

# Assignment - 1

a. 1) Checks no is even or odd

→ Algorithm -

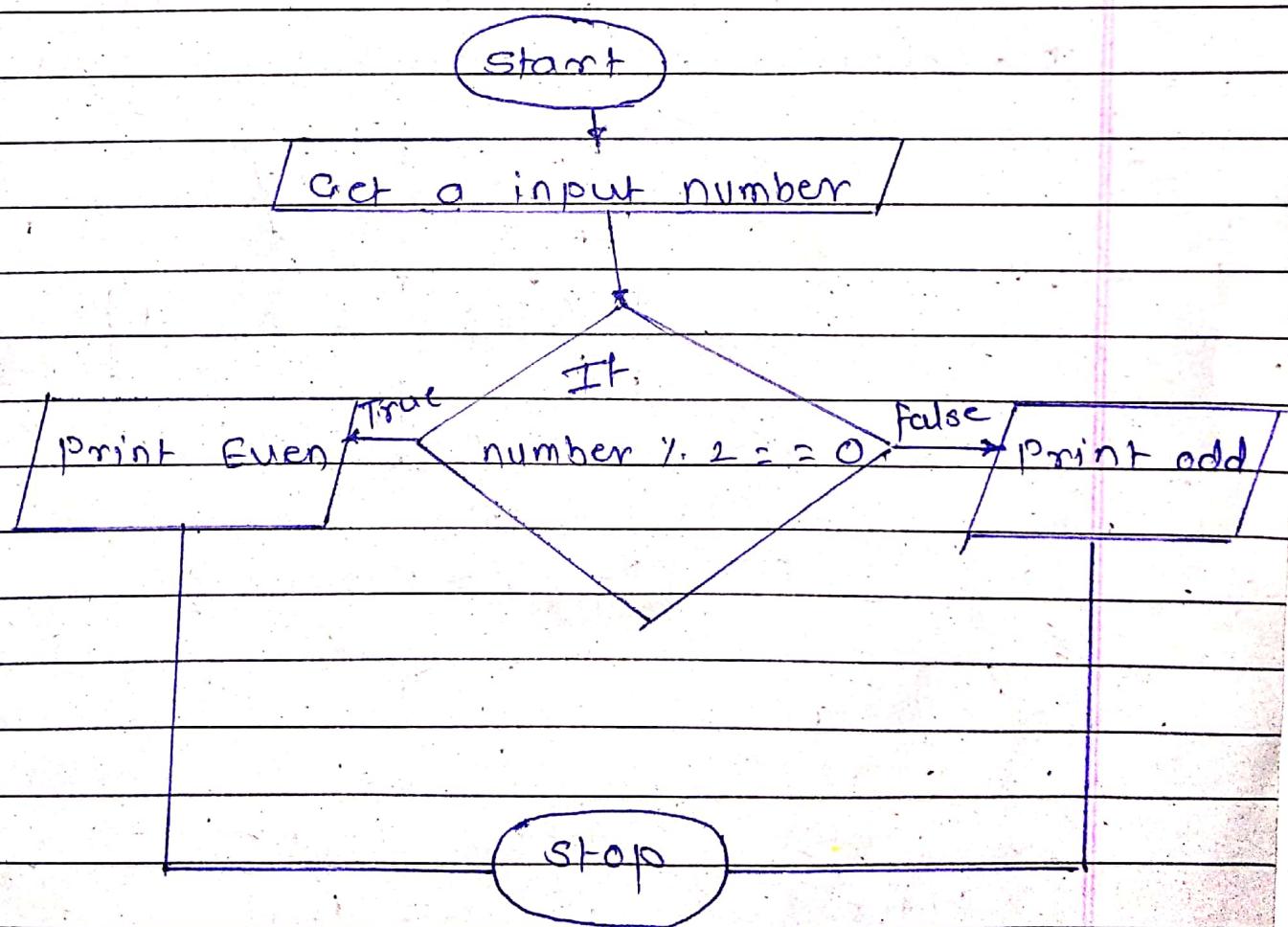
1) start -

2) Get input -

3) Check whether it is odd or even using  
 $\text{num} \% 2 == 0$

4) If true print even no . Else print odd no.

