5] Flowchart to check no. is positive or negative in Java.

```
                    ( Start )
                        |
                        v
                  [ Read int ]
                        |
                        v
                 < IS int > 0 >  ---yes--->  [ Print positive ]
                        |
                        | NO
                        v
                 < is int < 0 >  ---yes--->  [ Print negative ]
                        |
                        | NO
                        v
                 [ Print zero ]
                        |
                        v
                    ( end )
```
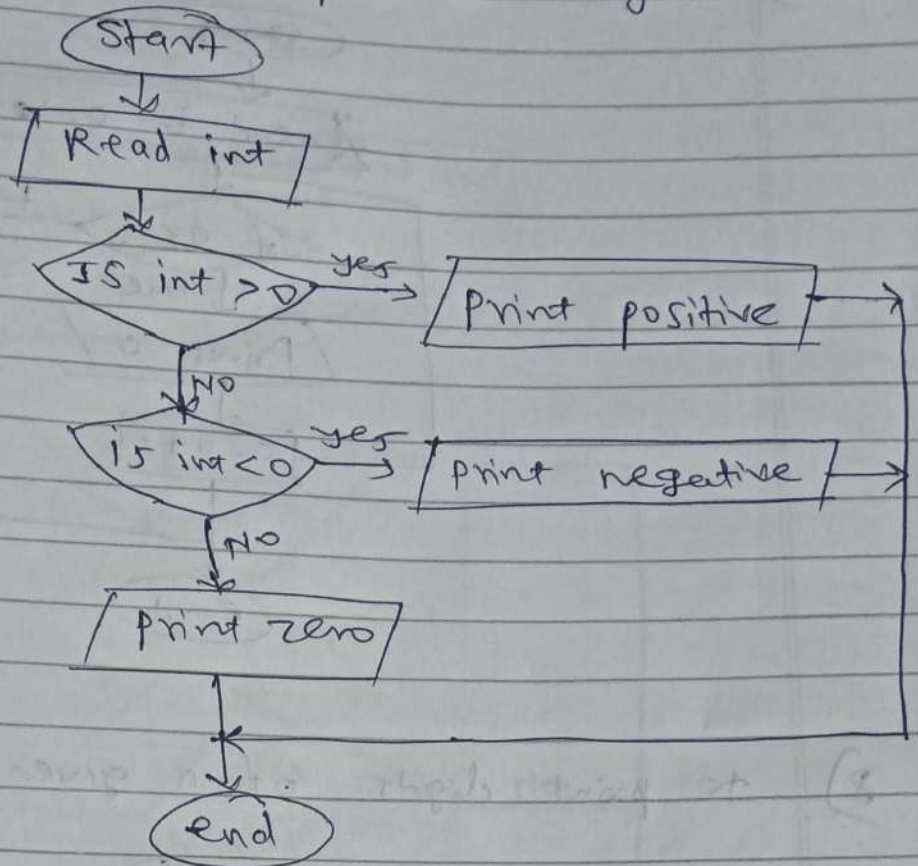
6] Flowchart to check number is leap year or Not.

```
                    ( Start )
                        |
                        v
                [ read input y ]
                        |
                        v
                     < is
                  y % 400 == 0 >  ---yes--->  [ Print - leap year ]
                        |
                        | NO
                        v
                     < IS
                  y % 100 != 0 &&
                     y % 4 == 0 >  ---yes--->
                        |
                        | NO
                        v
              [ Print Not a leap year ]
                        |
                        v
                     ( end )
```
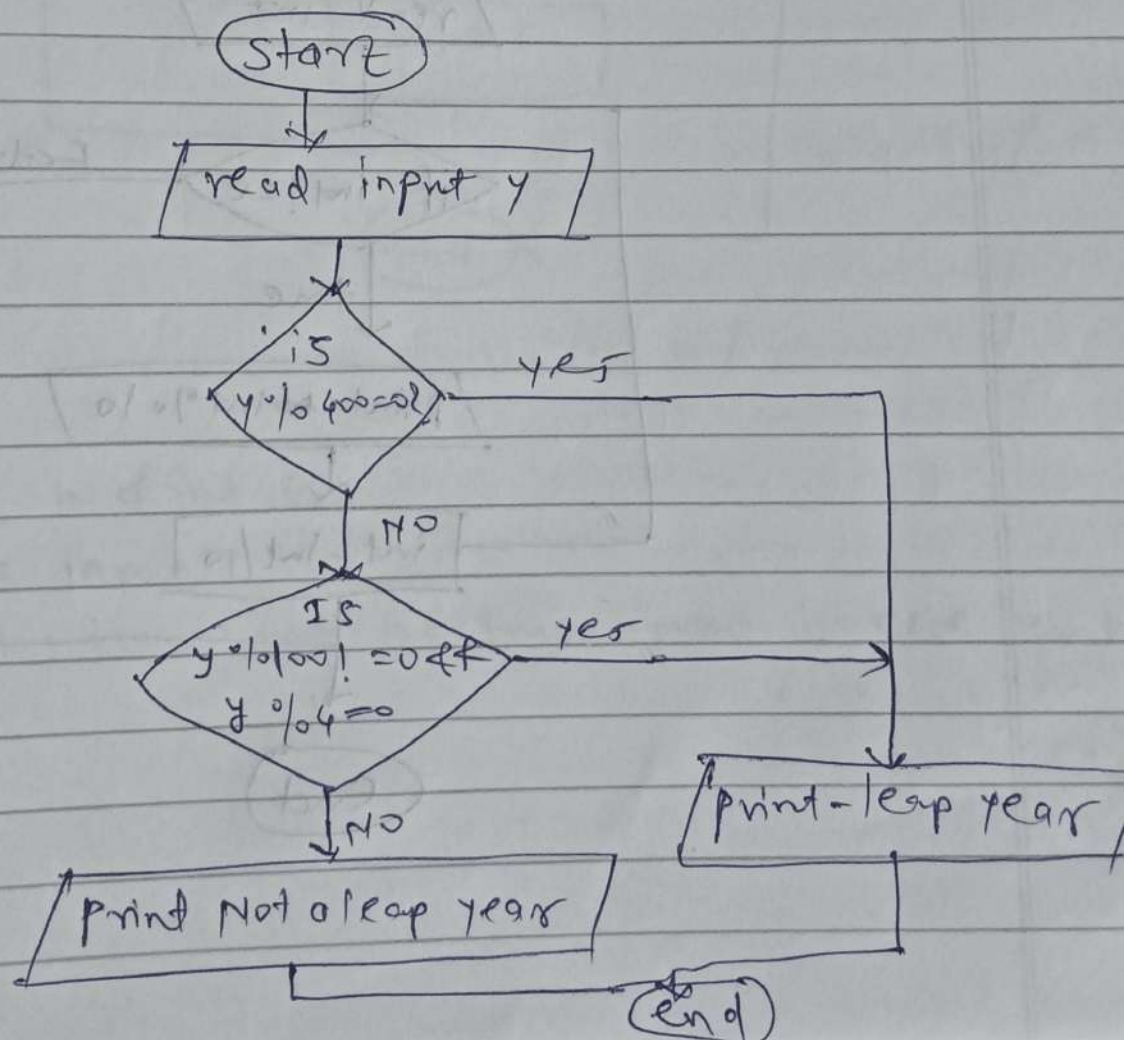
10) Flowchart to find sum of digits of a given n

