

DSA

→ getting started
Problem Solving

IP

- ↳ Under stand
- ↳ given values
- ↳ approach
- ↳ Program

P → Soln

Flowchart +
Pseudocode
Rough Soln

Program
HLL

o/i
Machine
Understandable

15

~~36~~
~~79~~
~~5~~
~~120~~
~~199~~

36

~~79~~

~~5~~

~~120~~

~~170~~

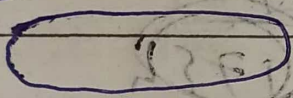
~~199~~

270

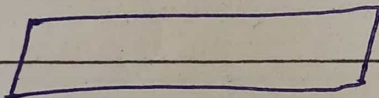
→ Flowchart

An approach in systematic form.

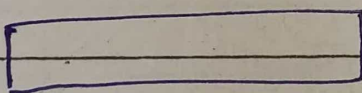
Component



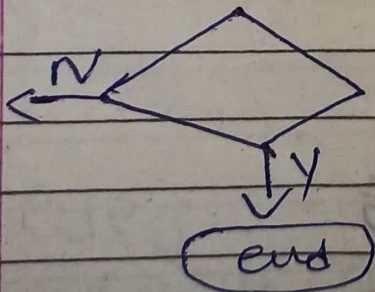
→ Terminator
(Start, end)



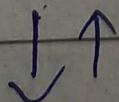
→ Input/output



→ Process



→ Decision Making

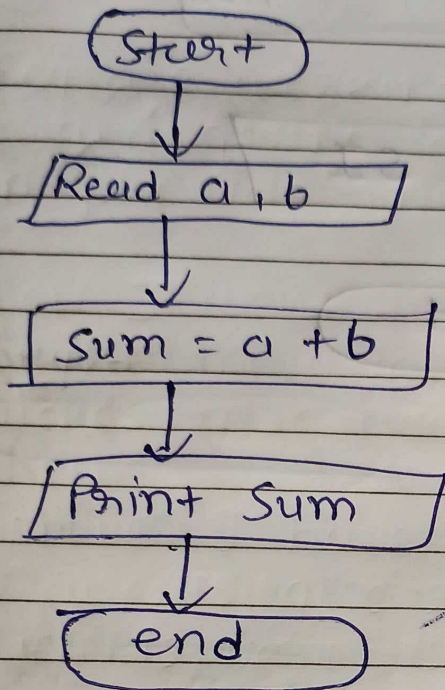


Arrows



Connector

★ Sum of 2 numbers



$a = 5$
 $b = 10$
 $Sum = a + b$

$\boxed{5}$ $\boxed{10}$
 a b

$\boxed{15}$
Sum

★ Pseudocode

generic

↳ way of representing logic

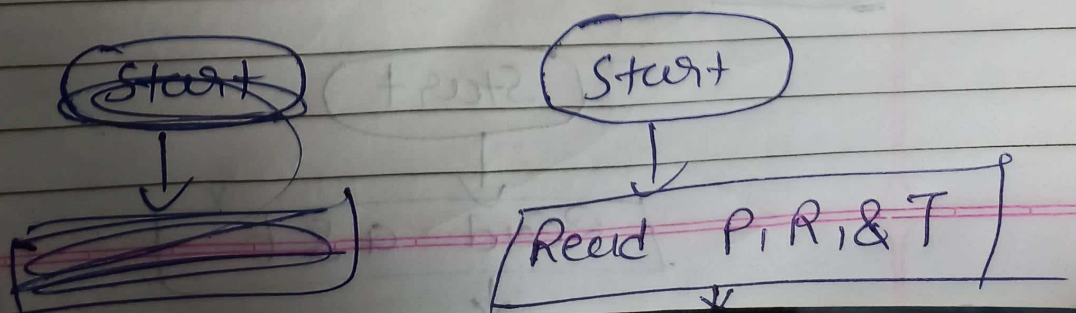
→ Sum of 2 no.