5) Stop -



Q.2) Factorial of given number.

→ Algorithm

1) Start

2) Declare variable num, fact = 1, i = 1

3) get a input number

4) Repeat until  $i \leq num$

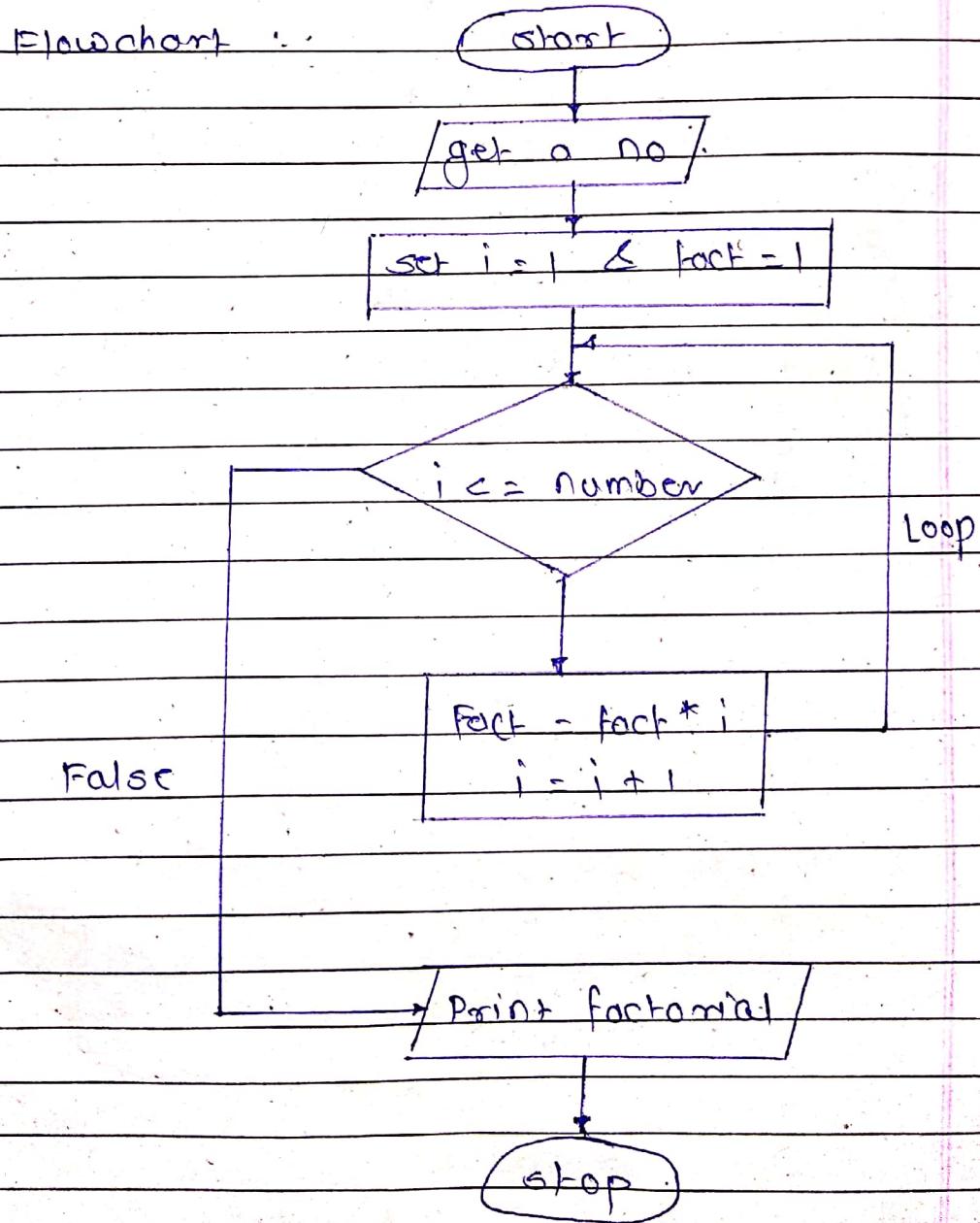
    Fact = fact \* i

$i++$

5) print factorial

6) Stop

Flowchart :-



Q. 8) Factorial using recursion.

→ Algorithm

1) Start.

2) Define variable fact = 1

3) get a no. from user

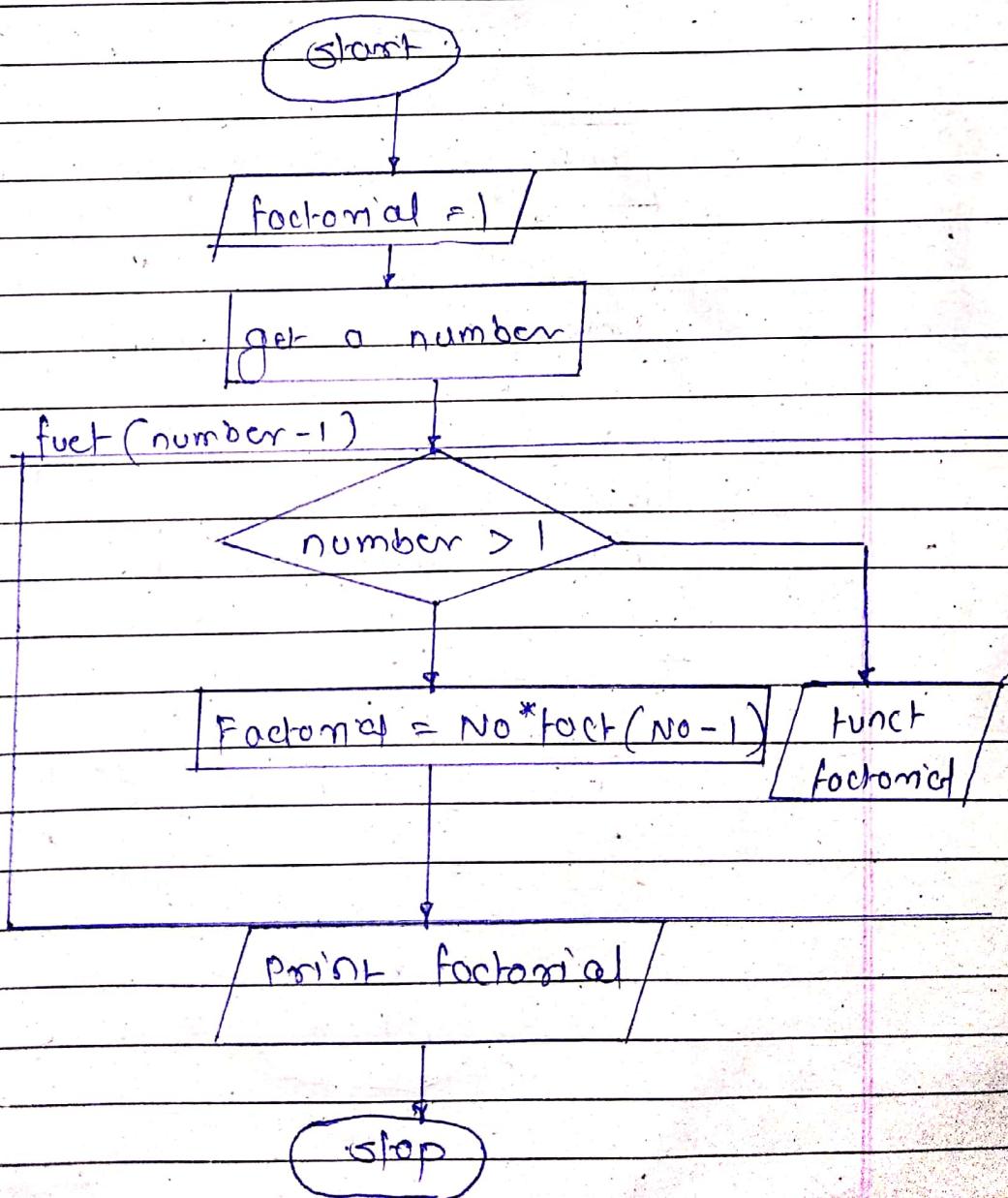
4) call method fact()

recursively until value of number > 1

5) print factorial.