Start

Enter number

If number >0   NO

yes

n = number %10;
num = num /10;
add = add + n

print sum

end
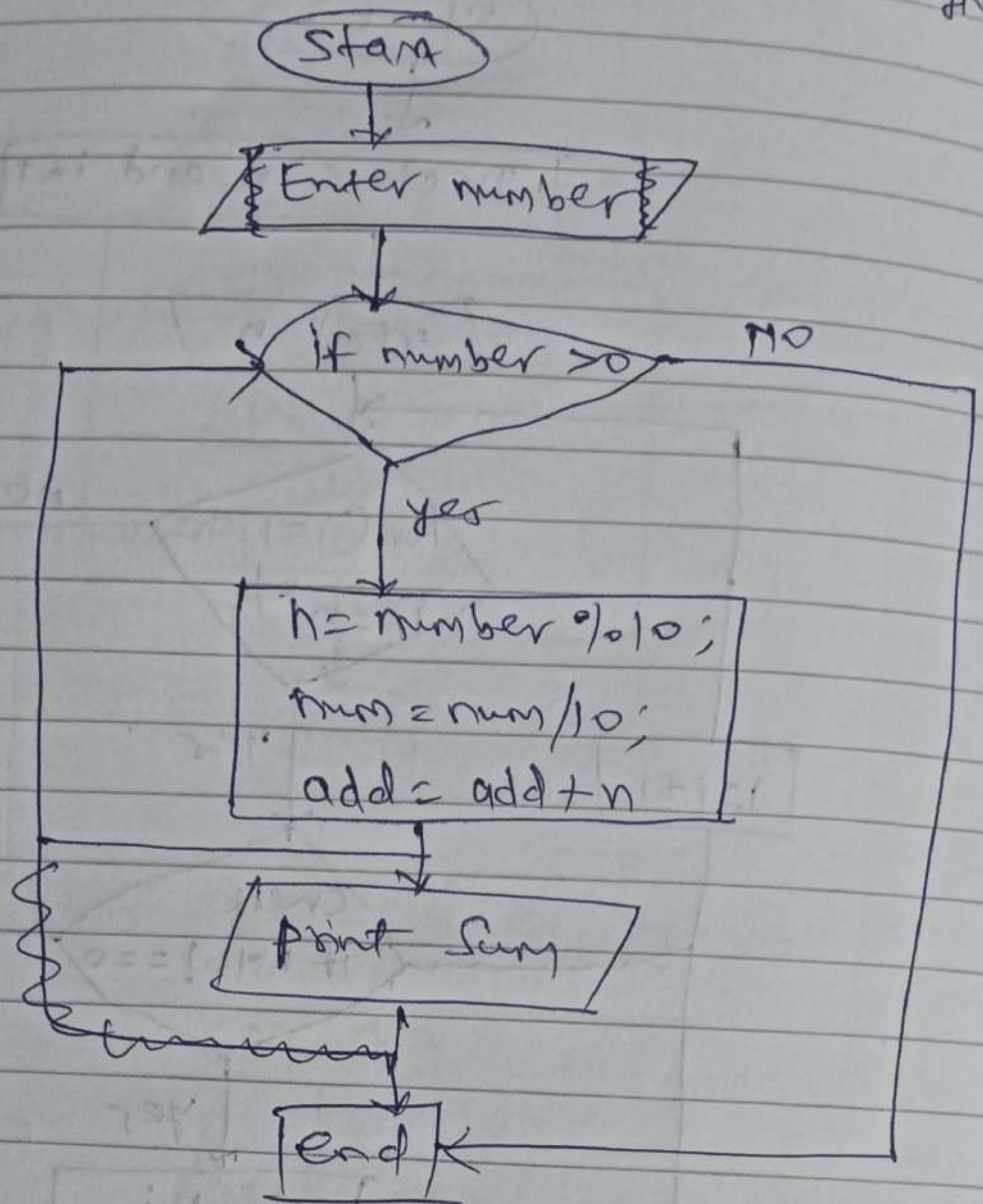
10) Algorithm to find sum of digits of number

1) Start
2) declare sum = 0
3) Get the input
4) if num > 0 is true then num = num %10;
   num = num/10; sum = sum + n repeat until num =0
5) else go to 6
5) print sum
6) Stop

h] Flowchart to find smallest of three numbers.

```
                    ( Start )
                        |
                        v
              [ read int a,b,c ]
                        |
                        v
    yes  <---------- < if a<b > ----------> No
     |                                          |
     v                                          v
 <  if  >                                    <  if  >
yes  a<c  NO                             yes  b<c   No
 |        |                               |          |
 v        v                               v          v
/print a is/  /print c is the/    /b is the /    /c is the/
/the smallest/ /smallest     /    /smallest /    /smallest/
/number      /                    /         /    /        /
     |            |                    |            |
     |            +--------------------+------------+
     |                      |
     +----------------------+
                            |
                            v
                        ( end )
```
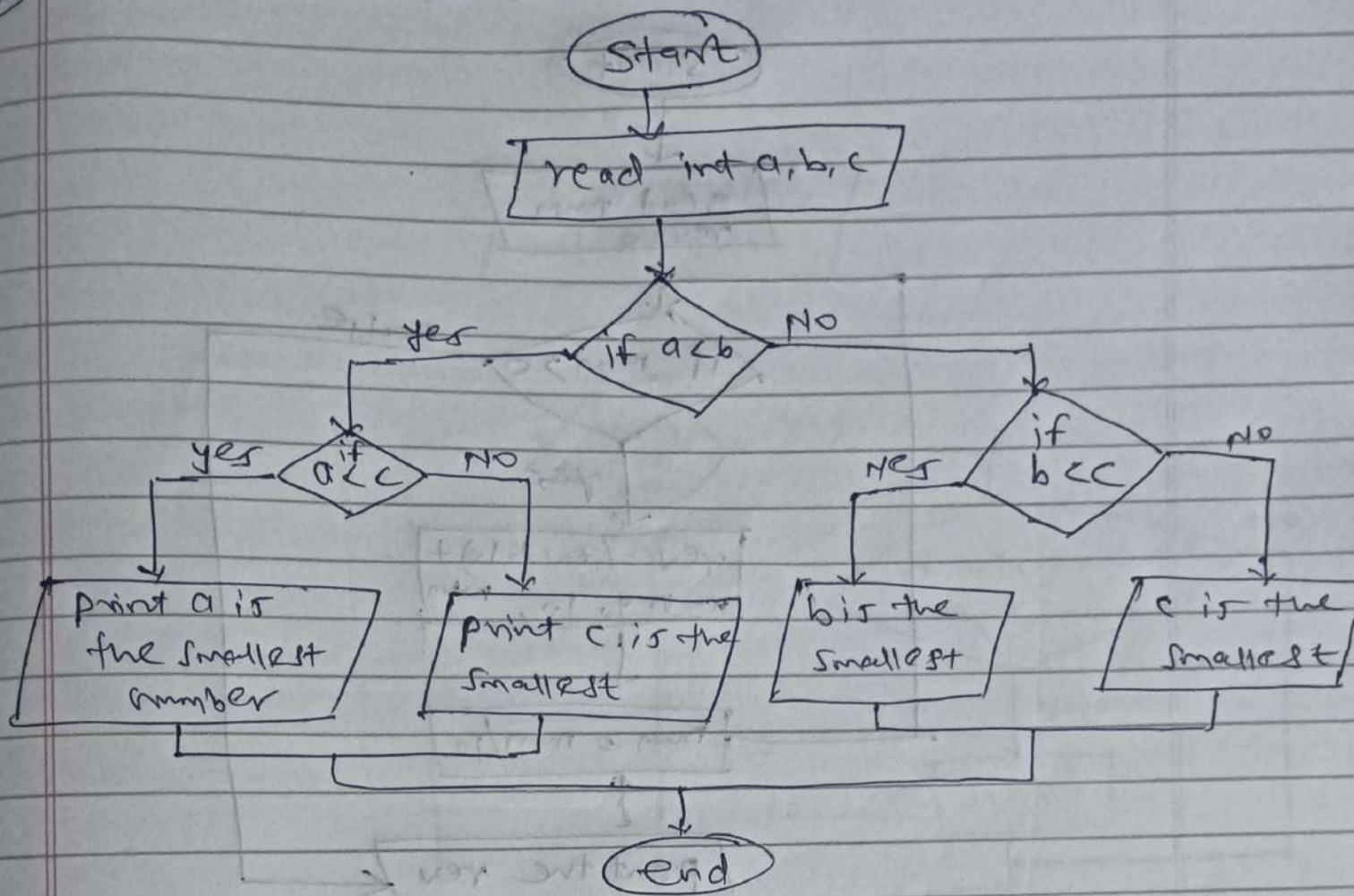
II] Algorithm to find smallest of three numbers.

1) start

2) get input a, b, c

3) if a < b is true go to ④ else go to ⑤

4) check for a<c if it is true print a is the smallest number else print 'c' is the smallest number

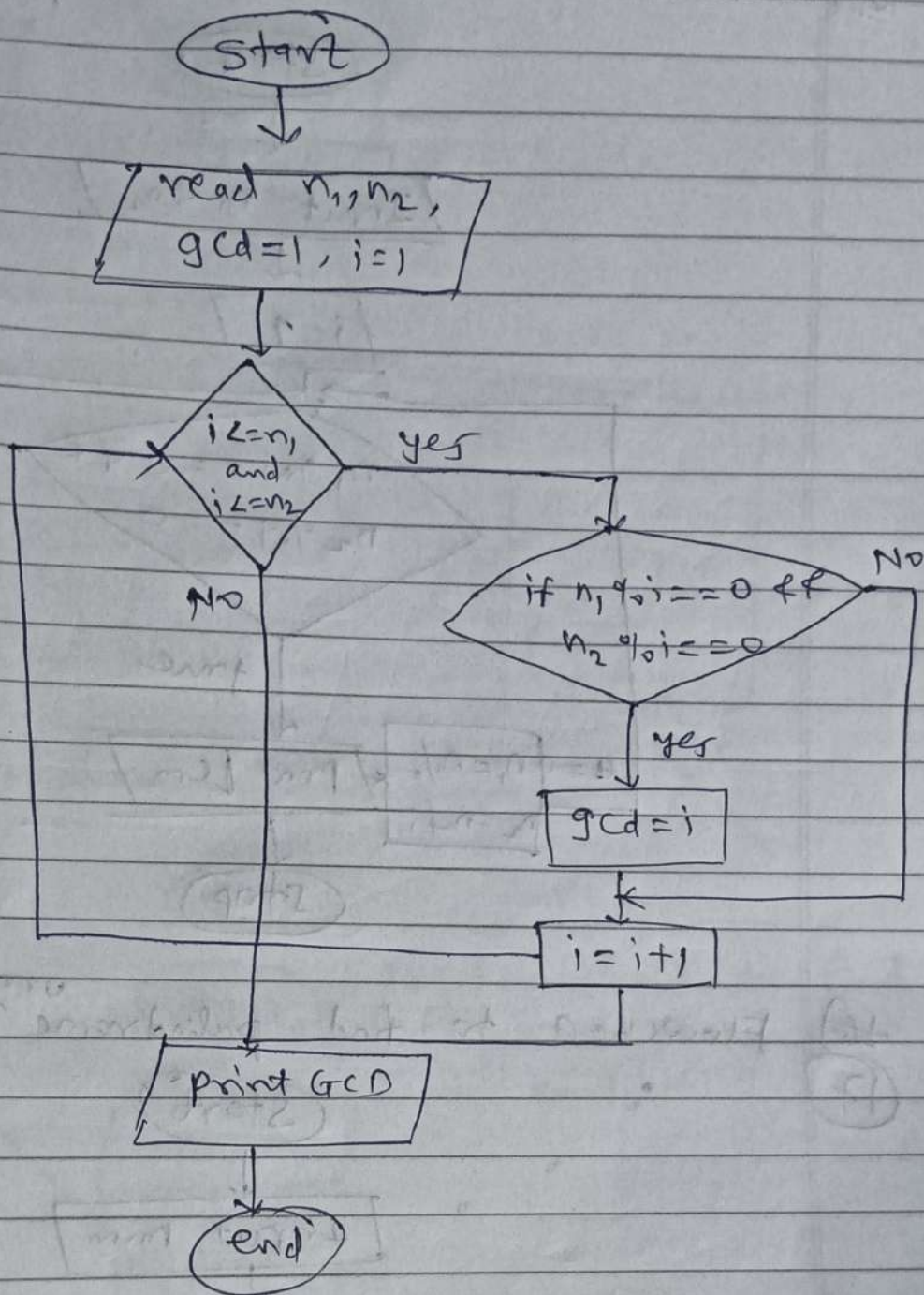5) check b<c if it is true b is the smallest number else 'c' is the smallest number.

6) stop

13) Flowchart to reverse a given number

```
                  ( Start )
                      |
                      v
            +-------------------+
            |   read num        |
            |   rev = 0         |
            +-------------------+
                      |
                      v
                 /    if    \        False
               <  num > 0     >------------+
                 \          /              |
                      |                    |
                      | yes                |
                      v                    |
            +-------------------+          |
            | rev = (rev × 10) +|          |
            | n = n % 10;   num |          |
            +-------------------+          |
                      |                    |
                      v                    |
            +-------------------+          |
            |   num = num / 10  |          |
            +-------------------+          |
                      ^                    |
                      |                    |
            +-------------------+          |
            |  print the rev    |<---------+
            +-------------------+
                      |
                      v
                   ( end )
```
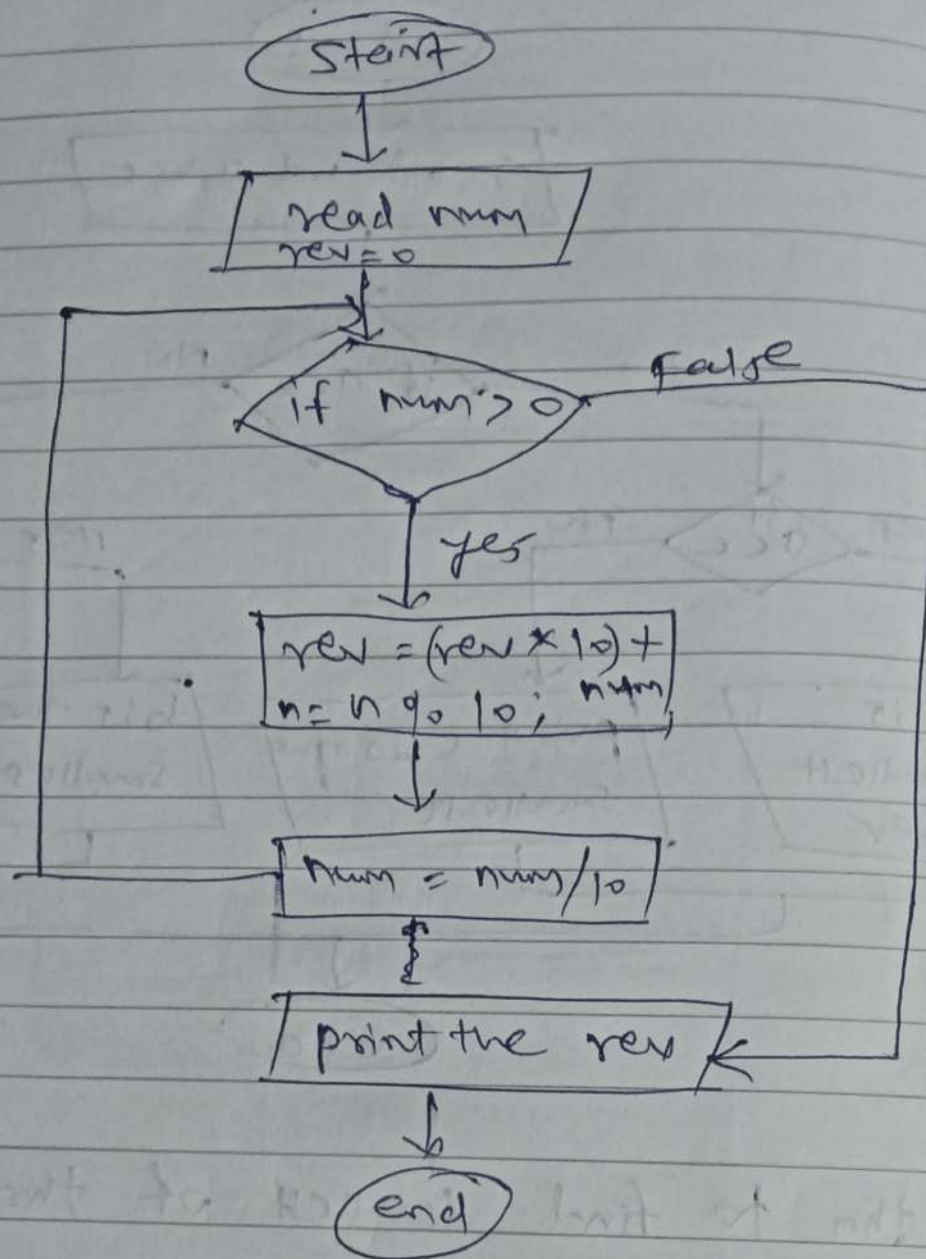
13) Algorithm to find the reverse of a given

1) Start
2) declare rev = 0
3) get the input
4) if num > 0 is true   rev = (rev x 10) + num
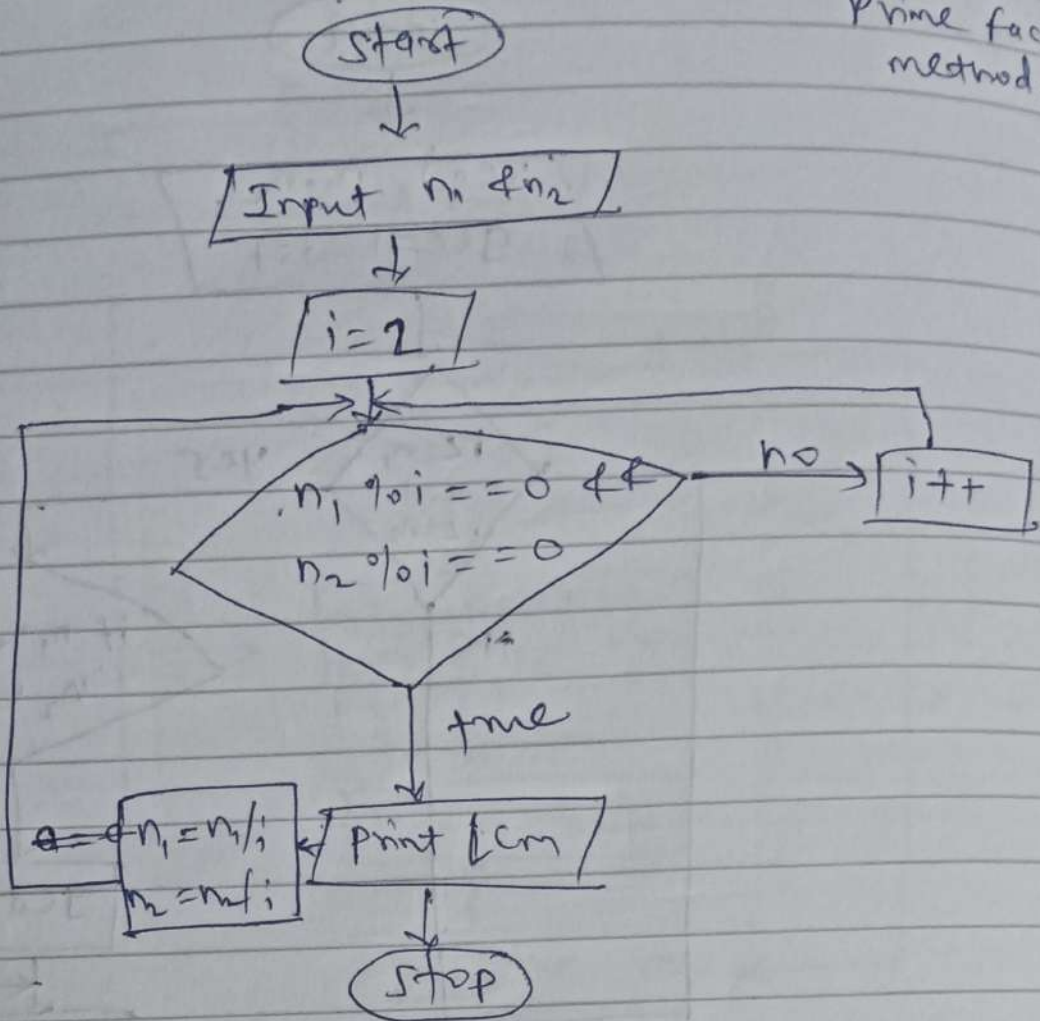   n = n % 10   else go to ⑥
5) num = num/10
6) print the num
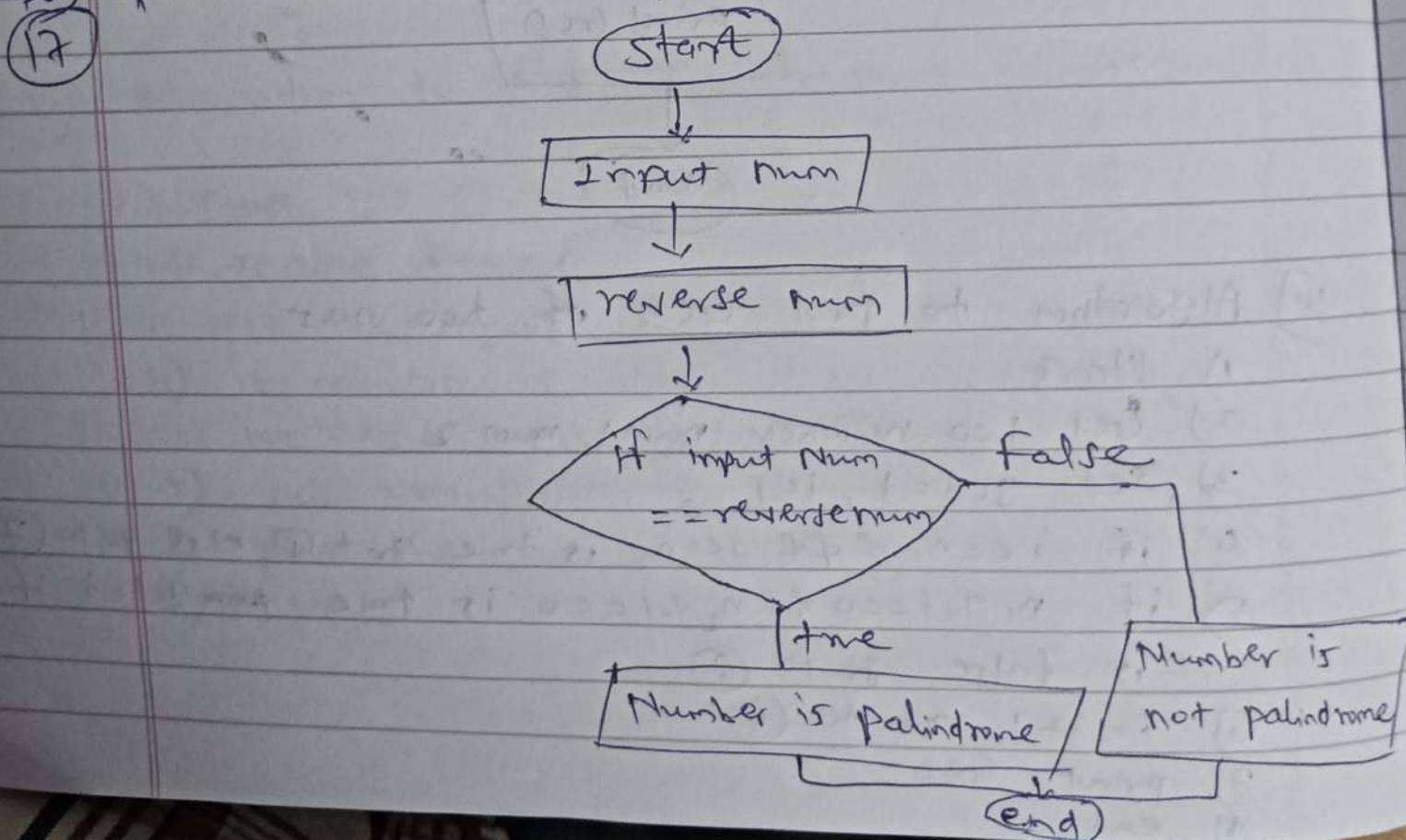7) Stop

14) Flowchart to find GCD of too given numbers.

```
                    ( start )
                        |
                        v
              / read n₁, n₂, /
             /  gcd=1, i=1  /
                        |
                        v
                  /i<=n₁\        yes
                 < and    >--------------+
                  \i<=n₂/                |
                        |                v
                       No          / if n₁%i==0 &&  \    No
                        |         <   n₂%i==0        >-----+
                        |          \                /      |
                        |                |                 |
                        |               yes                |
                        |                v                 |
                        |             [ gcd=i ]            |
                        |                |                 |
                        |                v<----------------+
                        |            [ i=i+1 ]
                        |                |
                        v
                 / print GCD /
                        |
                        v
                    ( end )
```

14) Algorithm to find GCD of two no's

1) Start
2) Get two number num 1, num 2
3) Set gcd=1, i=1
4) if i<=n₁ & && i<=n₂ is true go to ⑤ else go to ⑦
5) if n₁%i==0 && n₂%i==0 is true print gcd=i if
   it is false go to ⑥
6) i=i+1 go to ④
7) print GCD
8) end

16) Flowchart for LCM of two positive integers. using Prime factors method.

Start

↓

Input n_1 & n_2

↓

$i = 2$

↓

$n_1 \% i == 0$ && $n_2 \% i == 0$ → no → $i++$

↓ true

$n_1 = n_1/i$; $n_2 = n_2/i$ ← Print LCM

↓

Stop

16)
17) Flowchart to find palindrome or not of a number

Start

↓

Input num

↓

reverse num

↓

If input Num == reverse num → false → Number is not palindrome

↓ true

Number is palindrome

↓

end

Flowchart to print even number series.

[a]

```
                    ( Start )
                        |
                        v
                  [ Num = 2 ]
                        |
                        v
             / read n (no. of terms) /
                        |
                        v
            +---------->
            |     < is i <= n >------ No ----+
            |           |                    |
            |          yes                   |
            |           v                    |
            |    / Print Num /               |
            |           |                    |
            |           v                    |
            +--- [ Num = Num + 2 ]           |
                        |                    |
                        v <------------------+
                    ( end )
```
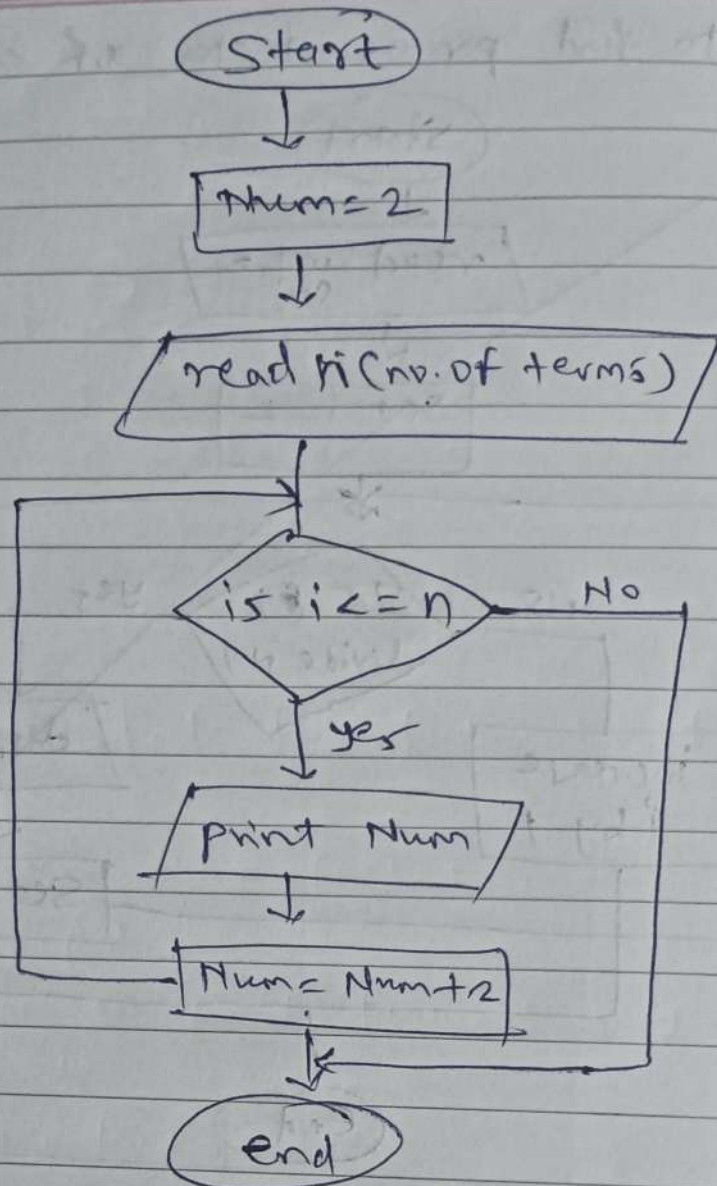
[b] Algorithm to print even series

1) Start
2) Num = 2
3) read the input
4) is i <= n is true go to ⑤ else go to ⑦
5) print Num
6) Num = Num + 2 go to ④
7) end

7] to print 1 - 10 without using loop.

```
                    ( Start )
                        |
                        v
          $ read int a=1  $
                        |
                        v
              if a<10  ---------> false
                        |
                        | true
                        v
                  / Print a /
                        |
                        v
                  | a = a+1 |
                        |
                        v
                    ( end )
```
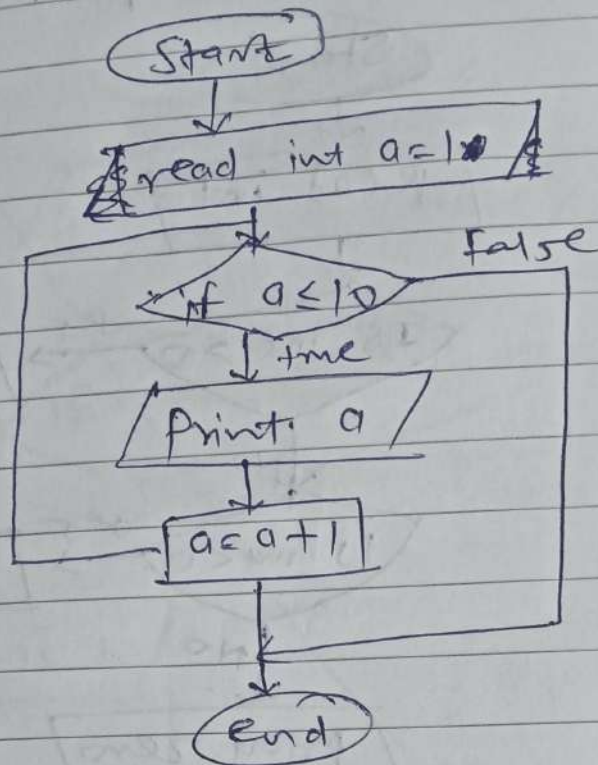
8] to print digits of a given number

```
                    ( Start )
                        |
                        v
              / read int /
                        |
                        v
              if int != 0  ---------> False
                        |
                        | true
                        v
              / Print int % 10 /
                        |
                        v
              | int = int/10 |
                        |
                        v
                    ( end )
```
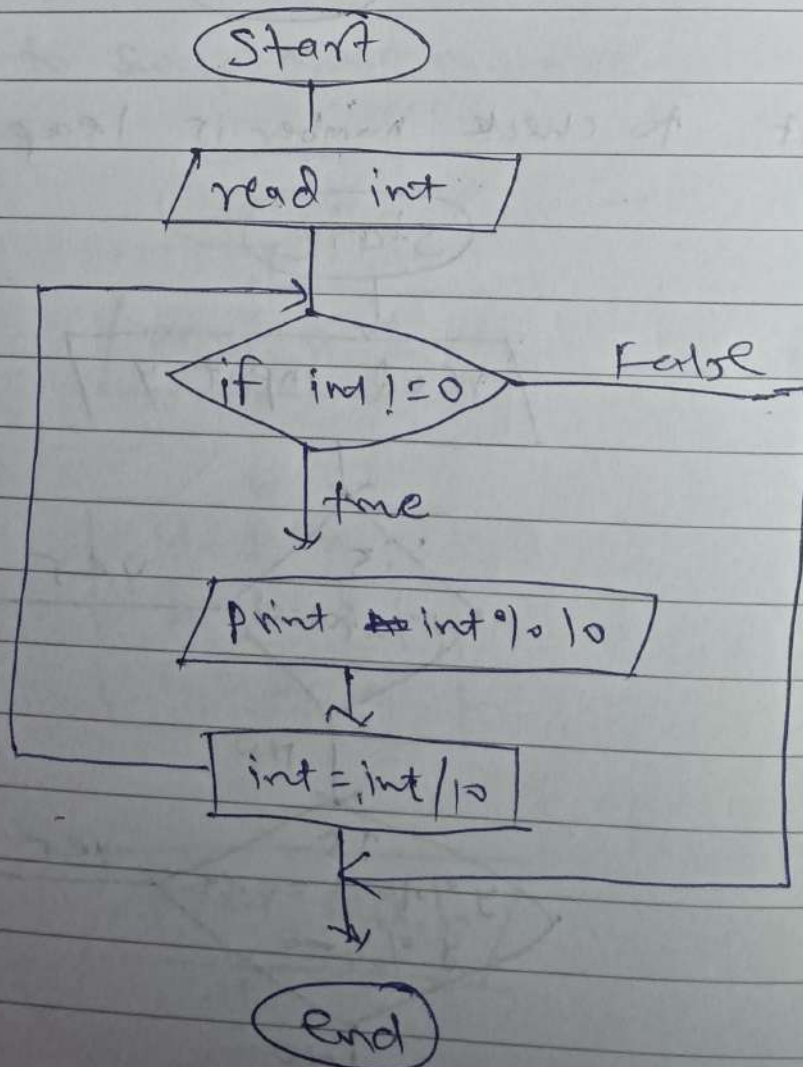
1] Flowchart to check given number is even or odd.

Start

read int

if int %2 ==0

false → Number is odd

true

Number is even

end

2] Factorial → Flowchart →

Start

Set i = 1 and fact = 1

Get the number

i <= num

false

true

fact = fact × i
i = i+1
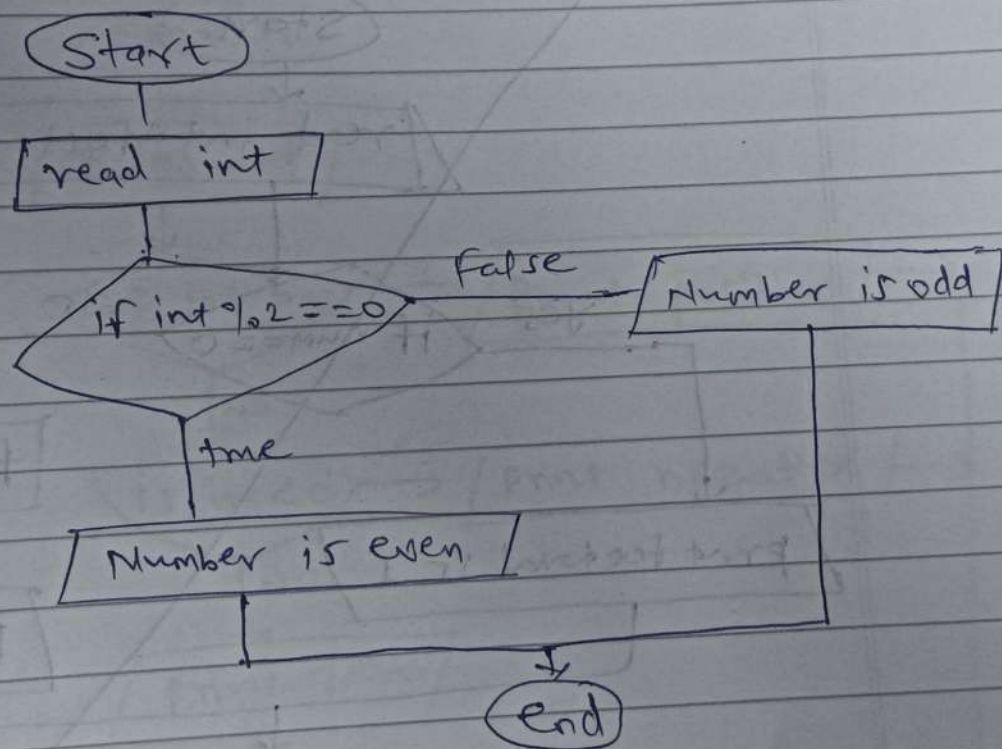
print factorial
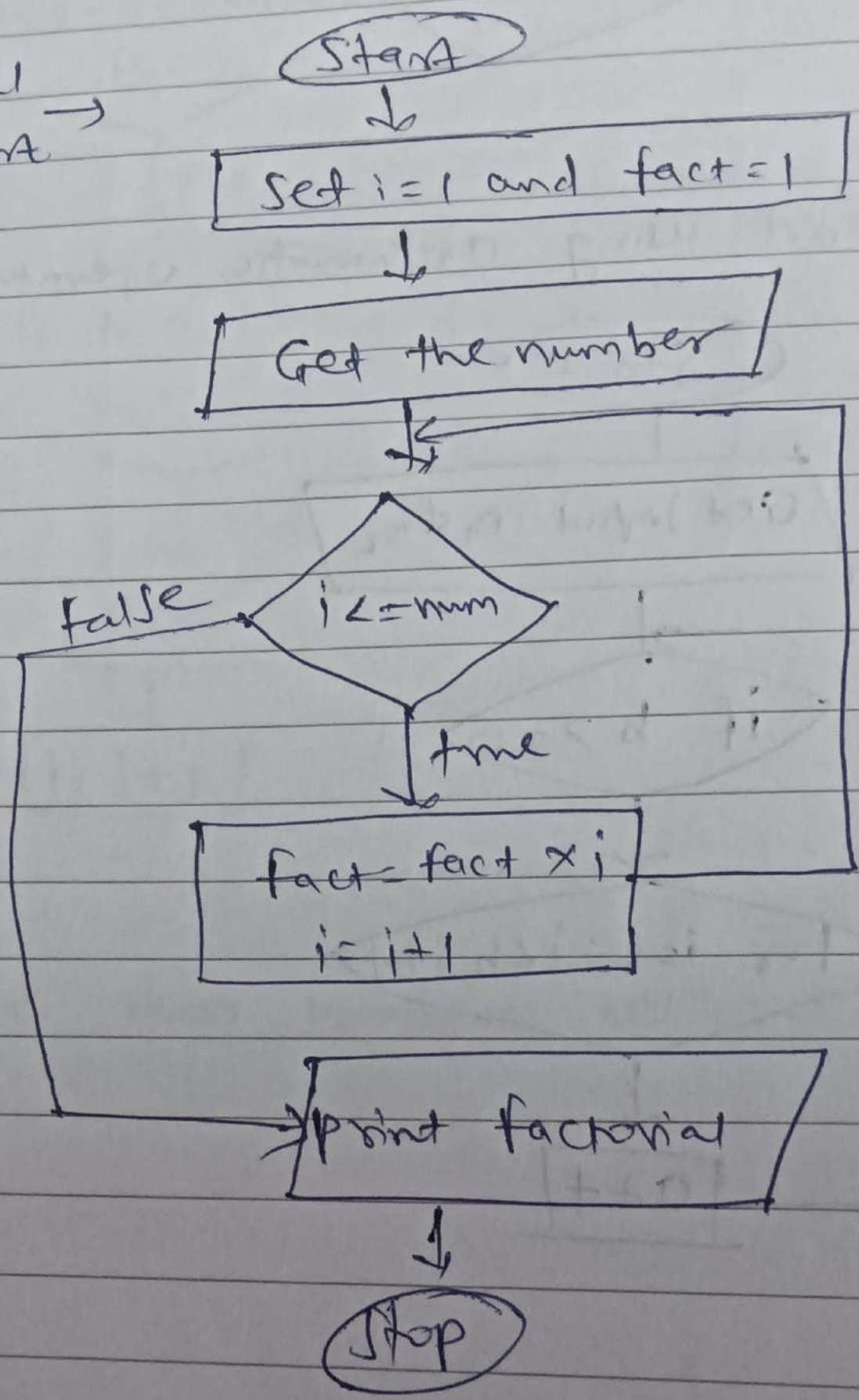
Stop

4] Flowchart to swap two numbers without using third variable

$$\text{Start}$$

$$\text{read } a, b$$

$$a = a + b$$
$$b = a - b$$
$$a = a - b$$

$$\text{Print } a, b$$

$$\text{end}$$
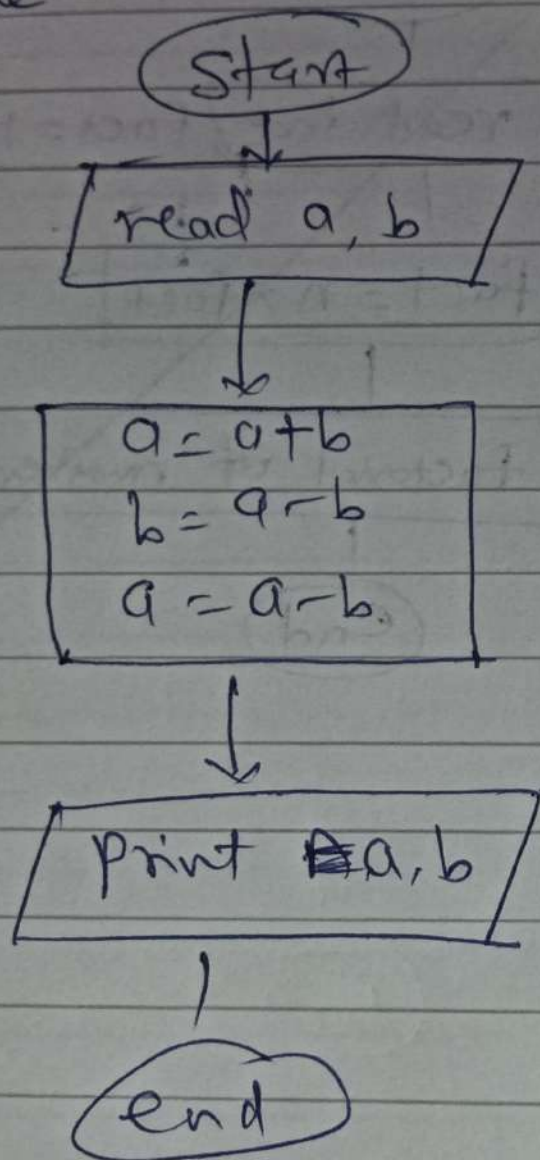
9) Flowchart for factors of a number →