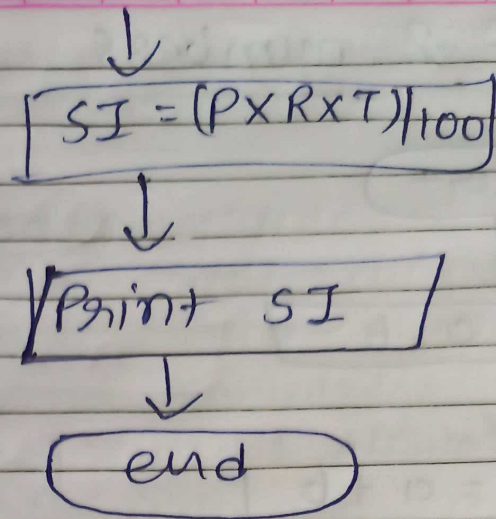
→ read 2 no. a & b

→ $Sum = a + b$

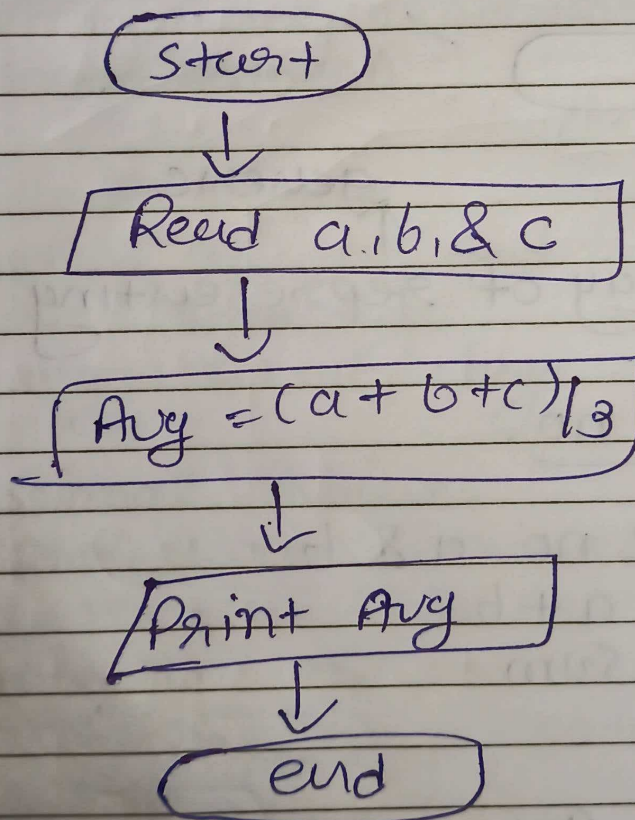
→ print Sum

★ Flowchart for Simple Interest

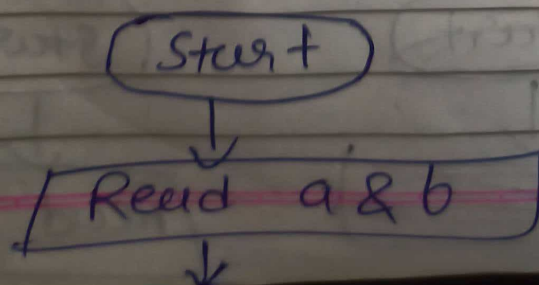