6) Stop.

Flowchart :-



Q. 5) Swap 2 numbers without using third variable

→ 1) start :

2) Get two numbers num<sub>1</sub>, num<sub>2</sub> ;

3) print unswap numbers.

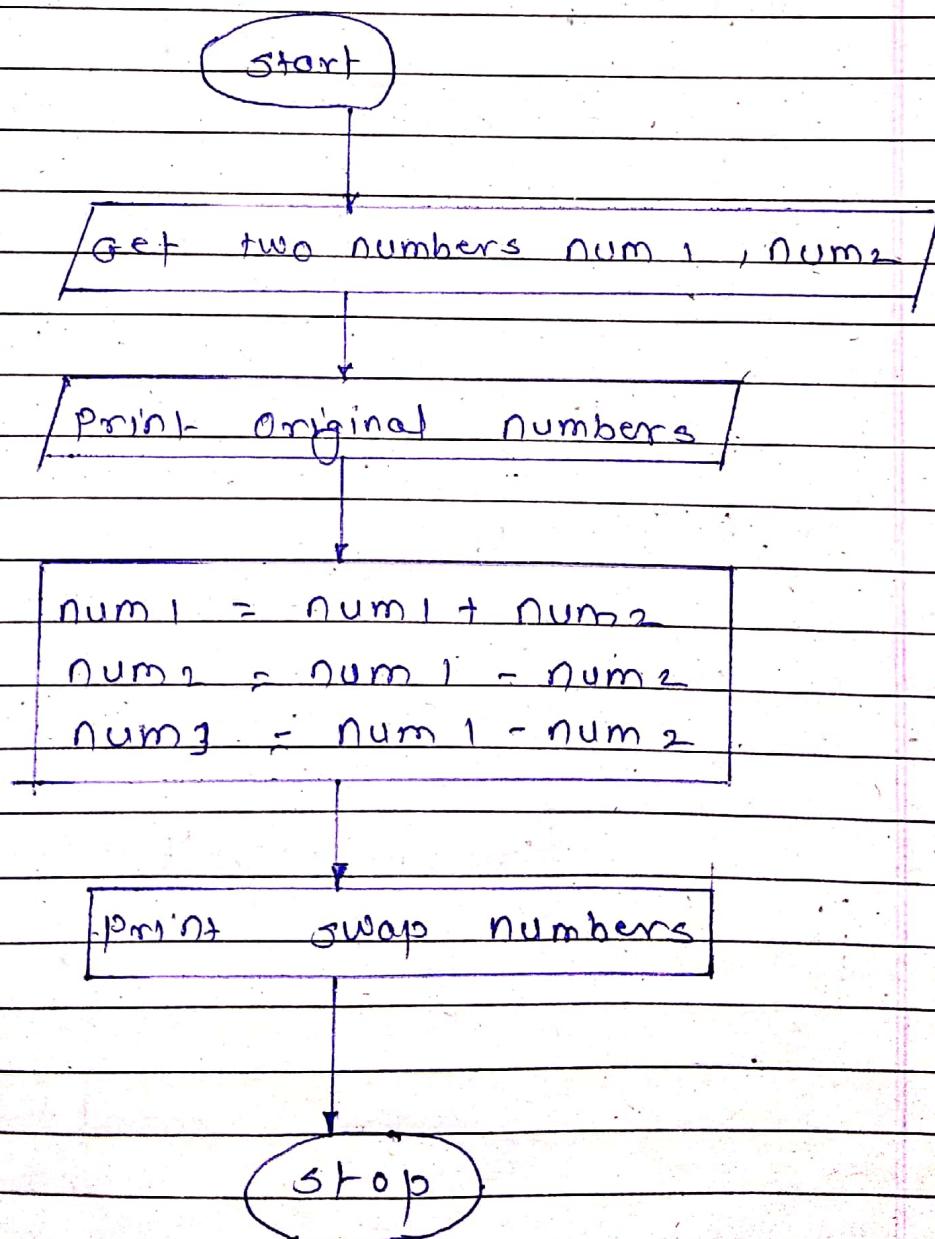
$$\text{num } 1 = \text{num } 1 + \text{num } 2$$

$$\text{num } 2 = \text{num } 1 - \text{num } 2$$

$$\text{num } 3 = \text{num } 1 - \text{num } 2$$

4) print swap numbers .

5) stop :



Q.5) check given no. whether is a positive or negative.

→ Algorithm ;

1) start .

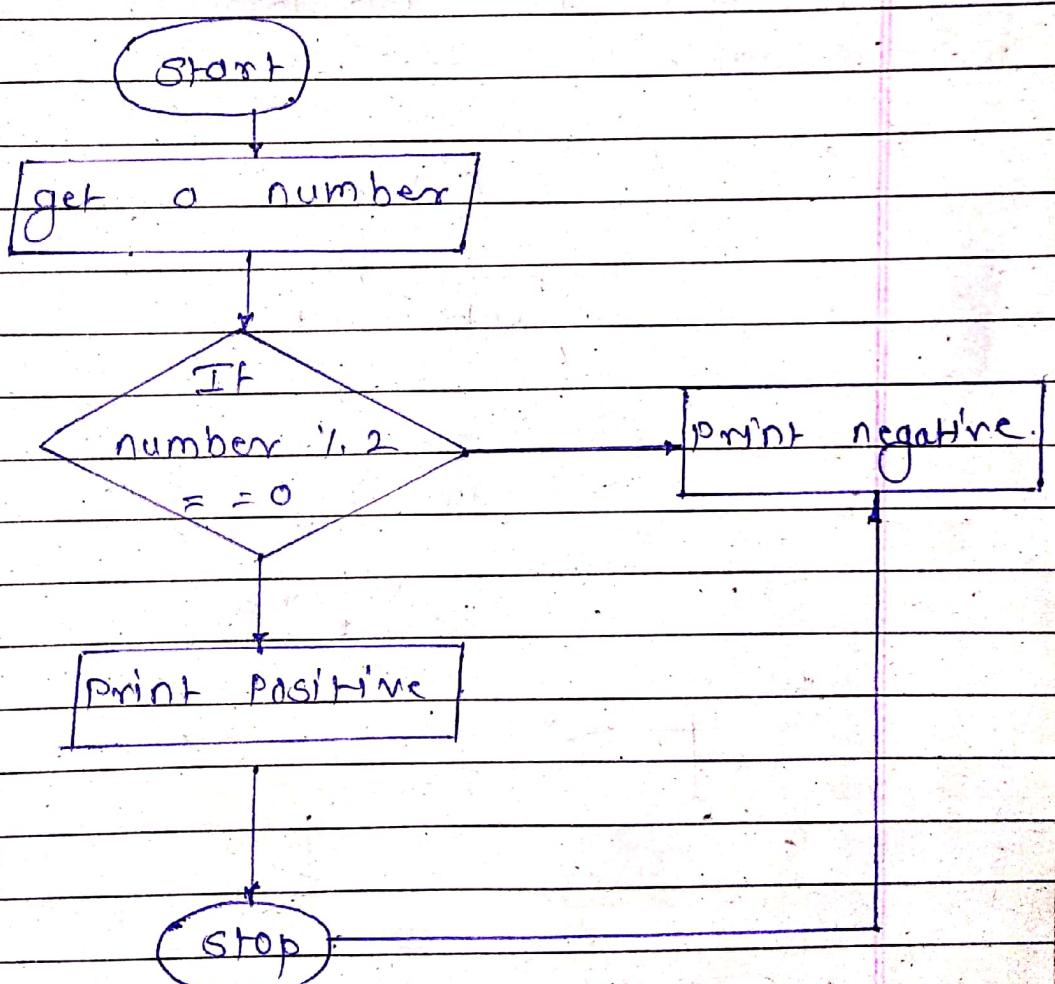
2) get a number .

3) Checks number  $\geq 0$

• If true , Print positive

else print negative .

4) stop .



a. e) Leap year.

→ 1) Start

2) get a input year.

3) check year divisible by 4, If true go to step 4 else go to step 7.

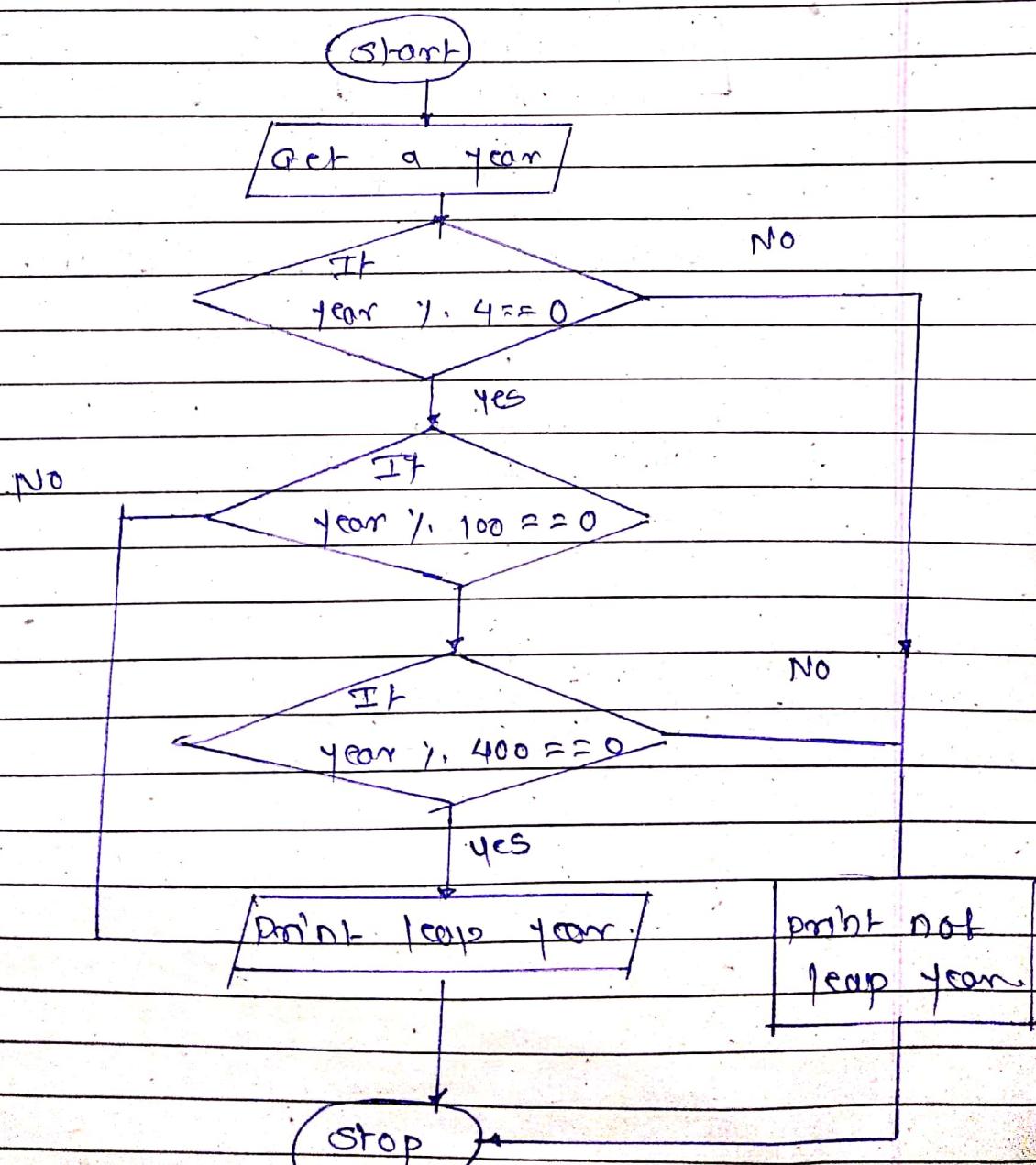
4) check year divisible by 100, If true go to step 5 else go to step 6.

5) check year divisible by 400, If true go to step 6, else go to step 7.

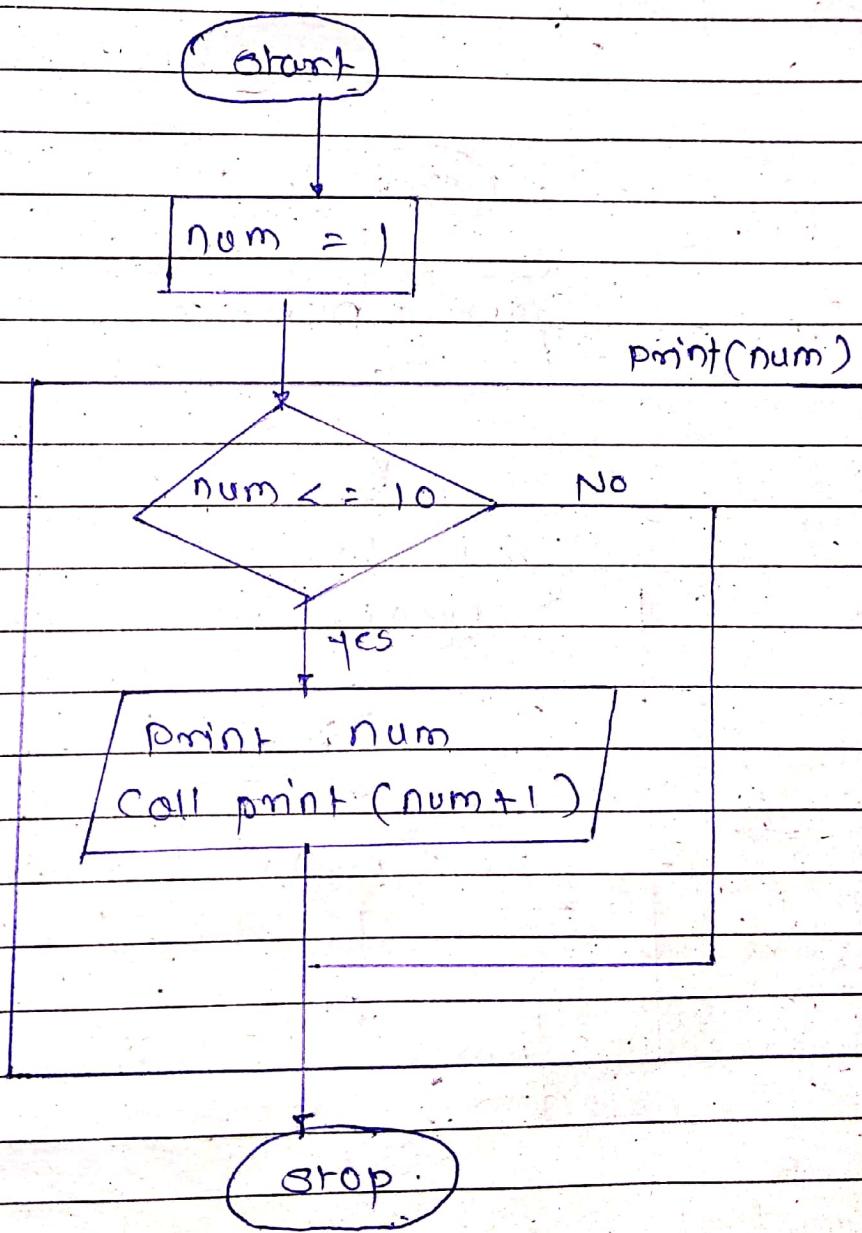
6) print leap year

7) print not leap year.

8) