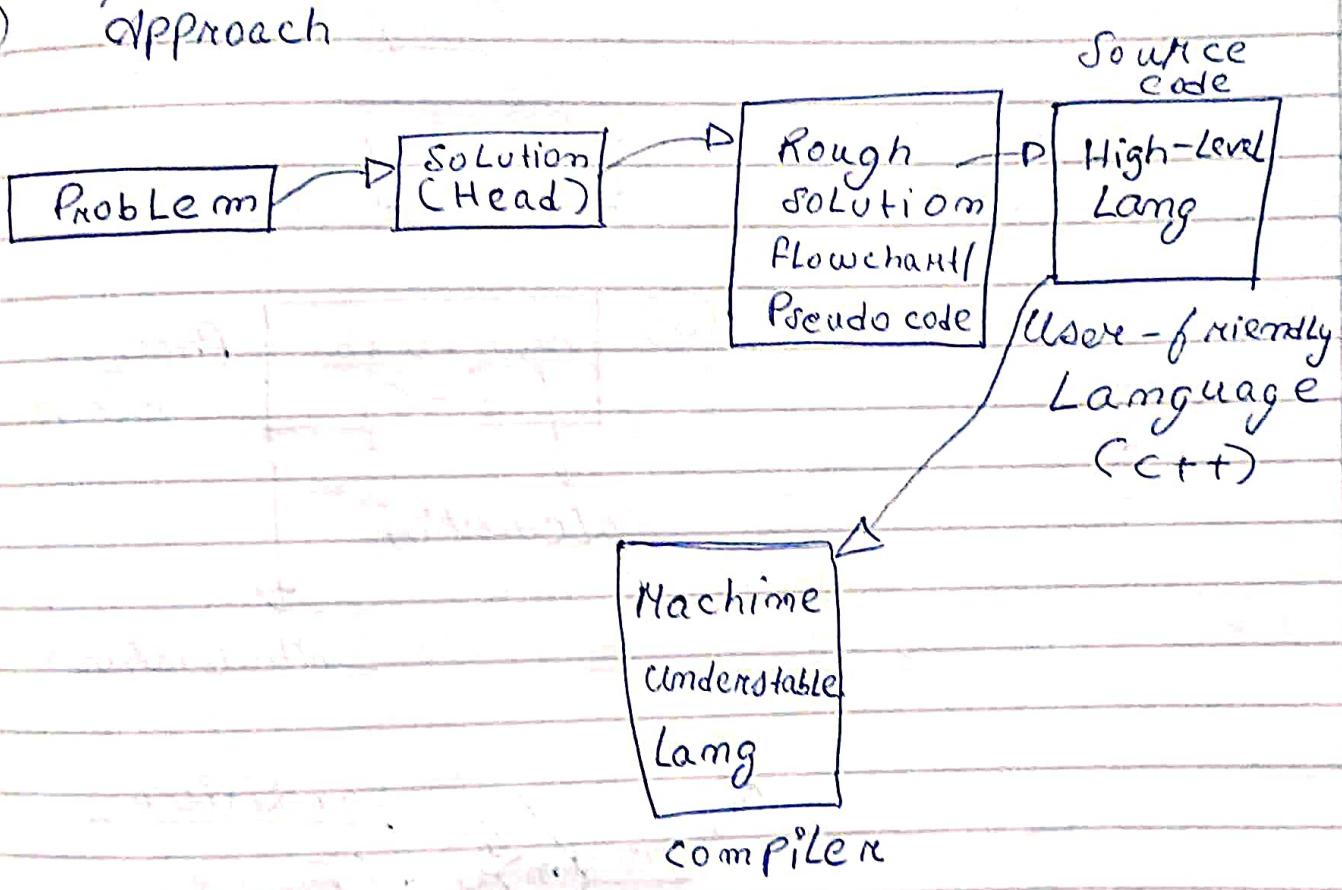


Programming Fundamentals

① Thought to solve Problem.

- ① Understand the Problem.
- ② i/p values.
- ③ Approach



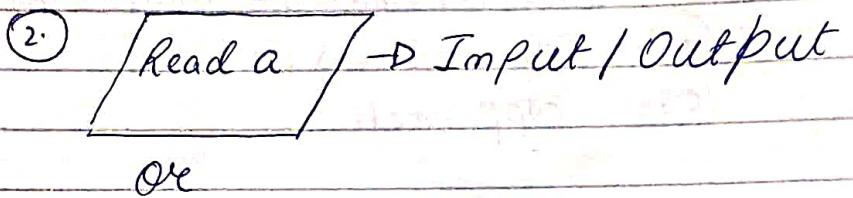
② Using Comp to solve Problem.

$$13/2 = N \in \mathbb{N} \rightarrow 1 \quad (n = 13)$$

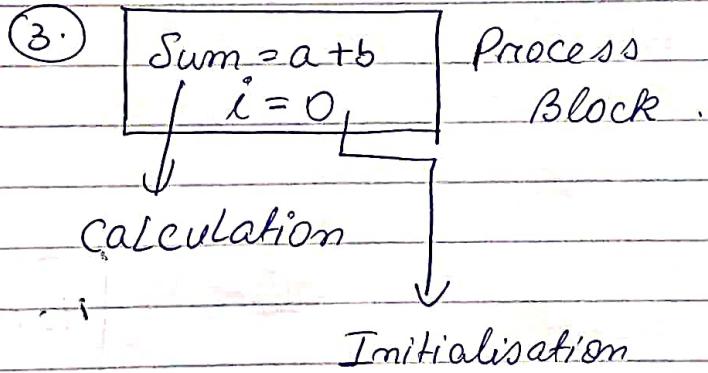
- Prime or not
- I
- itself

Flowchart :- Flowchart is a graphical representation of an algorithm.

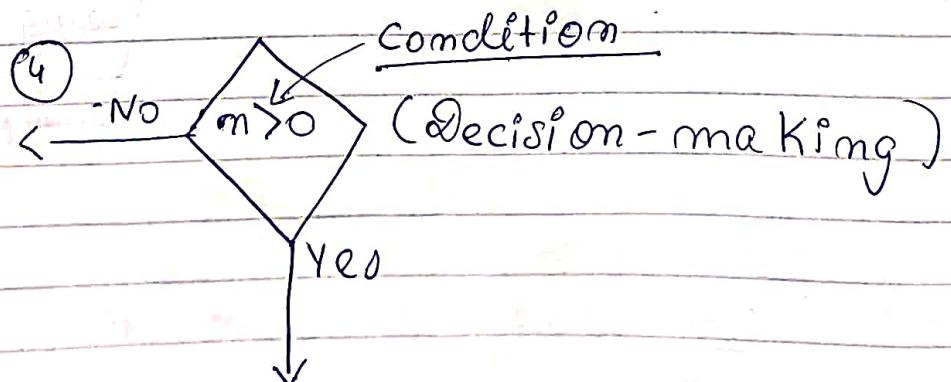
Components :- ① Start / End → Terminator



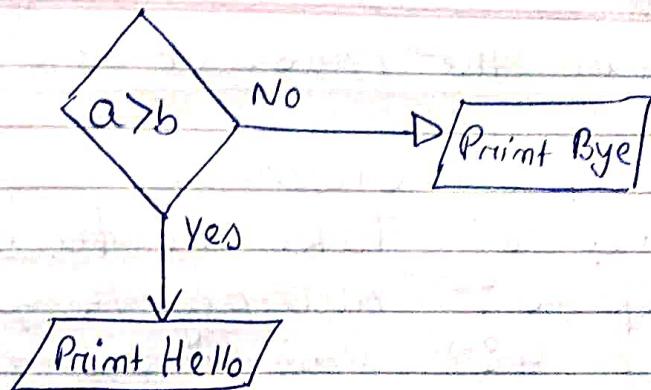
[Print a]



Initialisation



Eg: $a = 3, b = 5$



Output = Print Bye

⑦ flows → (using arrows)

⑧ Connectors → (A) function

Pseudo Code :- (fake Code) It is an algorithm (readable by a user) for approach (intexture - form). Step in the process for developing program.

Step 1 :- Start

Step 2 :- Enter two Numbers A & B .

Step 3 :- Add the numbers together

Step 4 :- Print Sum

Step 5 :- End .

HICW

Differ → a - b → Step 1 :- Start

Step 2 :- Enter two Num
A & B

Step 3 :- Sub the Num together

Step 4 :- Print Num

Step 5 :- End .

Print the Product a & b

Step 1: Start

Step 2: Take the two numbers a & b

Step 3: \rightarrow Multiply the product = $a \times b$.

Step 4: \rightarrow Print the product.

Step 5: Exit.

Avg of two Numbers

Step 1: \rightarrow Start

Step 2: \rightarrow Take the two numbers a & b

Step 3: \rightarrow average of two numbers
$$\text{avg} = \frac{a+b}{2}$$

Step 4: \rightarrow Print the Avg.

Step 5: \rightarrow Exit.

Add 2 numbers by taking Input

Step 1: \rightarrow Start . input

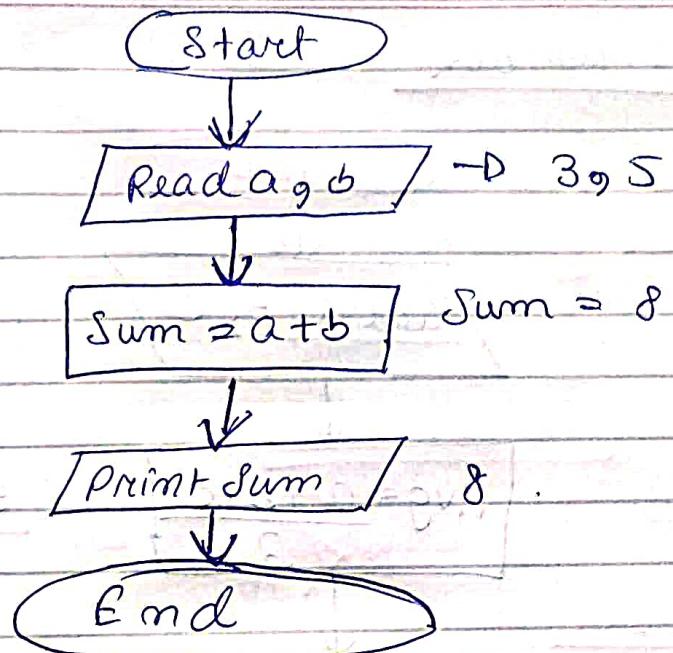
Step 2: \rightarrow Take the two numbers a & b.

Step 3: \rightarrow Add the 2 numbers

Step 4: Print the sum

Step 5: Exit

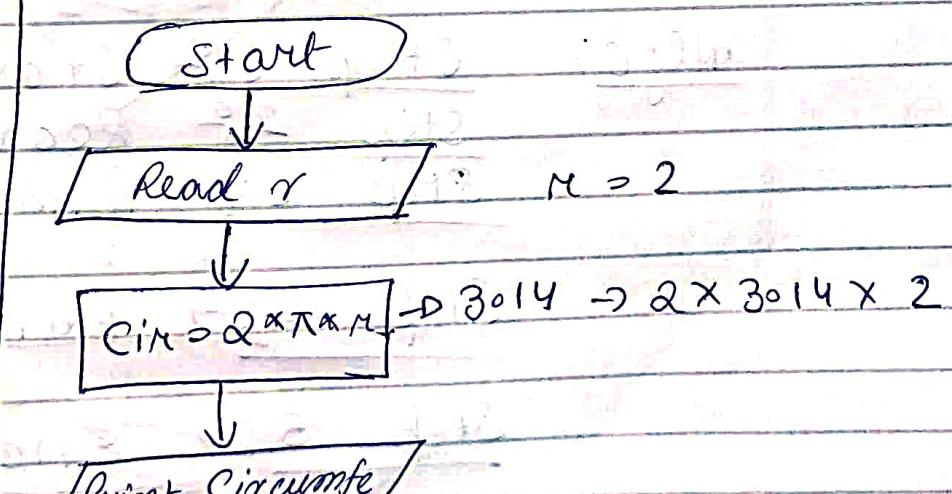
Flowchart: -

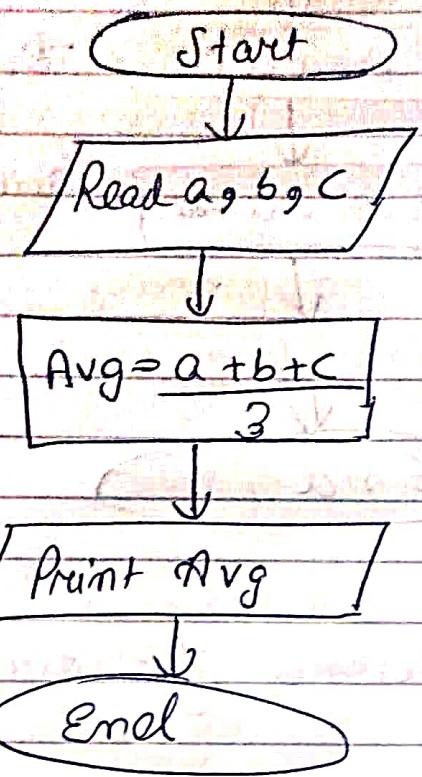


Find Circumference of a circle

$$\rightarrow 2\pi r.$$

Step 1: Start
 Step 2: → Read r
 Step 3: Taking Circumference
 $\rightarrow 2 \times \pi \times r$ (formula)
 Step 4: → Print circum
 Step 5: Stop .



Avg 3 numbers

Alg0:-

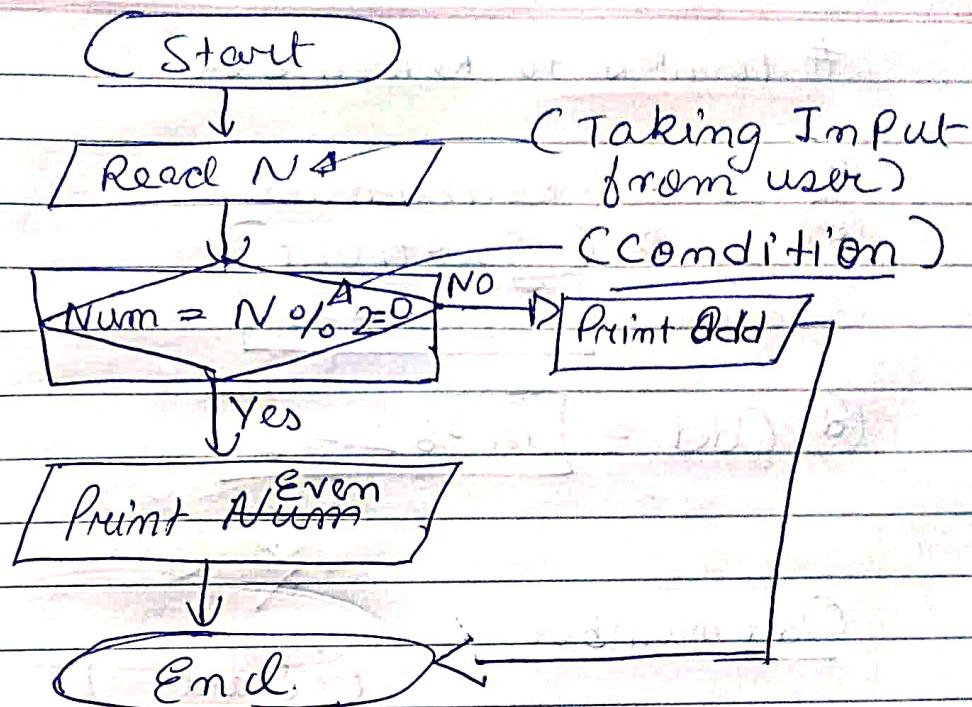
- Step 1 :- Start
- Step 2 :- Read 3 numbers a, b, c .
- Step 3 :- Formula $\rightarrow \frac{a+b+c}{3}$.

Step 4 :- Print Avg.

Step 5 :- End.

number is

Check, even or odd :



Algorithm :- Step 1 :- Start .

Step 2 :- Read N (From user)

Step 3 :- Formula to check Even or Odd - $\rightarrow \%$

$$\text{Num} = (N \% 2)$$

Step 4 :- If remainder comes out will be Even,
else if rem comes 1 then it will be odd.

Step 5 :- Print Num .

Step 6 :- End

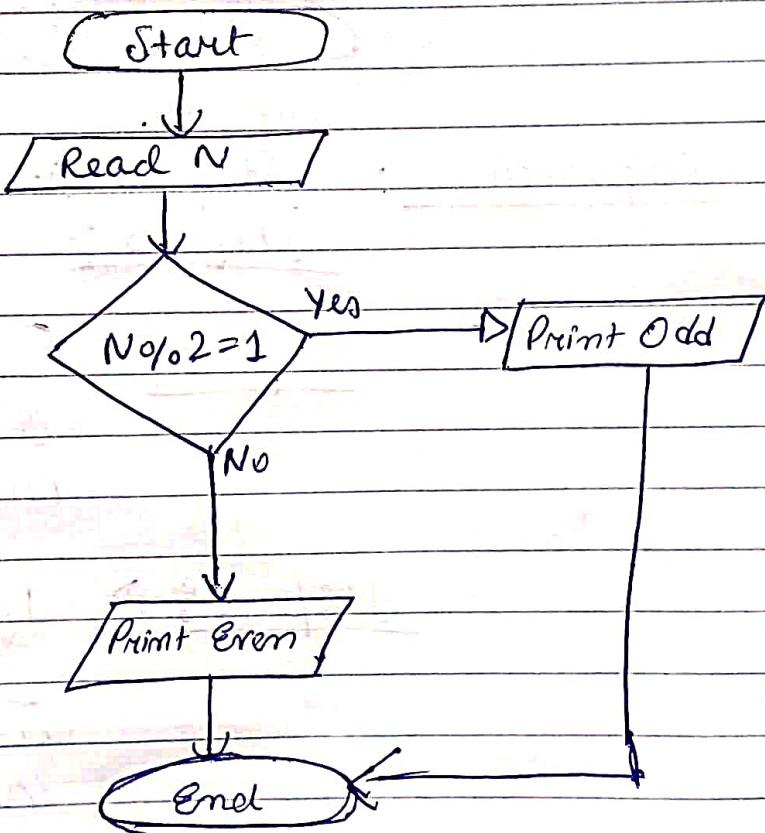
Points to Remember

(a) $\%$ \rightarrow (Remainder)

(b) $/ \rightarrow$ (Quotient)

(c) Even = $m \% 2 = 0$

(d) Odd = $m \% 2 = 1$

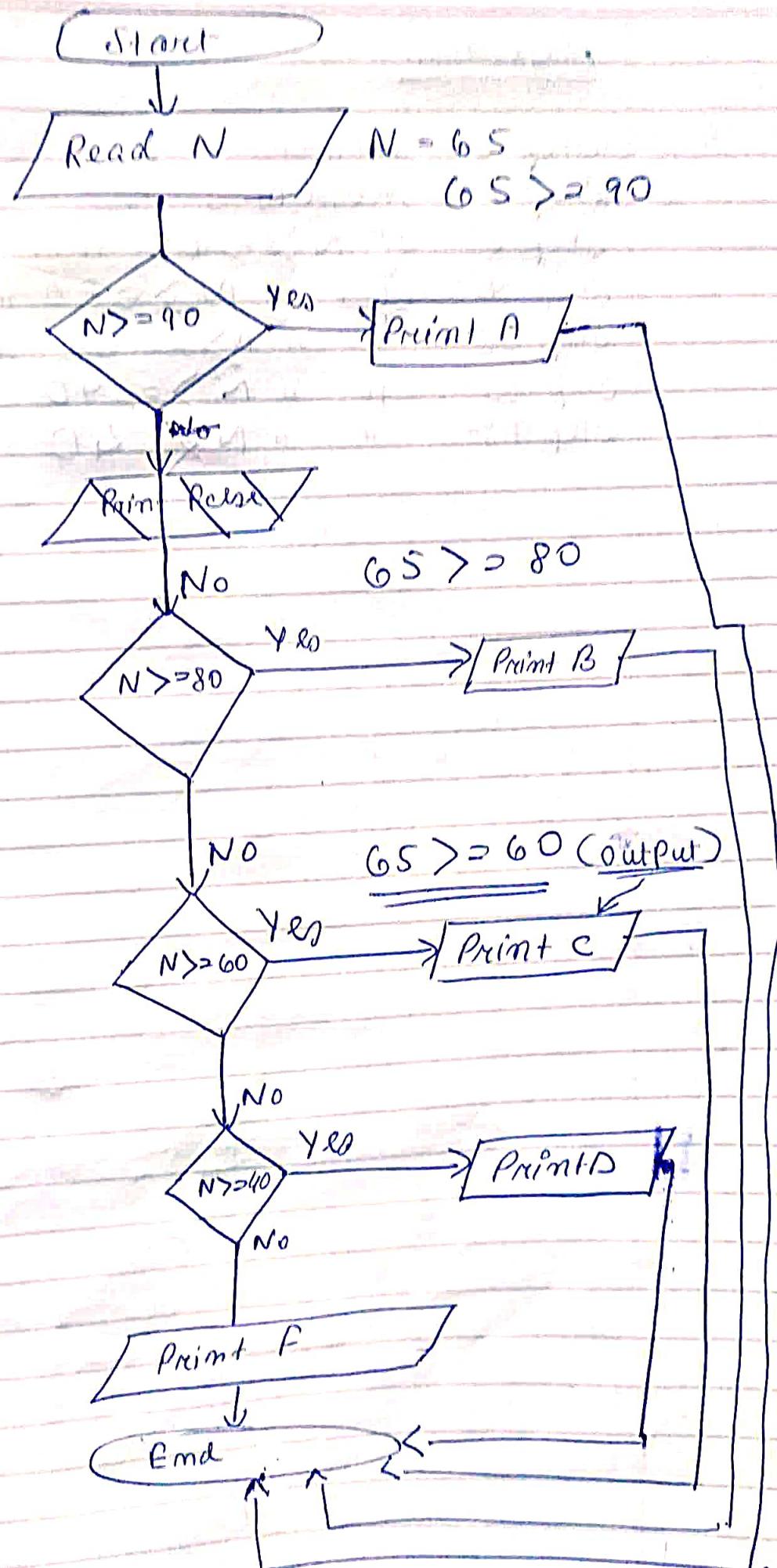
Odd numberStudents & Grade Flowchart

$>= 90 \rightarrow A \quad < 40 \Rightarrow P$

$>= 80 \rightarrow B$

$>= 60 \rightarrow C$

$>= 40 \rightarrow D$



Algorithm

Step 1 :- Start

Step 2 :- Read N

Step 3 :- if $N \geq 90$ then Print A

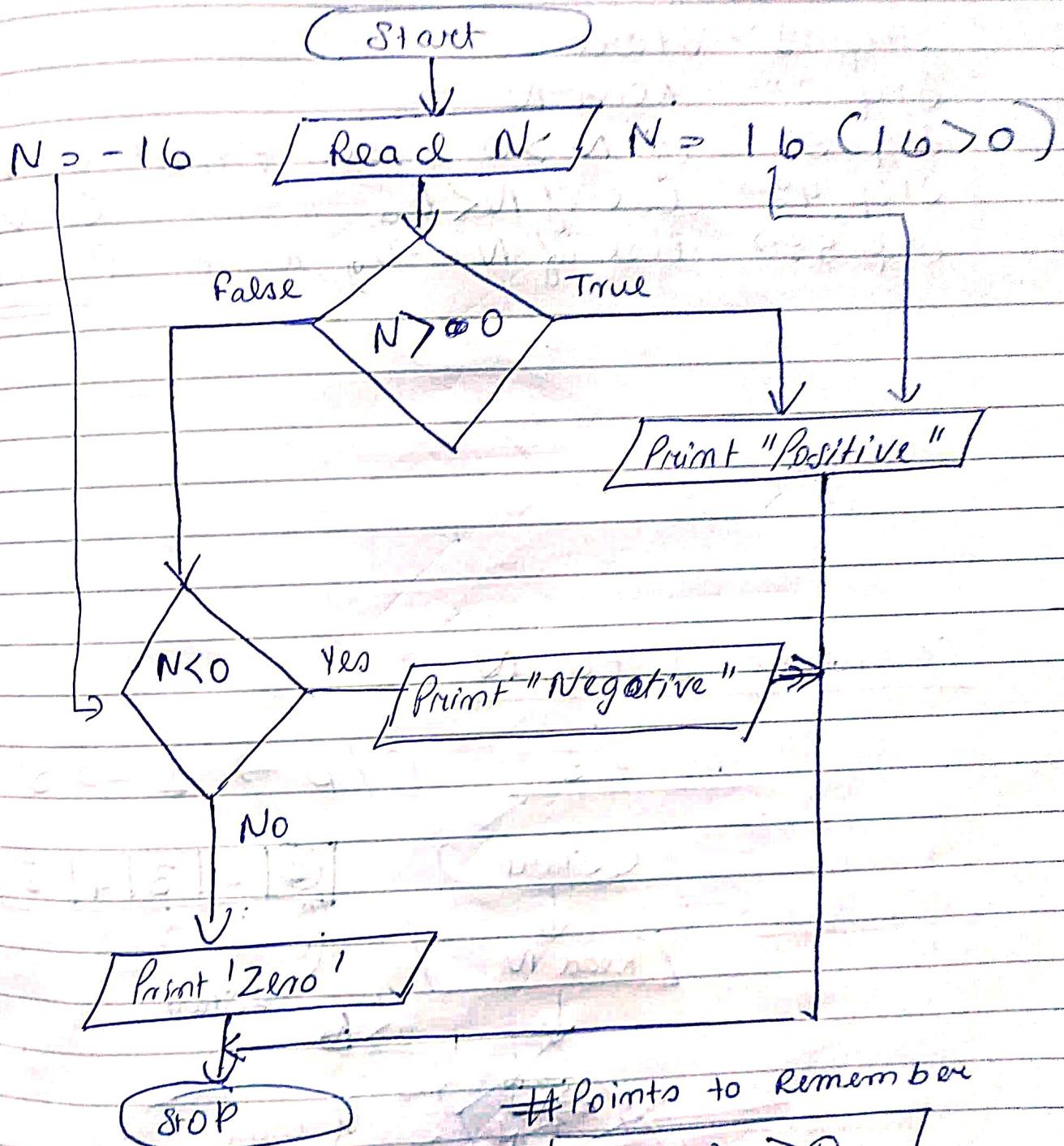
Step 4 :- else if $N \geq 80$ then B

Step 5 :- " " $N \geq 60$ then C

Step 6 :- " " $N \geq 40$ then D

Step 7 :- " " $N < 40$ then E

Check number is +ve, -ve, 0.



Points to Remember

+ve	\rightarrow	> 0
-ve	\rightarrow	< 0
0	\rightarrow	$= 0$

Algorithm

Step 1 :- Start

Step 2 :- Read N

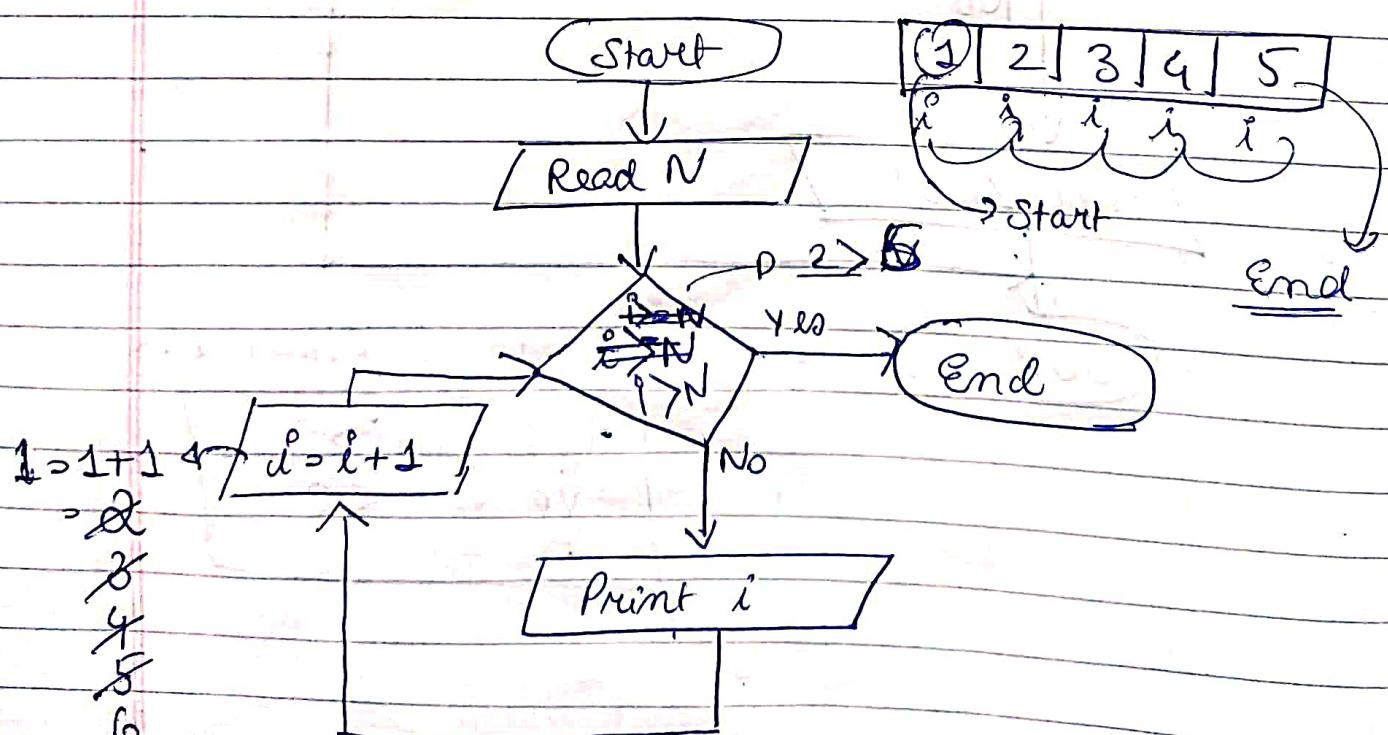
Step 3 :- if $N > 0$, then Print (+ve)

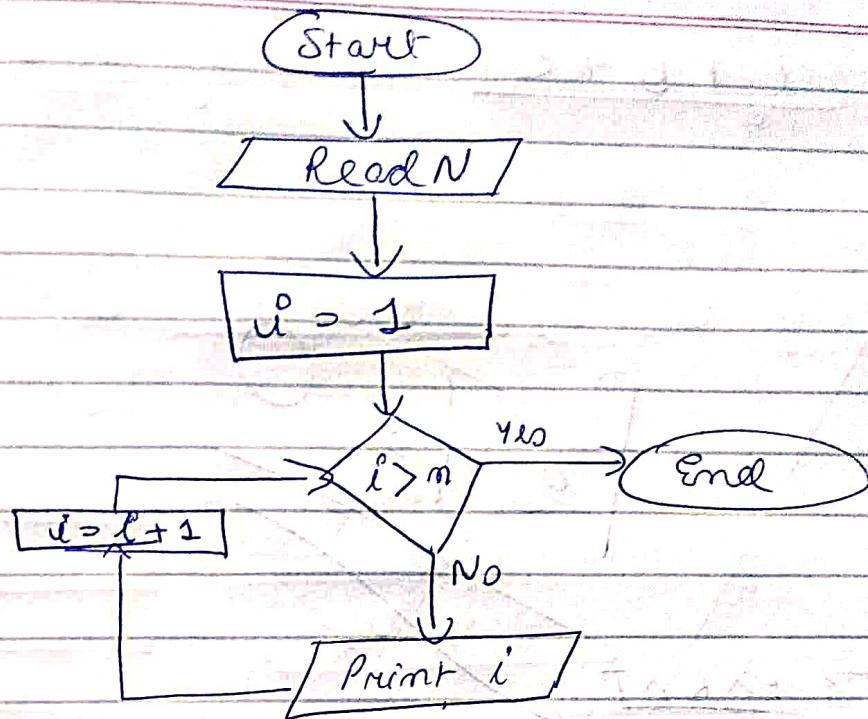
Step 4 :- else if $N < 0$, n n (-ve)

Step 5 :- else if $N = 0$, n n (Print 0)

Counting 1 to N

$$m = 5 \quad O/p \rightarrow 1 \rightarrow 5$$





Algorithm :-
 Step 1:- Start.
 Step 2:- Read N
 Step 3:- take $i = 1$ starting
 Step 4:- if $i > N$, if it is Yes, then
 Step 5:- Else if Print i and then increment i to cover till N.

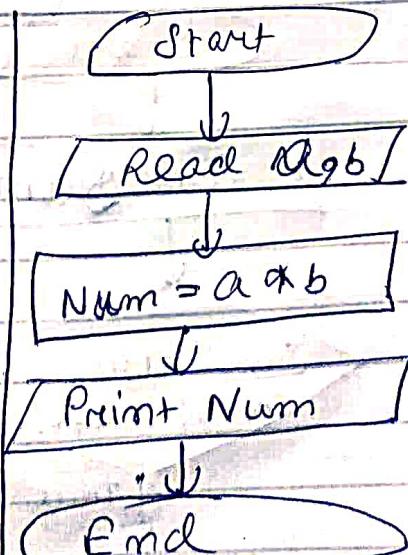
Homework +1, to cover till N.

① Multiply 2 no. by taking input.

Step 1:- Start
 Step 2:- Read a, b
 Step 3:- Multiply
 2 number which
 is used to store the
 value to the Num

Step 4:- Print
 Num

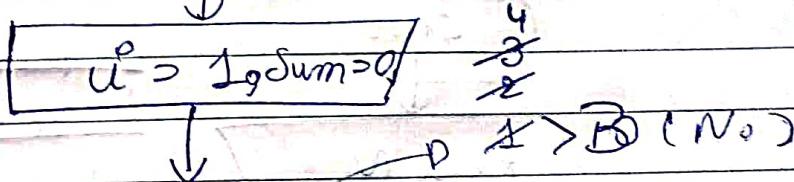
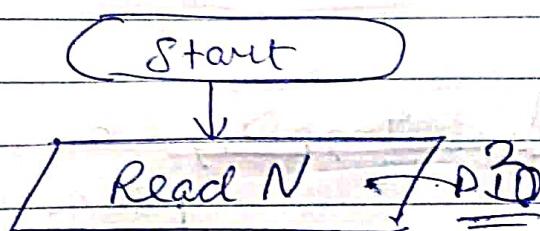
Step 5:- STOP.



odd N numbers from user

$n = 3$

$$\begin{aligned} & 2 \leftarrow J^{\text{st}} \\ & 4 \leftarrow J^{\text{nd}} \\ & 6 \leftarrow J^{\text{rd}} \end{aligned} \quad \left. \begin{aligned} & 2 + 4 \\ & 4 + 6 \\ & 6 + 2 \end{aligned} \right\} \text{Sum}$$



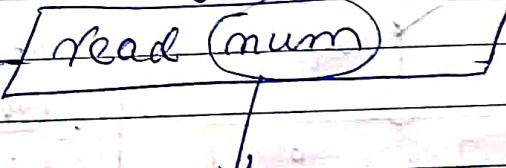
$i > N$ Yes → Print Sum → End

$i = i + 1$

No

$$2 + 4 + 6 = 12$$

$\text{sum} = \text{sum} + \text{num}$



$(0 + 0 = 2)$

A number should

be included to

sum e.g. 2, 4, 6

$0 \rightarrow 5 \rightarrow \text{No}$

$1 \rightarrow 5 \rightarrow \text{(No)}$

Algorithm :- Step 1 :- Start

Step 2 :- Read N

Step 3 :- Take $i = 1, \text{sum} = 0$

Step 4 :- Check $i > N$, yes print sum then exit.

Step 5 :- Else if, read num which is given by user and then $\text{sum} = \text{sum} + \text{num}$.

Step 6 :- Increment $i + 1$, move to again same loop.

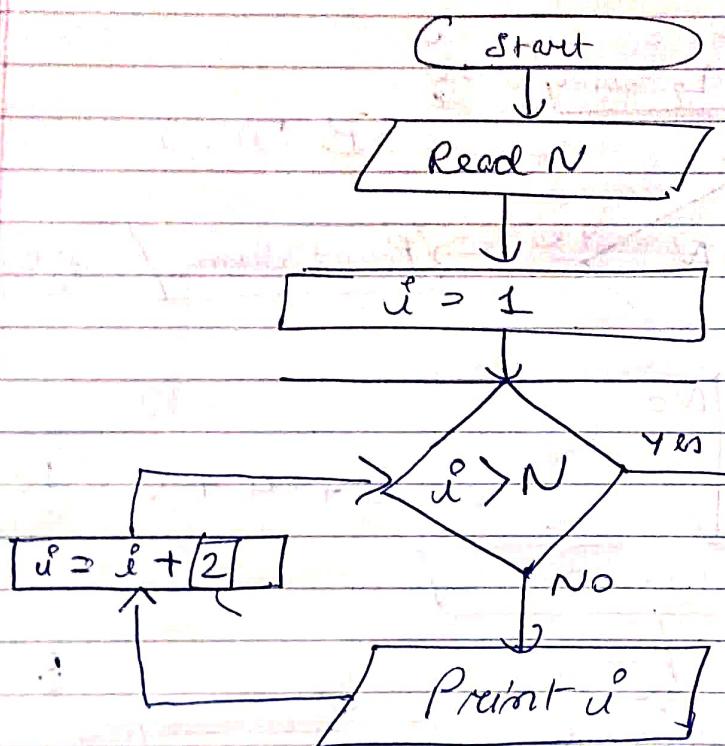
Step 7 :- End.

Priming 1 to N but only odd numbers

$$1 \leq i \leq 6$$

1, 2, 3, 4, 5, 6

Algorithm



Step 1: Start

Step 2: Read N

Step 3: Take $i = 1$

Step 4: Condition

If $i > N$, if it is true end from loop.

Step 5: Else if, Print i , increment

End $[i+2]$,

repeat loop

until i will not

OverLoop from N.

Step 6: Stop.

1 2 3 4 5 6
↑ ↑ ↑ ↑ ↑ ↑

Another

when we do with +1.

(Another Traditional Method)

Algorithm

Step 1 :- Start

Step 2 :- Read N

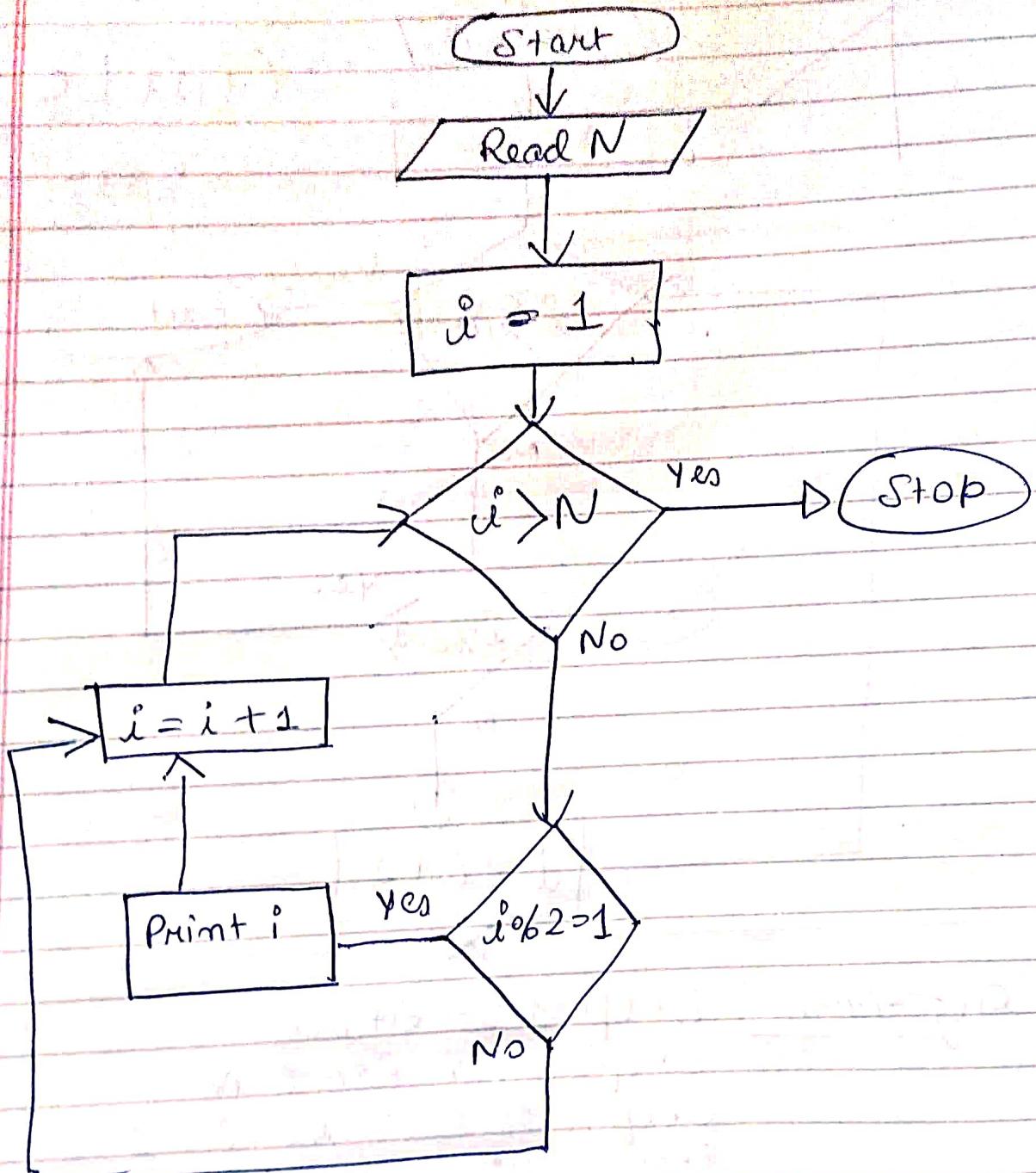
Step 3 :- Take i = 1

Step 4 :- check condition

if $i > N$, if it is yes exit from loop

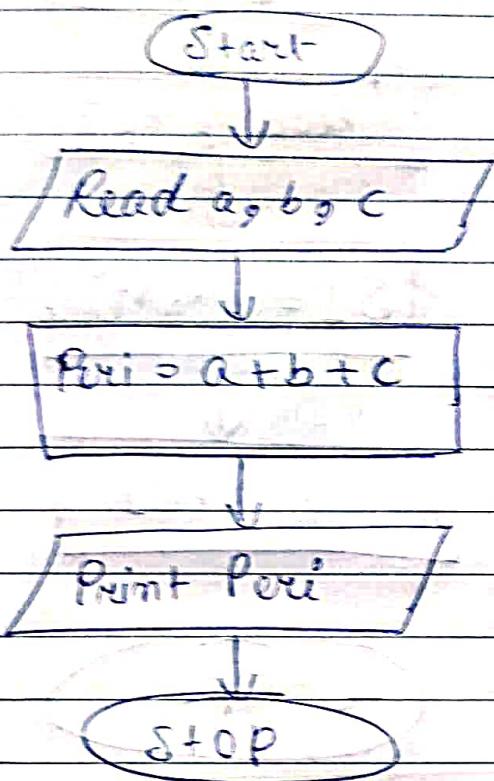
Step 5 :- Else if, it is no check
condition ($i \% 2 = 1$) if it is yes,
~~increment~~ print i, then increment
+ 1, if it does not satisfy the
Condition directly increment + 1.

Step 6: Stop



Homework

• find perimeter of a Triangle (Flowchart)



Algorithm

Step 1 :- Start

Step 2 :- Read a, b, c

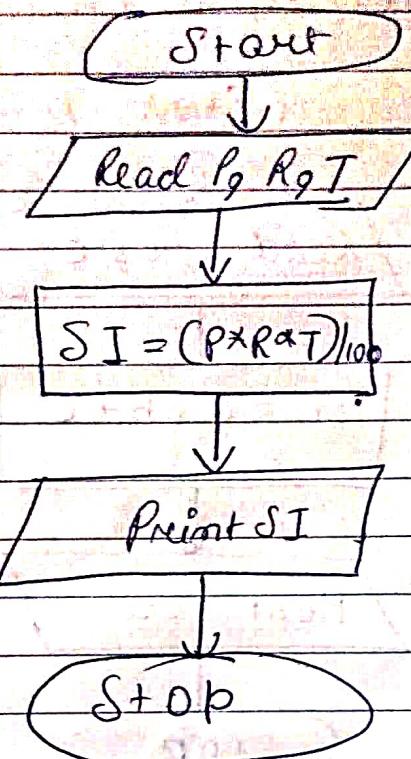
Step 3 :- Sum of all three values of $a + b + c$
the remainder stored in Perimeter.

Step 4 :- Print Perimeter

Step 5 :- Stop

• Find simple Interest

(Flowchart)



Algorithm

Step 1 :- Start

Step 2 :- Read P → Principal,
R → Rate,
T → Time (years)

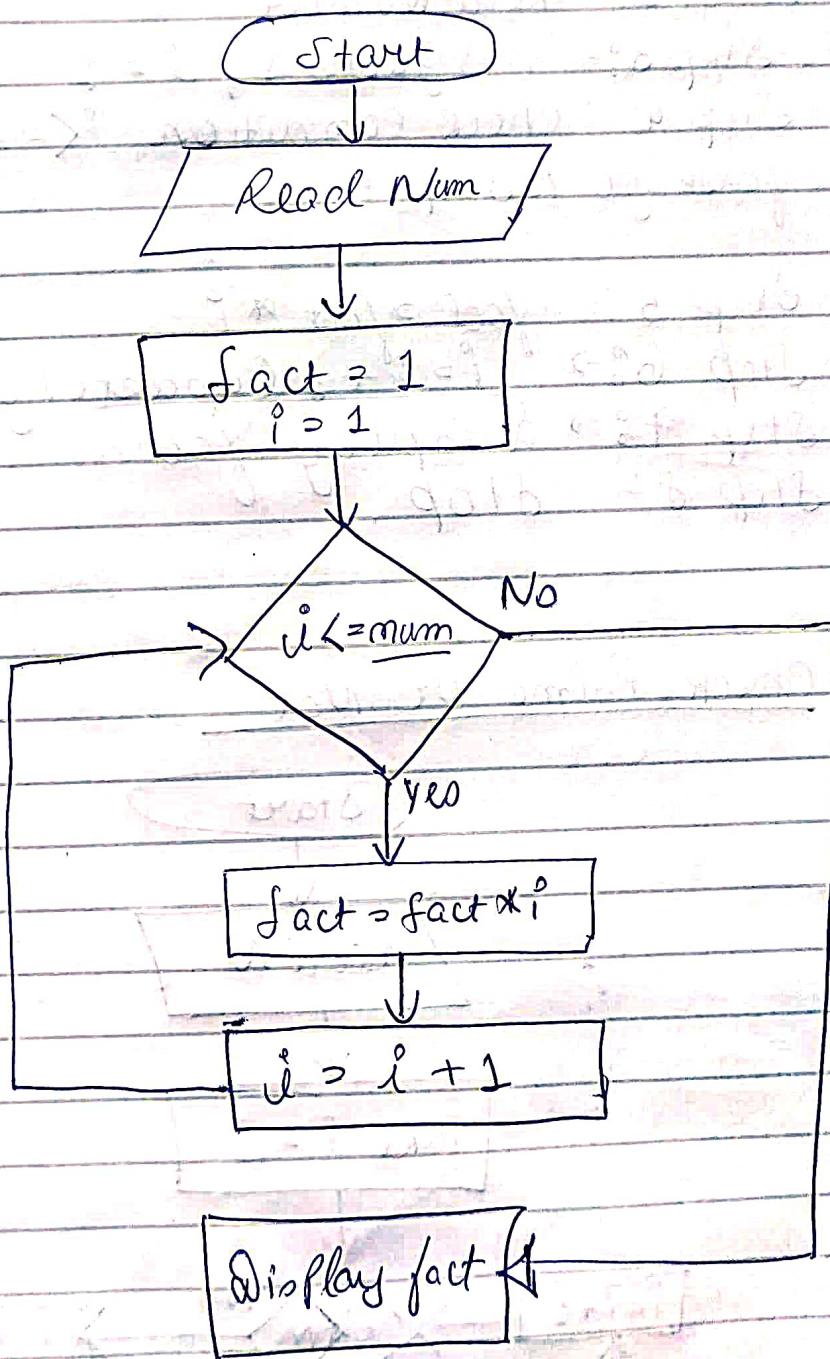
Step 3 :- Calculate $P \times R \times T / 100$
and store remainder to S.I.

Step 4 :- Print S.I.

Step 5 :- Stop

o find factorial of a Number .

Flowchart



N = 5

$$\begin{aligned} \text{fact} &= 1 \times 2 \times 3 \times 4 \times 5 \times 6 \quad \text{fact} = 5! = 120 \\ i &= 1, 2, 3, 4, 5, 6 \end{aligned}$$

$\frac{24}{120} 2$

Algorithm

Step 1 :- Start

Step 2 :- Read Number

Step 3 :- Set fact = 1, i = 1

Step 4 :- Check condition $i \leq \text{number}$ if false go to step 7.

Step 5 :- fact = fact \times i

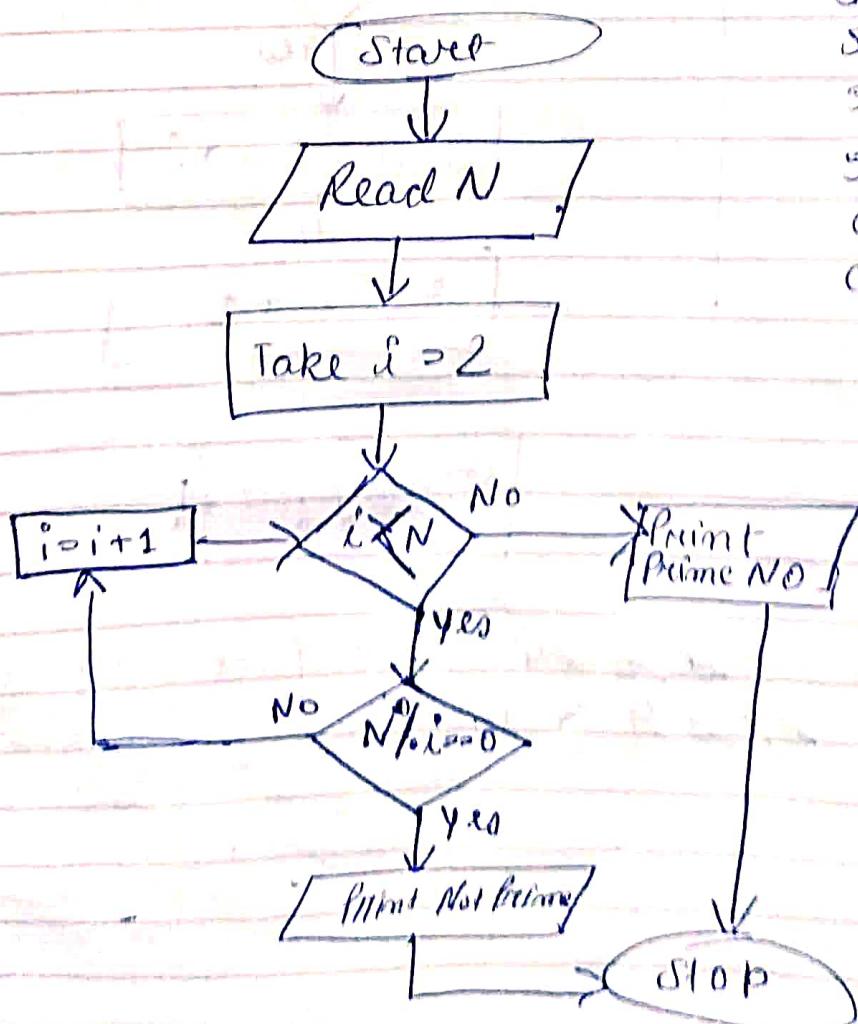
Step 6 :- $i = i + 1$ (update) go to step 4.

Step 7 :- Display fact.

Step 8 :- STOP.

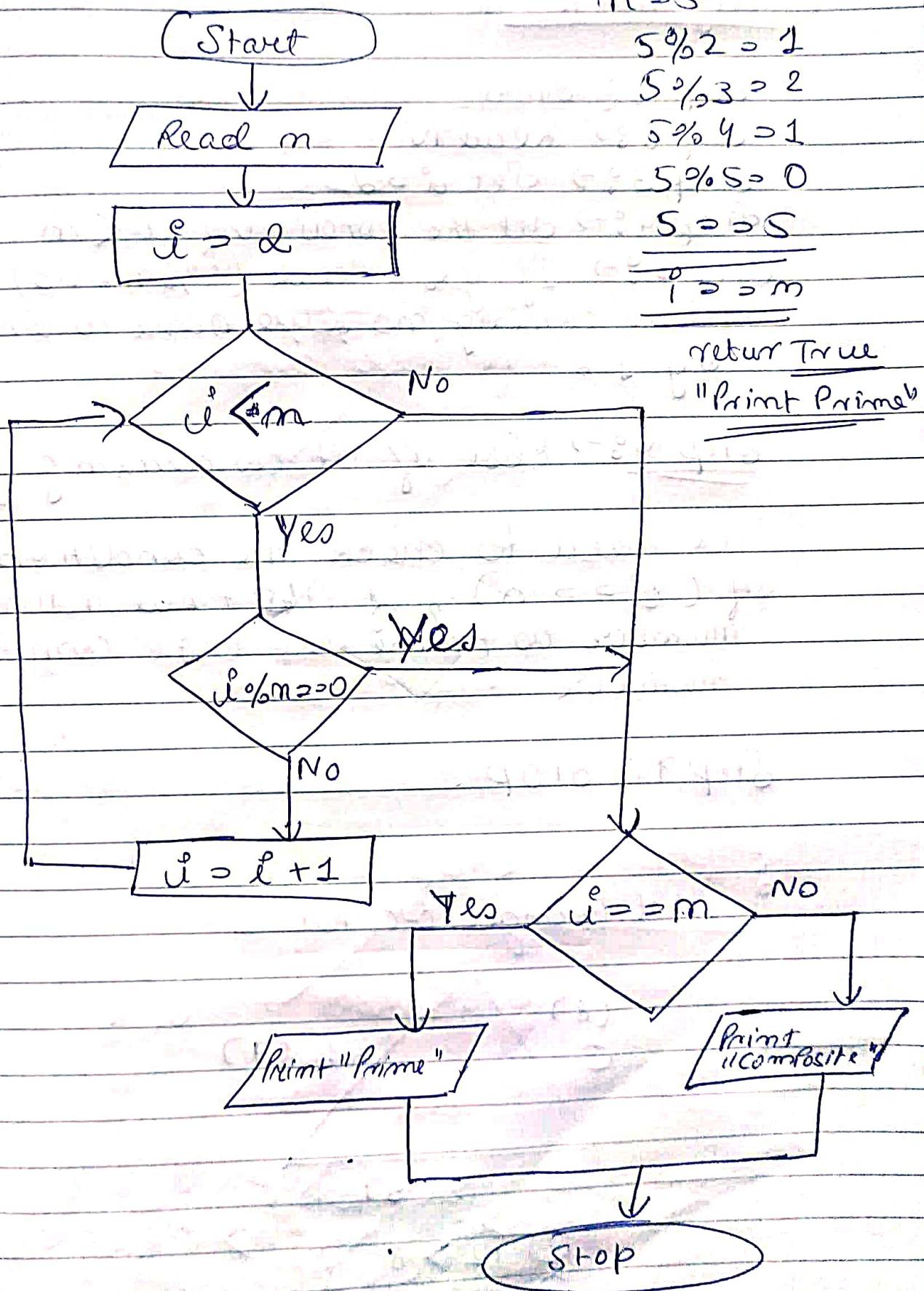
Q

check Prime Or Not



$$\begin{array}{r}
 \textcircled{5} \quad \underline{5} \\
 5/2 = 1 \\
 5/3 = 2 \\
 5/4 = 1 \\
 5/5 = 0, \\
 6/5 = \text{Error} \\
 (\text{overlap})
 \end{array}$$

$$\begin{array}{r}
 \textcircled{1} \quad \underline{\frac{2}{5}} = 1 \\
 \textcircled{2} \quad \underline{\frac{3}{5}} = 1 \\
 \textcircled{3} \quad \underline{\frac{4}{5}} = 1 \\
 \textcircled{4} \quad \underline{\frac{5}{5}} = 1
 \end{array}$$

or

Algorithm

Step 1 : Start

Step 2 : Read N.

Step 3 : Set i = 2

Step 4 : Set the condition $i^2 < n$.

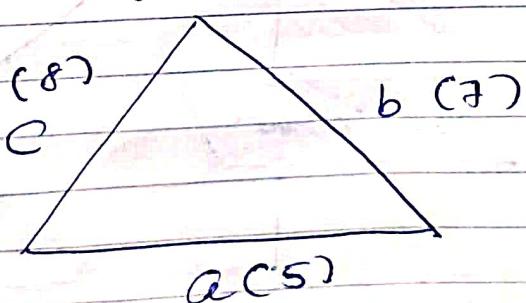
Step 5 : If yes, check ($i^2 \% n == 0$), if it is coming no, we have to increment by 1.

Step 6 : Else if it is coming yes,

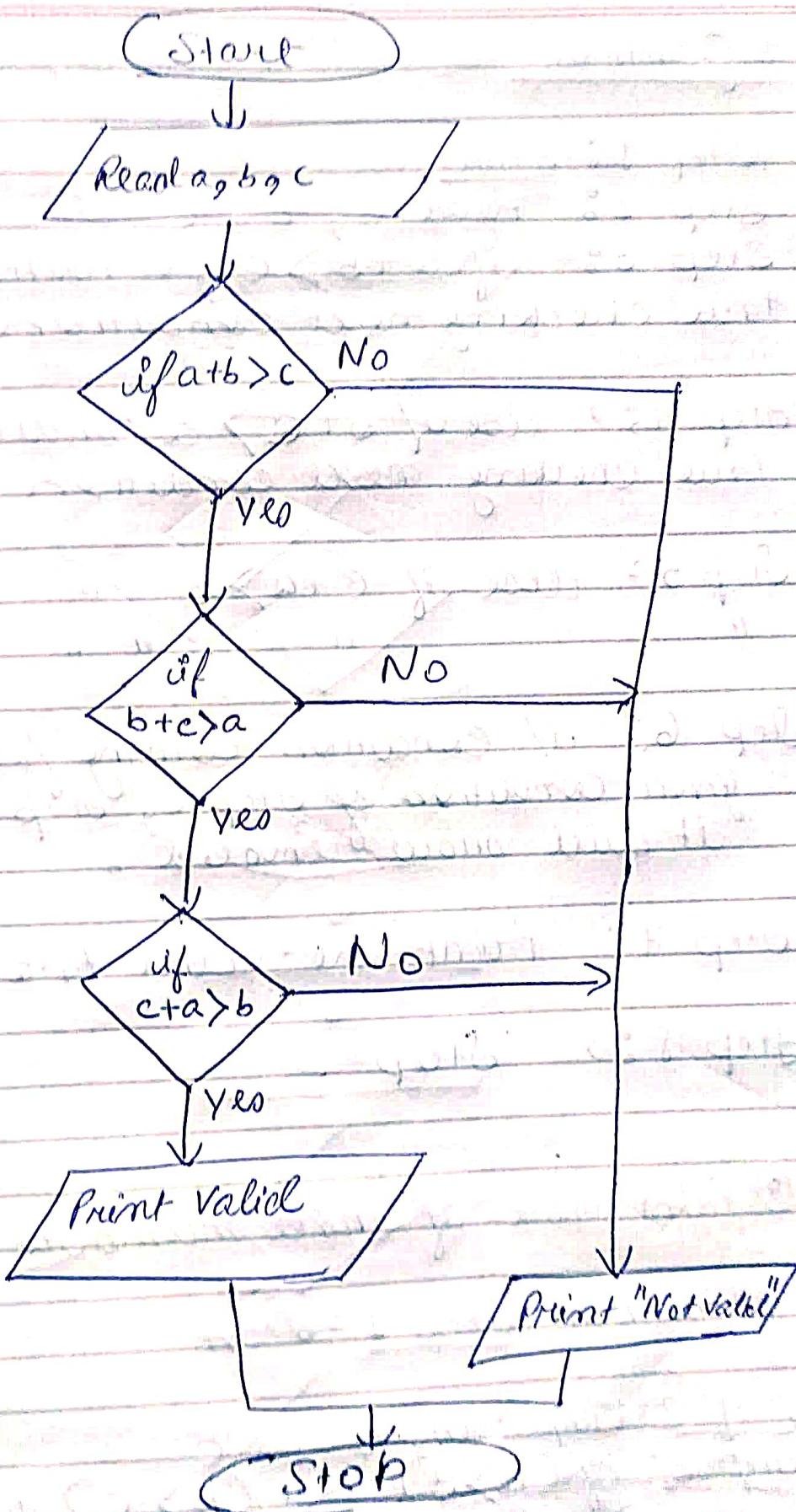
we need to check the conditions of ($i^2 = n$), if it's true return number as prime or else composite number.

Step 7 : Stop.

• Valid Triangle or not



- (i) $12 > 8$
 - (ii) $15 > 5$
 - (iii) $13 > 7$
- } Valid.



Algorithm

Step 1: Start

Step 2: Read a, b, c

Step 3: If $a+b > c$, condition coming true checking next condition.Step 4: Else if $b+c > a$ condition coming true checking next conditionStep 5: Else if $c+a > b$ " "
 " " " "Step 6: If condition coming false any given condition of Step 3, Step 4, Step 5.
It will show invalid.

Step 7: Print the valid triangle

Step 8: Stop.

② Print max of three numbers

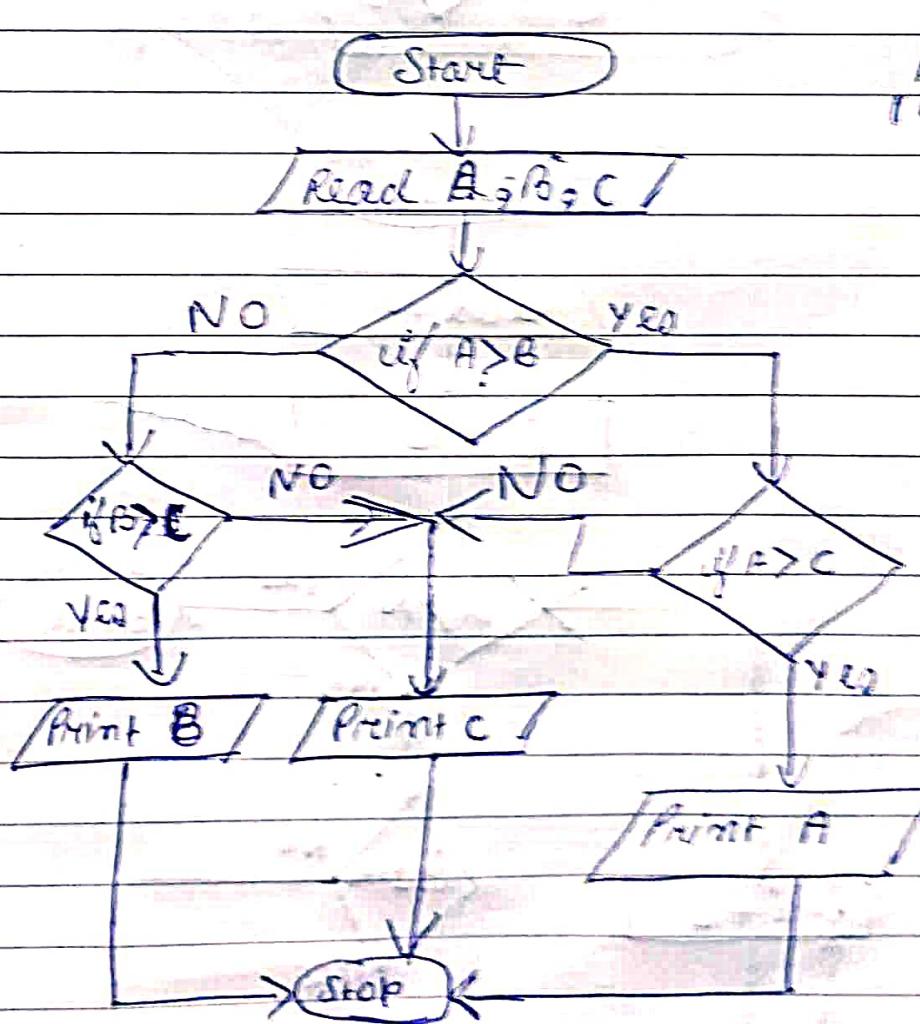
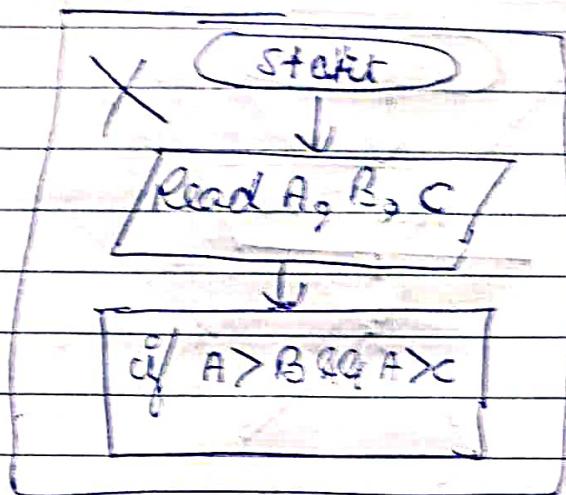
Sol: Algorithm: Step 1: Start

Step 2: Input A, B and C

Step 3: If $(A > B)$ and $(A > C)$ then print "A is greater". Else if $(B > A)$ and $(B > C)$ then print "B is greater", ~~$(C > A)$ and $(C > B)$~~
Else ~~Print~~ Print "C is greater".

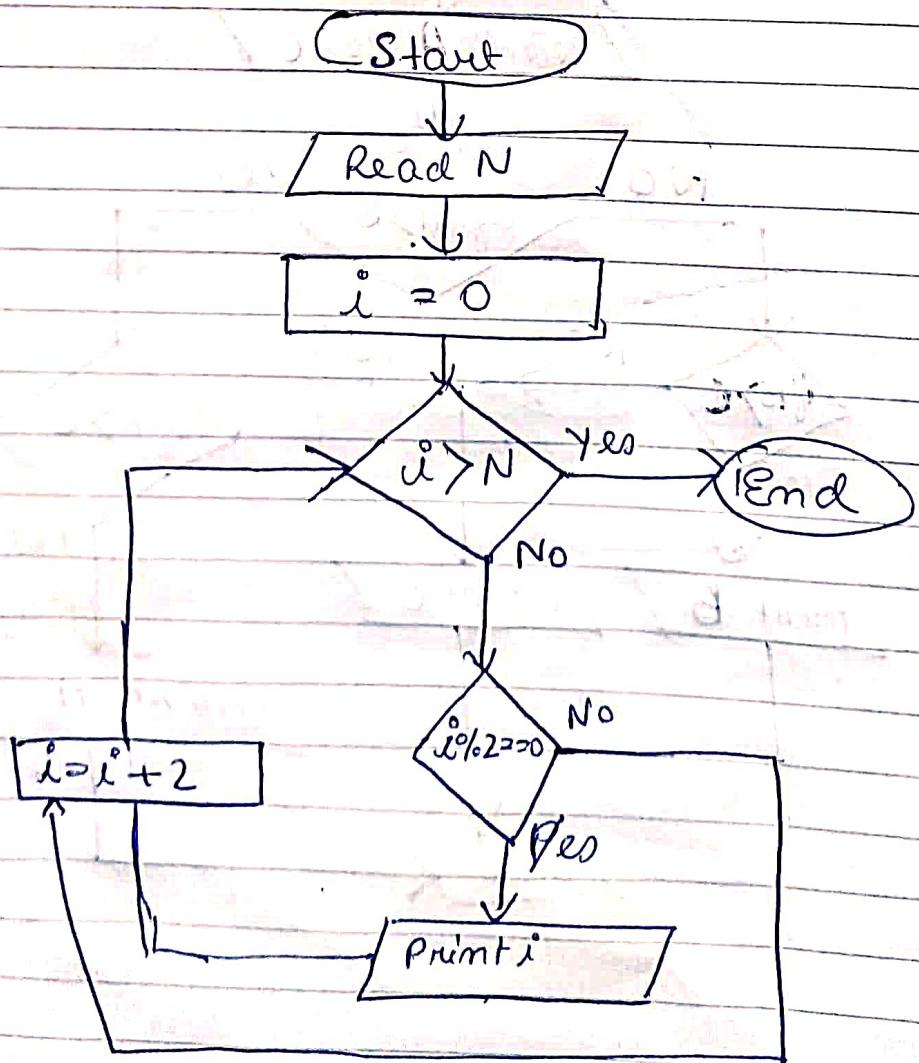
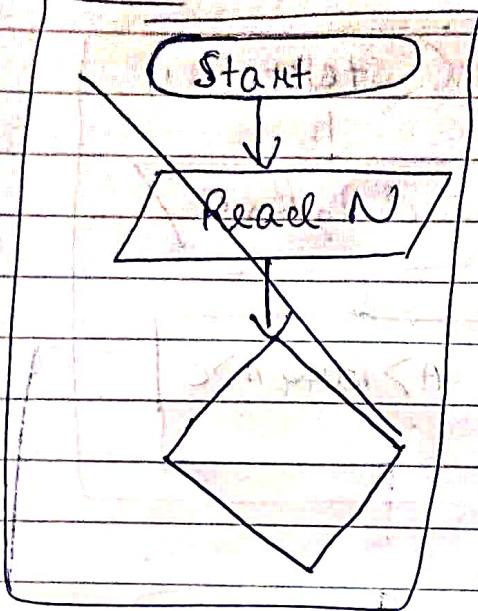
Step 4: \rightarrow stop.

Flowchart



- Print only the even Number till N.

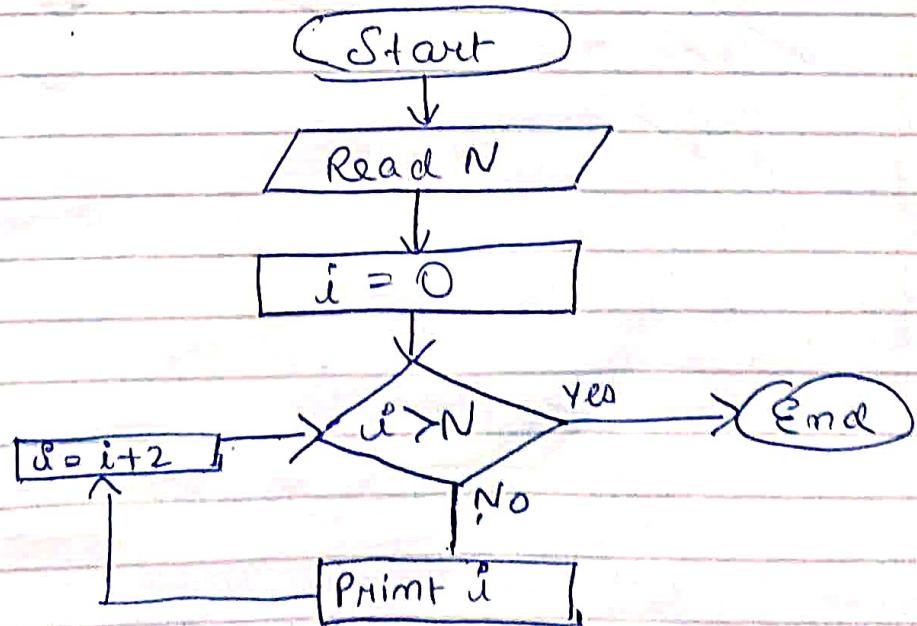
$$N = 6, i - N \Rightarrow 0, 1, 2, 3, 4, 5, 6$$



Algorithm :

- Step 1 : Start .
- Step 2 : Read N
- Step 3 : Take $i = 0$.
- Step 4 : Check condition that $i > N$, if it come a true Exit from Loop . Else it will again condition of ($i \% 2 == 0$)
- Step 5 : From Step 4 checking condition , If it is comes zero it will print i , And then increment by 1 .
- Step 6 : If i will overlap to N program will be stop
- Step 7 : End .

Or (Traditional method)



Date _____
Page _____

Algorithm :-

Step 1: Start

Step 2: Read N

Step 3: Take u=0

Step 4: Check condition
 $U > N$ if it is true
Exit from Loop . If
not Print U.

Step 5: Increment $U+2$,
Iterate the loop again and
again until U overLap
With N.

Step 6 :- Stop.