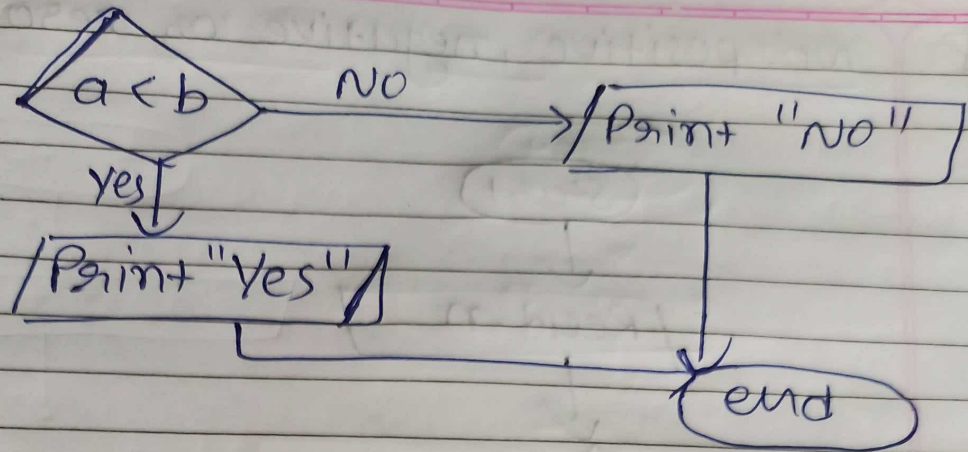


★ Avg. of 3 numbers



★ $a < b$ → Yes or No.





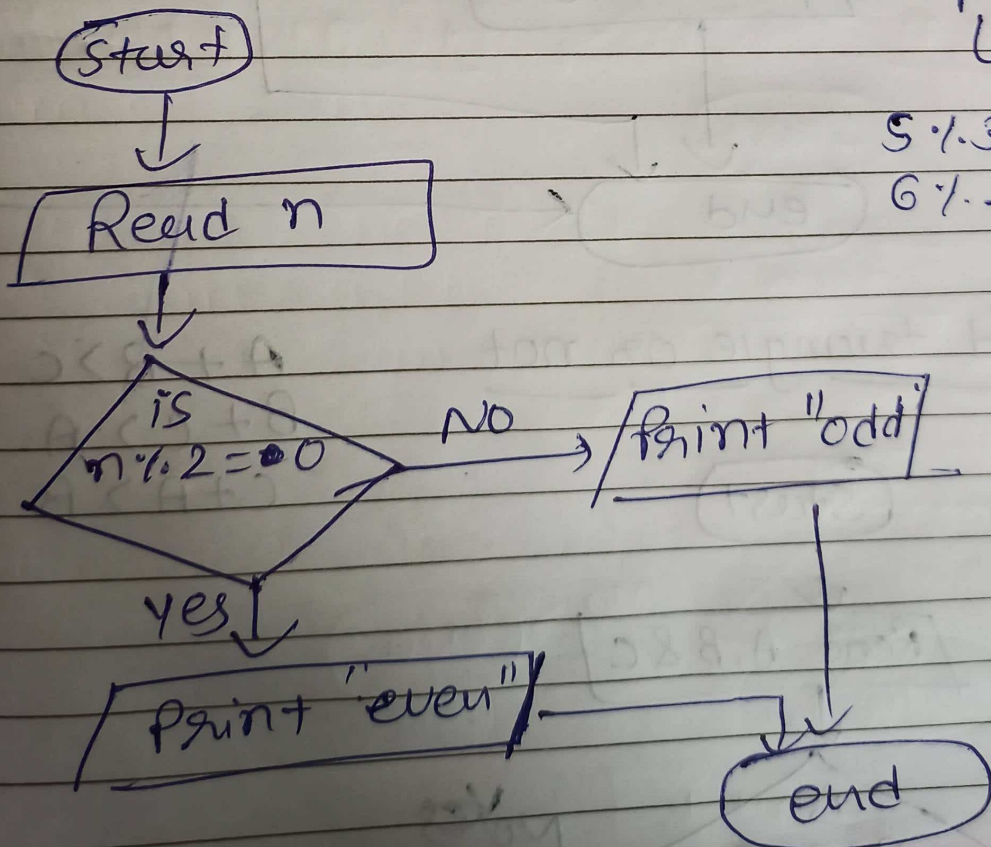
→ Pseudocode

→ Read a & b
 → if $a < b$ ~~then~~
 print "Yes"
 else
 print "No"

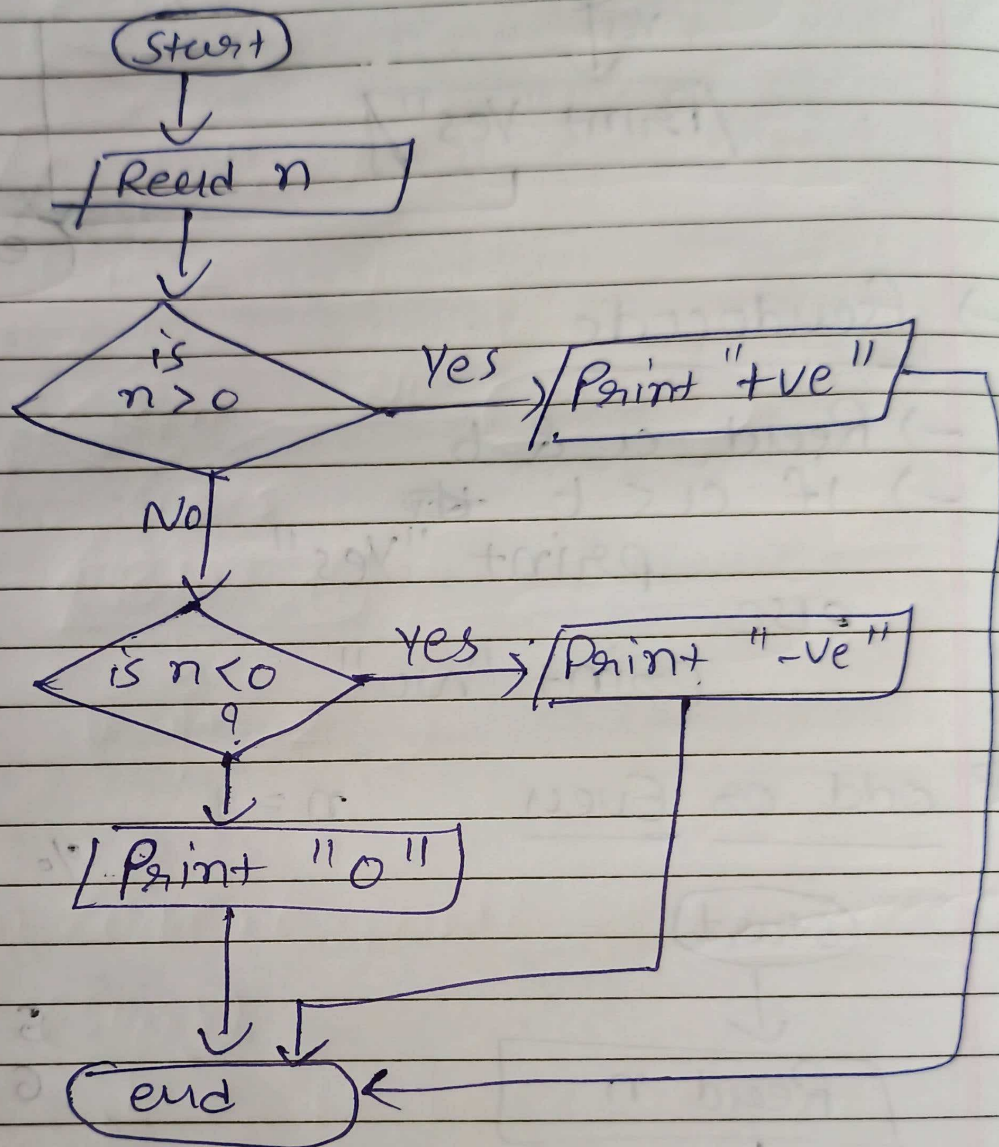
★ odd or Even $n = 3$

% operator
 ↳ remainder

$5 \% 3 \rightarrow 2$
 $6 \% 2 \rightarrow 0$

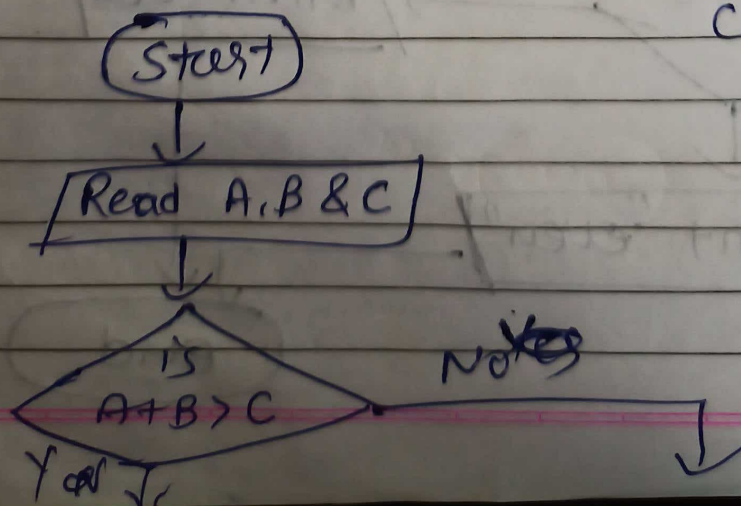


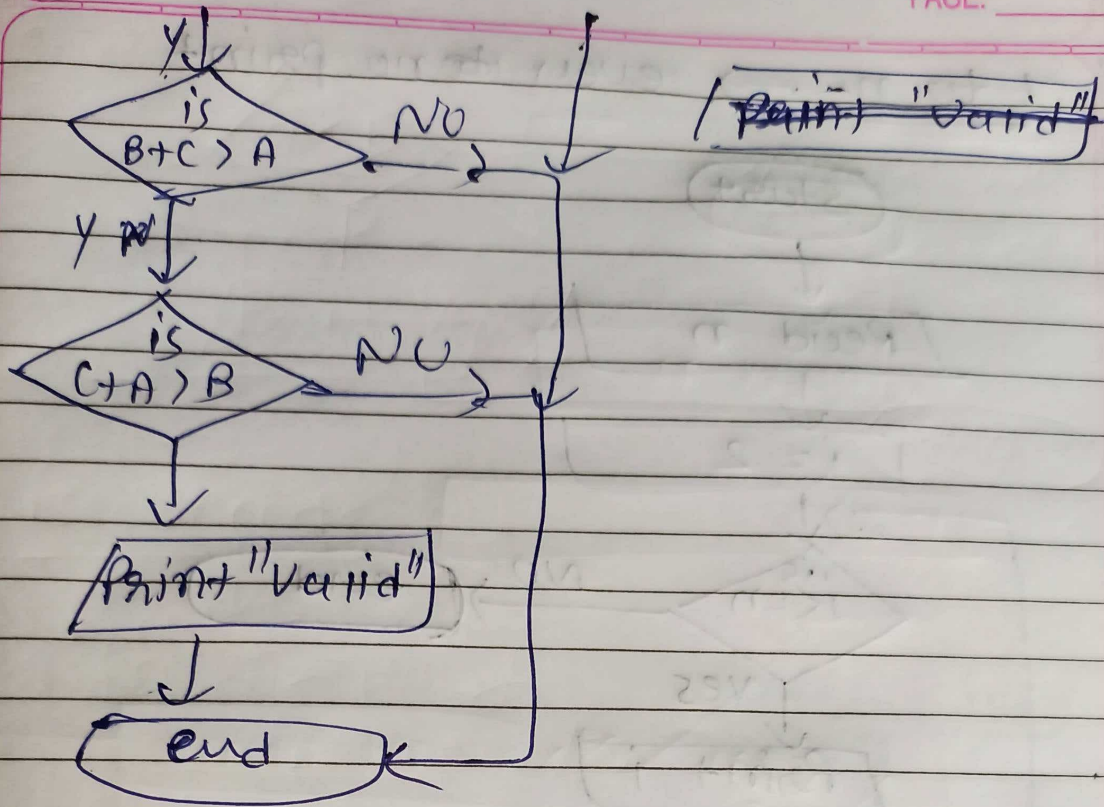
☆. no. positive, negative or zero.



☆ Valid triangle or not

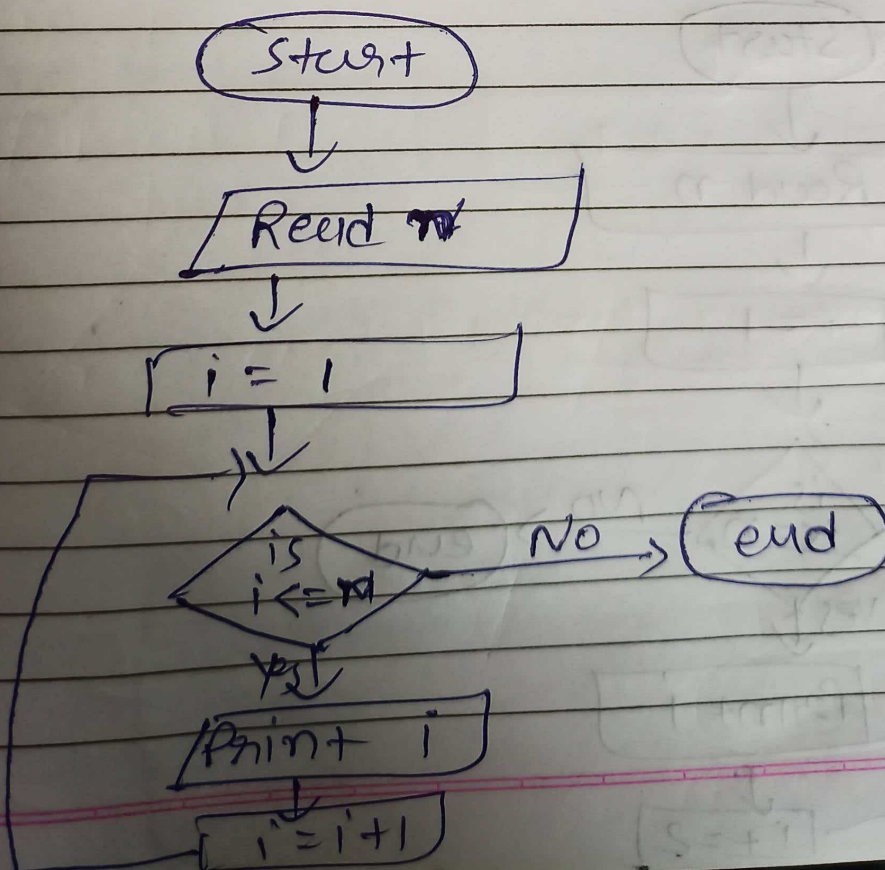
$$\begin{aligned}
 A + B &> C \\
 B + C &> A \\
 C + A &> B
 \end{aligned}$$



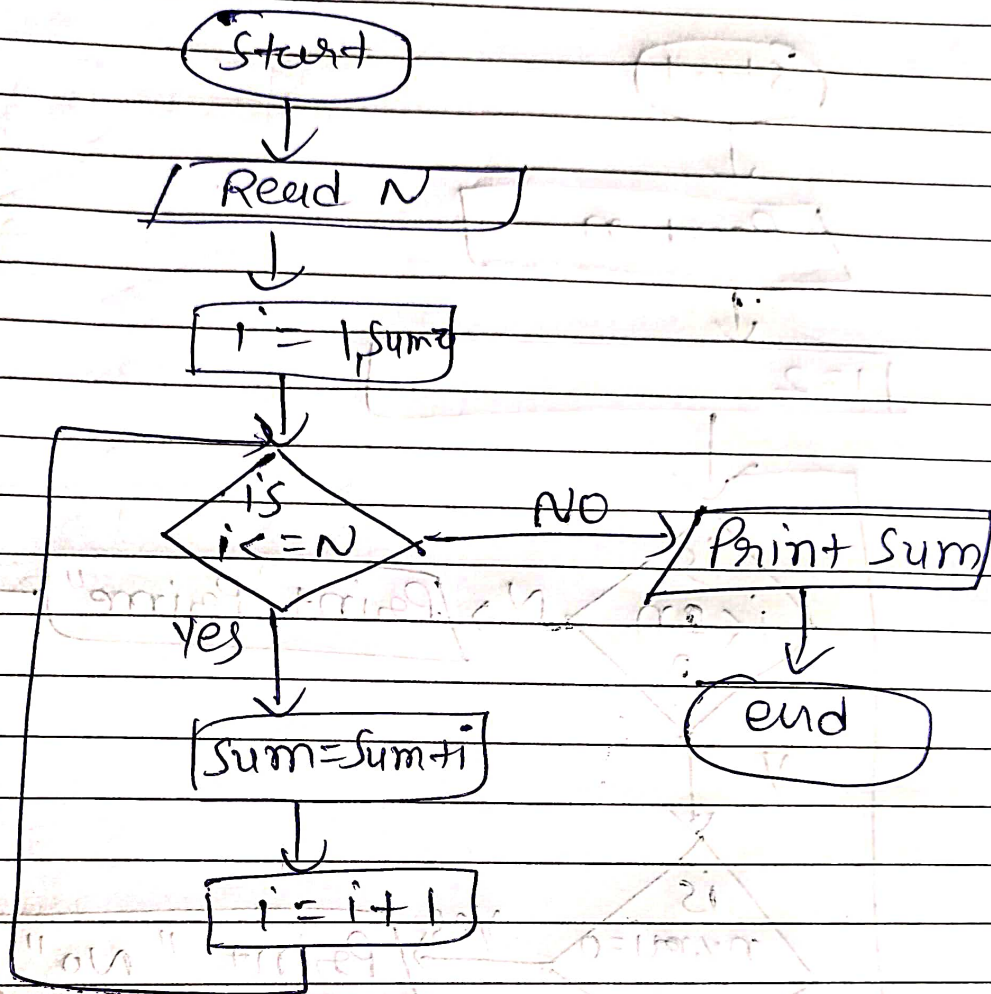


★ loops

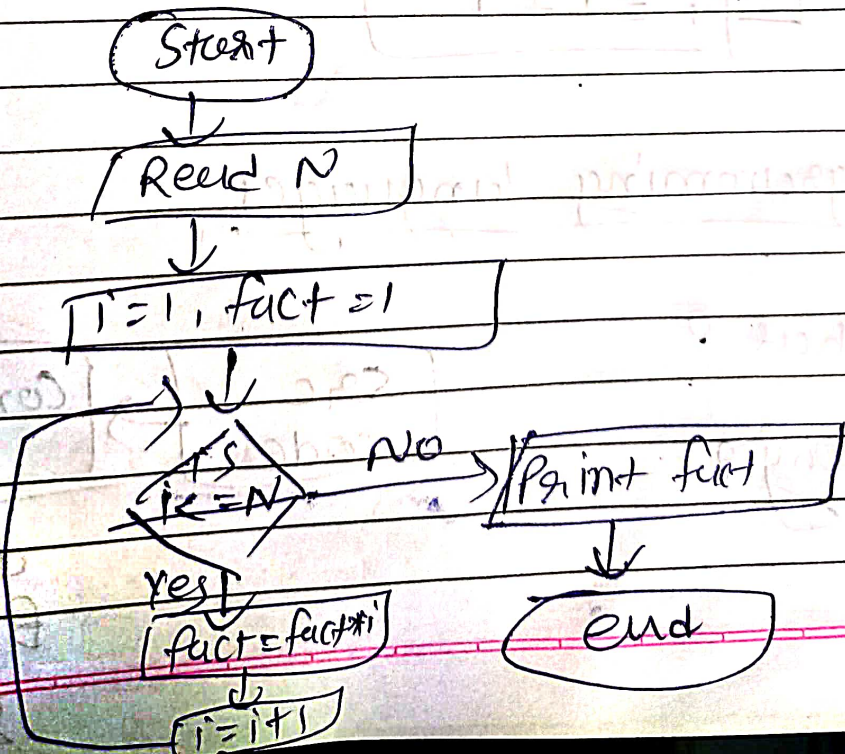
print 1 to N



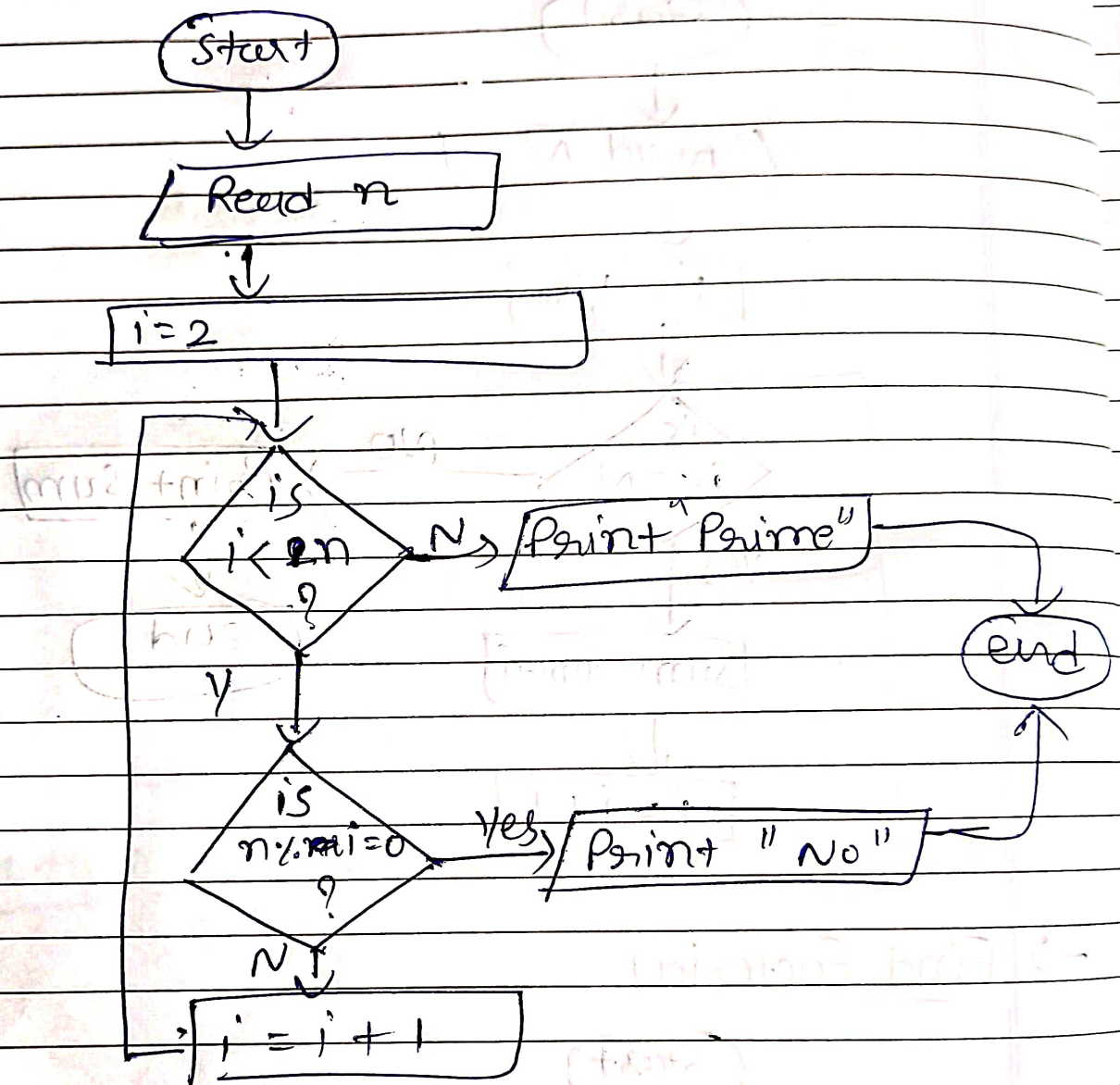
→ Find Sum 1 to N



→ Find Factorial



→ Check prime or not



★ Programming languages

What ?

Why ?

