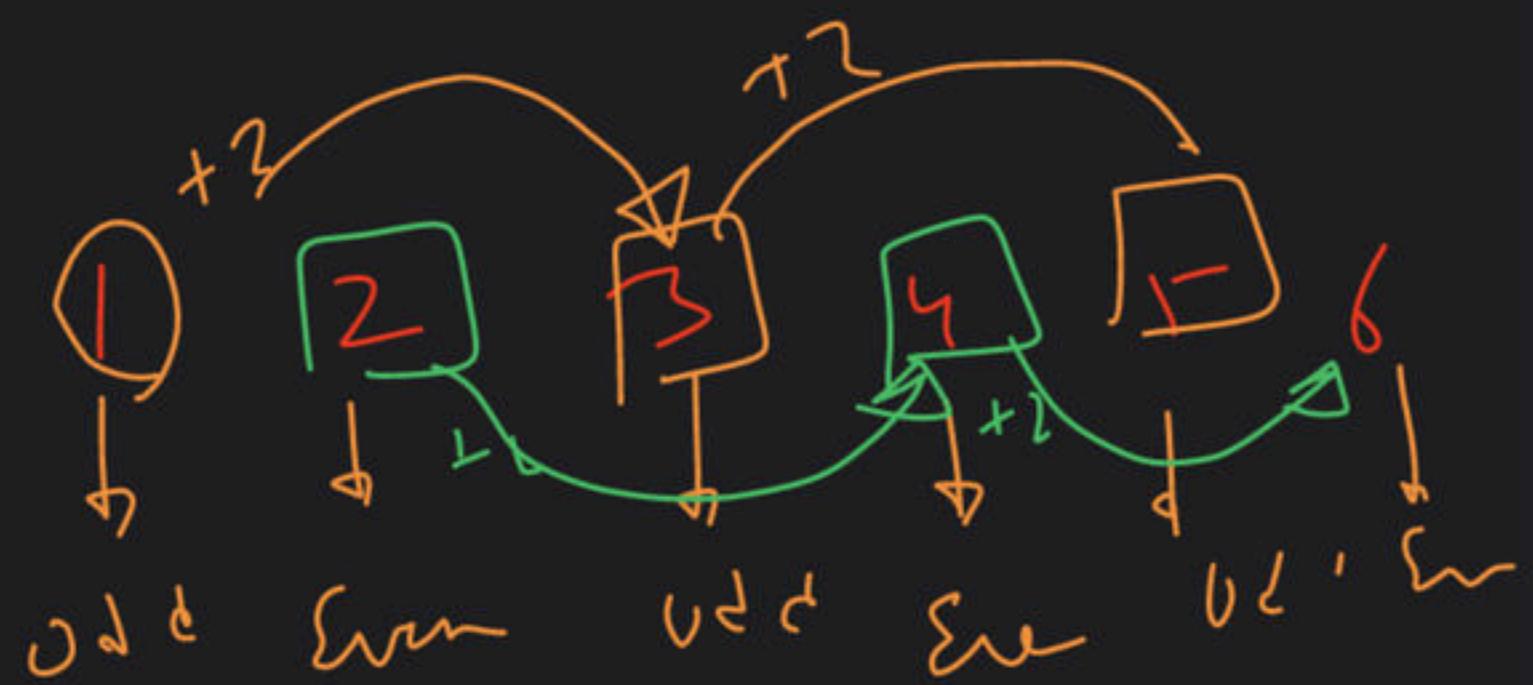