Start

Declare n and int

read n

for (i=1 to i<=n) *½* → NO

yes

Check
if n%i0! ==0 → No

yes

print i

i=i+1

end
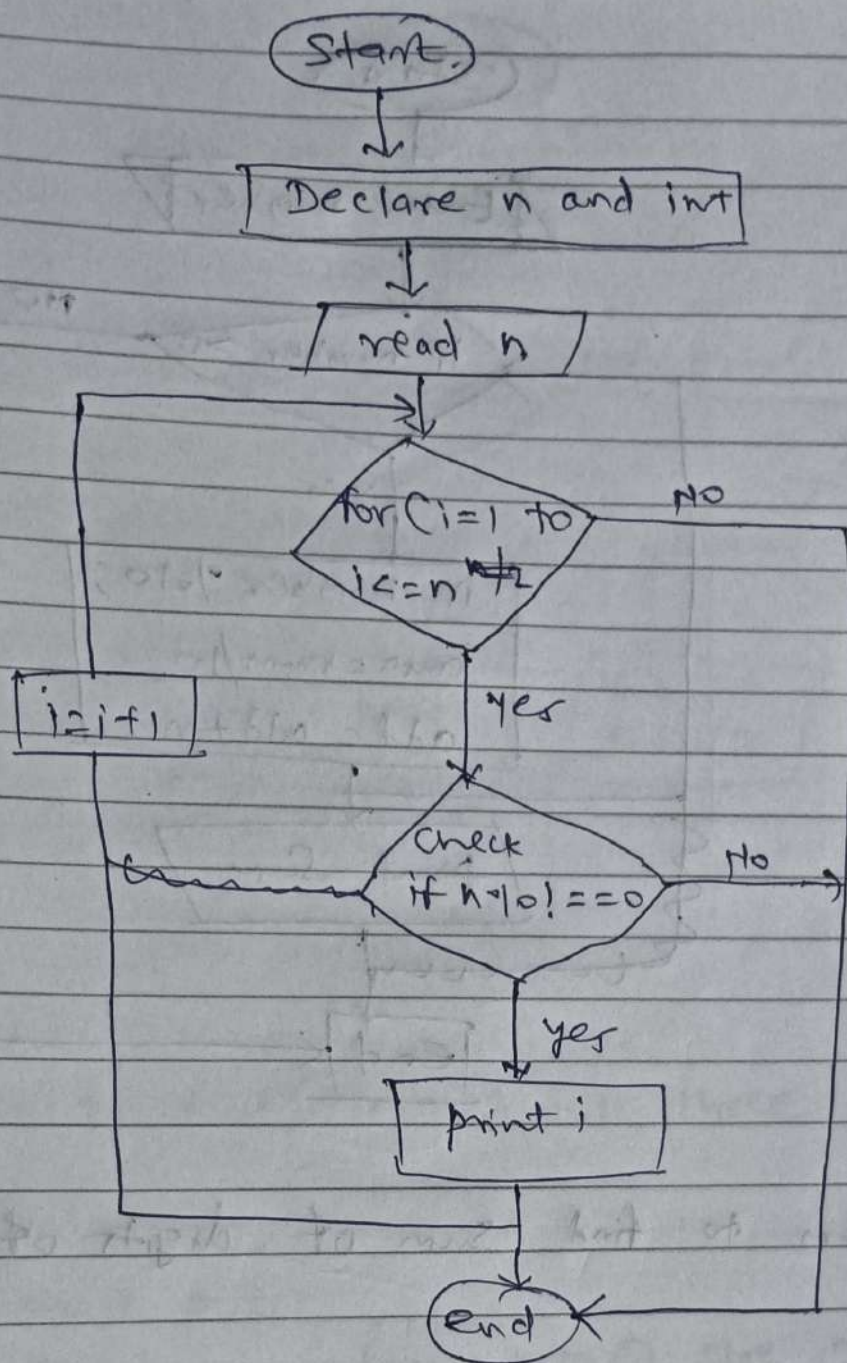
9) Algorithm
1) Start
2) declare n & int i=1
3) get the input
4) for (i=1 ; i<=n ; i++) if true print i else go to ⑥
5) i=i+1
6) Stop

18) Flowchart to find prime factor of a number

```
                    ( Start )
                        |
                        v
                  / read  a /
                        |
                        v
                 [ set i=2 ]
                        |
                        v
            /is a % i == 0\------> False
            \              /         |
                |                    |
                | true               |
                v                    |
            / print i /              |
                |                    |
                v                    |
            [ a = a/i                |
              i = i++ ]              |
                |                    |
                v<-------------------+
             ( end )
```
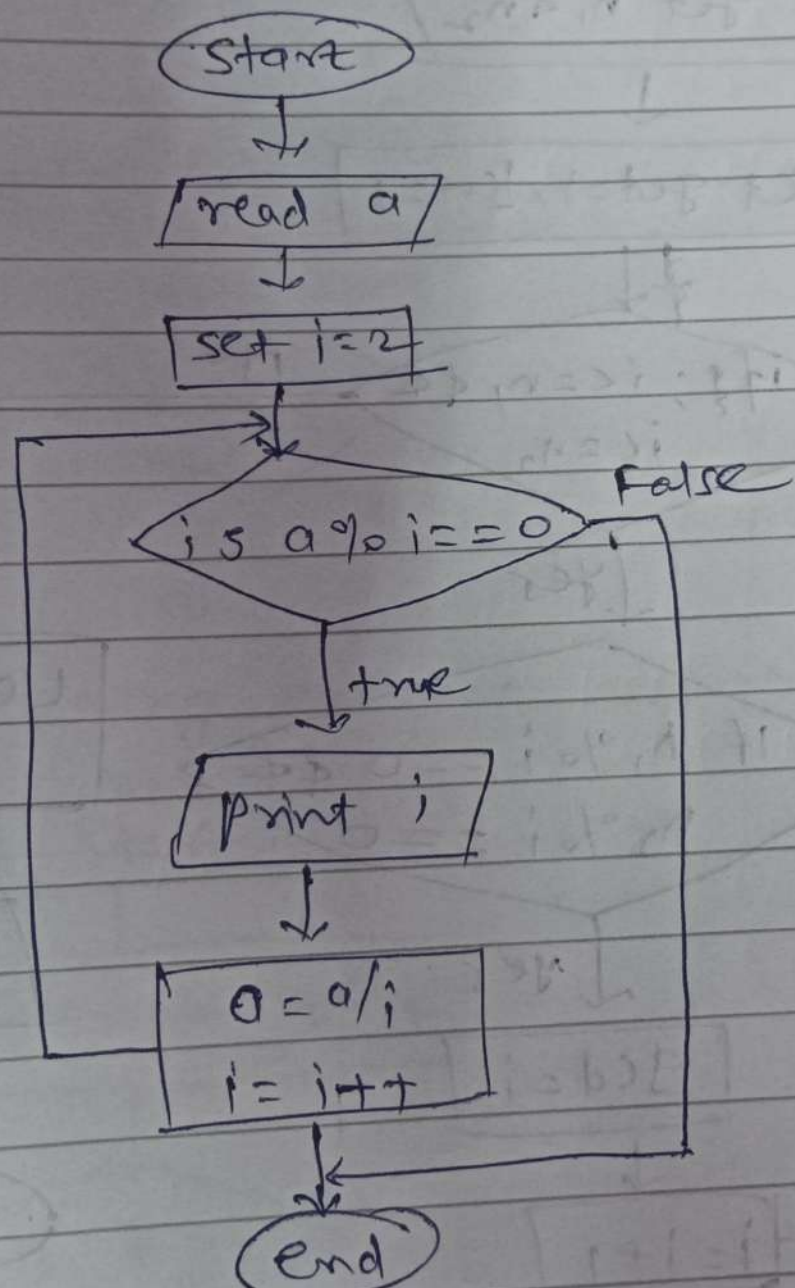
(11) Flowchart to find Lcm of two numbers

Start

get n_1 & n_2

set gcd=1, Lcm=1

if { ; i <= n_1 && i <= n_2

No

yes

if n_1 % i == 0 &&
n_2 % i == 0

No

yes

gcd = i

i = i + 1

$$Lcm = \frac{n_1 \times n_2}{gcd}$$
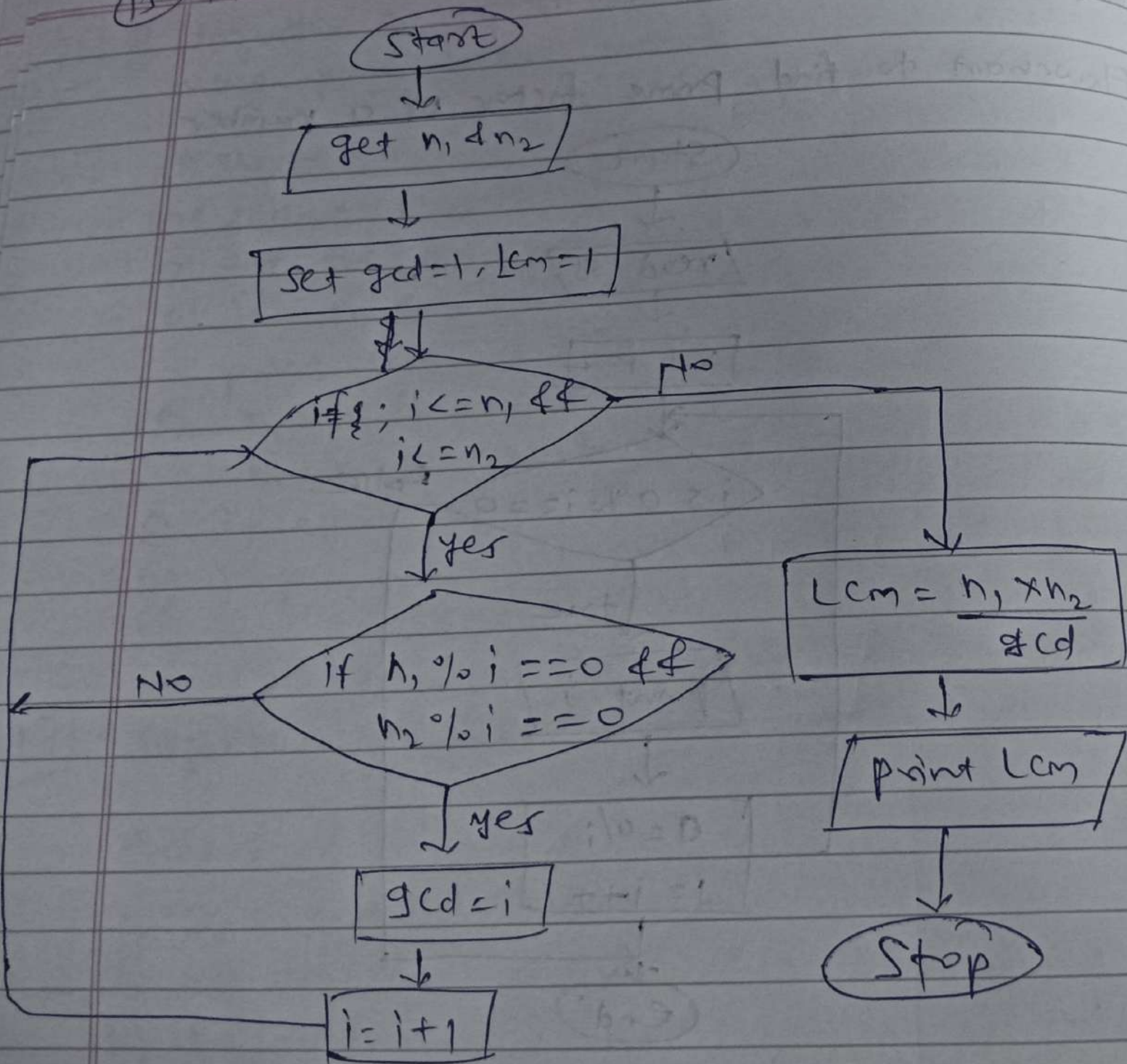
print Lcm

Stop

16). Algorithm to find Lcm of two positive integers using prime factor method.

1) Start
2) declare i=2
3) get the input n_1 & n_2
4) if $n_1 \% i == 0$ && $n_2 \% i == 0$ is true print Lcm if false go to ⑥
5) $n_1 = n_1/i$
   $n_2 = n_2/i$ go to step ④
6) i=i+1 go to step ④
7) Stop

17] Algorithm to find the given number is palindrome or not

1) Start
2) Input num
3) reverse the number
4) if input num == reverse number is true print palindrome number else print Not a palindrome number.

18) Algorithm to find prime factor of a number →

1) Start
2) Input num a
3) set i=2
4) if a % i == 0 is true display i else and go to ⑤ else go to ⑥ ⑦
5) a = a/i go to ④
6) i=i+1 go to ④
7) end

5] Algorithm to check Number is positive or negative.
  1) Start
  2) Get the input number
  3) If number is greater than zero print positive Number
  4) if Number is less than zero print Negative Number
  5) if ④ & ⑤ are false the print number is zero
  6) Stop

6) Algorithm for leap year
  1) Start
  2) Get the input year 'y'
  3) if year is y % 400 = 0 then print 'leap year'
  4) else check for y % 100 ! = 0 && y % 4 = 0
  5) if ④ is true print leap year else print Not a leap year
  6) stop.

7] Algorithm to print 1 to 10
  1) Start
  2) declare a = 1
  3) if a ≤ 10 then print a
  4) ~~repeat 3~~ a = a + 1
  5) repeat ③ ~~until~~ until it becomes false
  6) Stop

8) Algorithm to print digits of a number
  1) Start
  2) Get the input 'i'
  3) if i ! = 0 then print i % 10 else go to ⑤
  4) int = int / 10 repeat ③
  5) Stop

Algorithm

① Even odd →
1) Start
2) read the number
3) if number %2 ==0 is true go to 4, if false go to 5
4) Number is even
5) Number is odd
6) Stop

2] Algorithm to find factorial of a given number
1) Start
2) Declare variable n, fact=1, i=1
3) Get the input
4) if input is 0 print factorial is 1 else
5) repeat until i<=num
   fact = fact ×i ;
   i++
6) print factorial
7) Stop

4] Algorithm to swap two numbers without using third variable
1) Start
2) Get the input a & b
3) perform operations   a=a+b
                         b=a-b
                         a=a-b
4) print swapped numbers
5) Stop

20) Flowchart to print odd number series
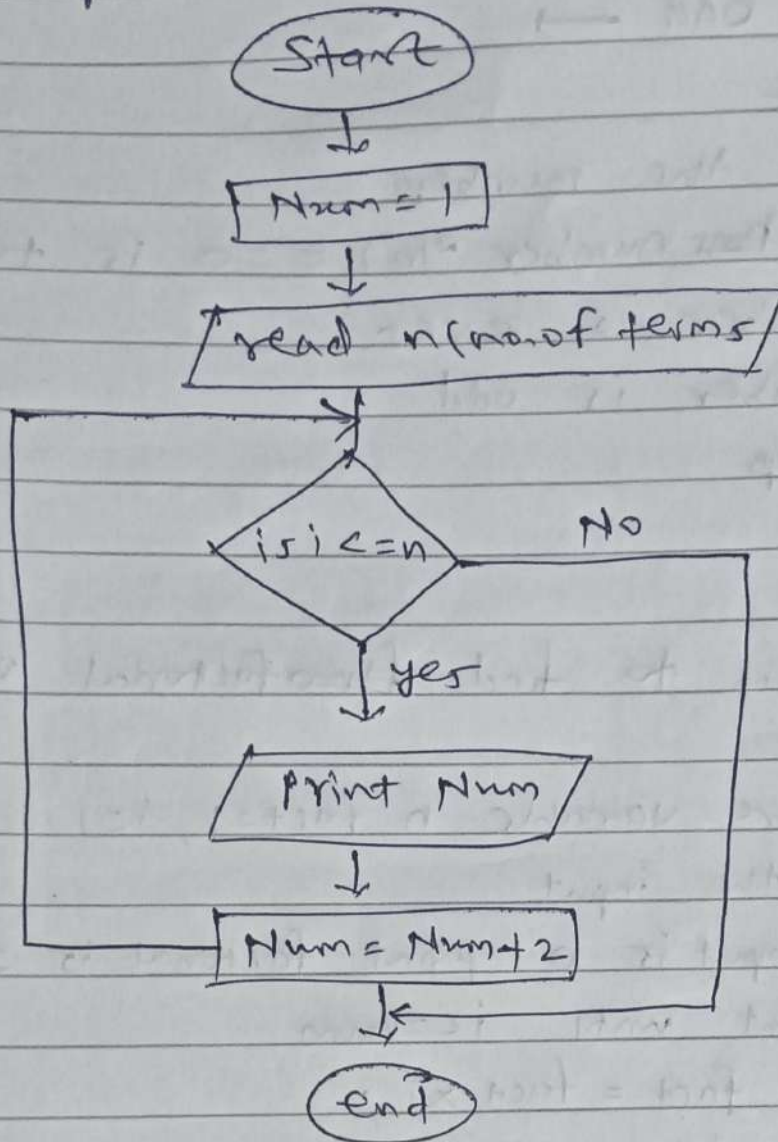
20) Algorithm to print odd number series

1) Start
2) Num = 1
3) read the input
4) is i <= n is true go to ⑤ else go to ⑦
5) print Num
6) Num = Num + 2 go to ④
7) end

15) Algorithm to find LCM of too numbers

1) Start
2) set gcd = 1
3) get n_1 & n_2
4) if i <= n_1 && i <= n_2 is true go to ⑤ else go to ⑦
5) if n_1 % i == 0 && n_2 % i == 0 go to ⑥
6) ~~print~~ gcd = i & i = i+1 go to ④
7) LCM = $\dfrac{n_1 \times n_2}{gcd}$

8) print LCM
9) stop