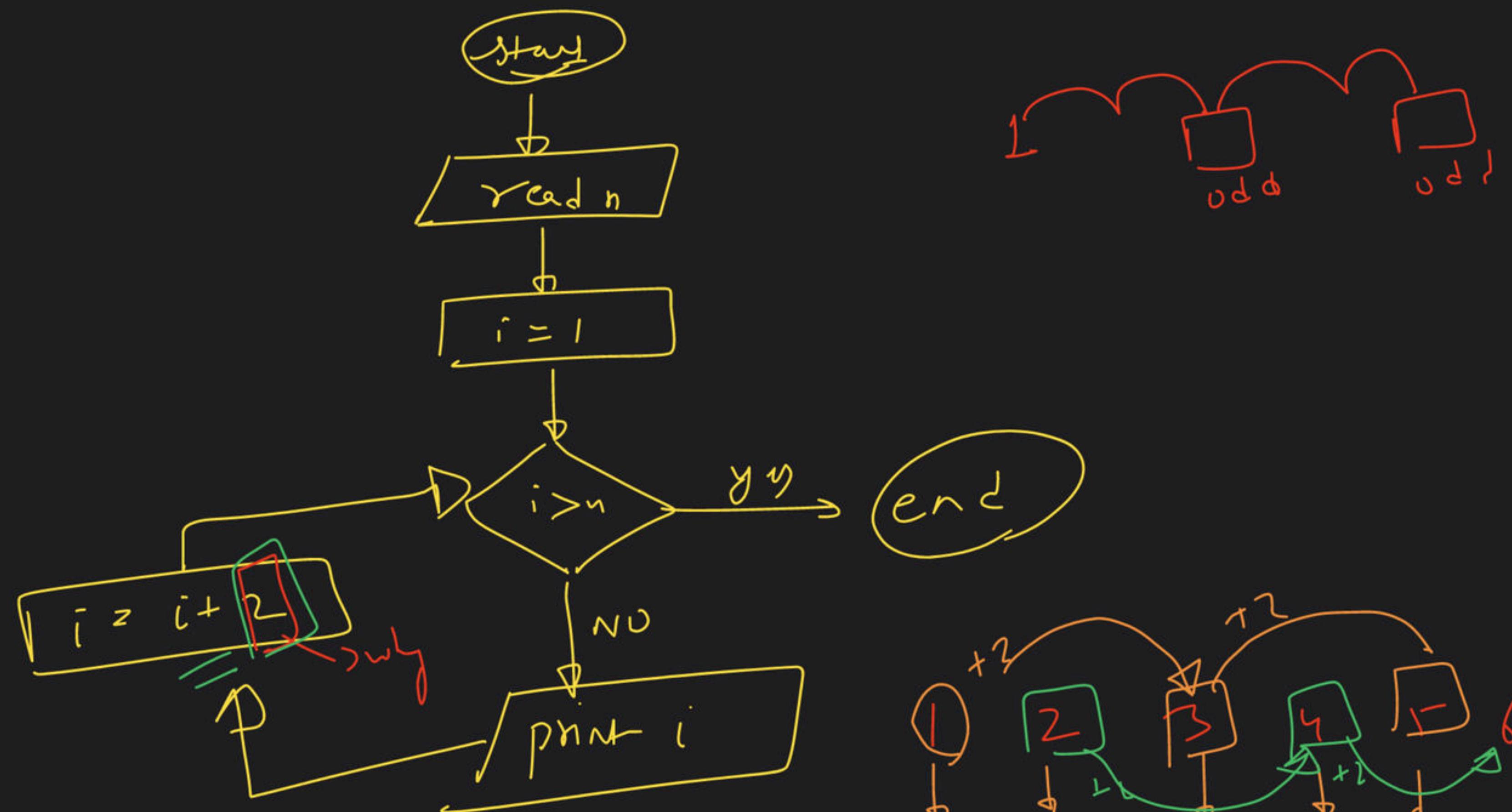
$i = i + 1$

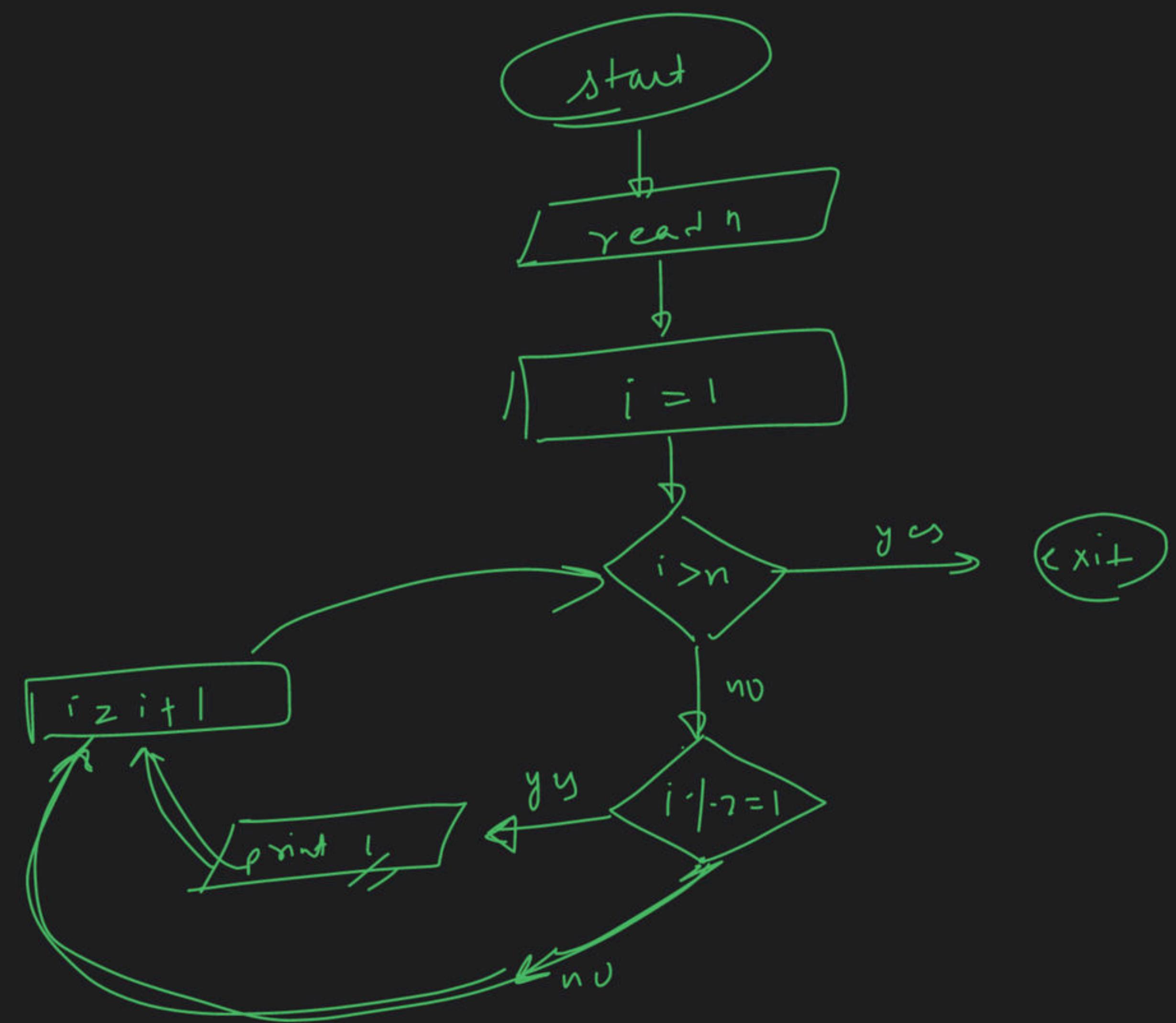
go to step 4

# Printing 1 to N but only odd numbers

for







# HomeWork:

12 bjc → next class

- Multiply 2 no. by taking input
- Find perimeter of a Triangle
- Find Simple Interest

- Print counting from N to 1
- Find Factorial of a number
- Check Prime or not

$$SI \rightarrow \frac{P \times R \times T}{100}$$

$$a > b$$

$$a + b + c$$

$$\frac{P \times S \times T}{100}$$

# HomeWork:

Blond h

- Valid Triangle or Not
- Printing 1 to N but only even numbers.
- Print max of three numbers

# **What is a Programming Language ?**

## **Why we need it ?**

- A Language using which, we can instruct the computer to carry out real life tasks and computations is called a programming language. It acts as a language in which we could easily express our thoughts to the machine.
- Like natural languages, programming language has a fixed set of rules according to which programs could be written in it. These programs are then converted into a language which machines can understand. This task is carried out by a special software called compiler.
- Every language has its own compiler/Interpreter.
- Once a program is compiled and linked, its executable is created and the computer can run our program now.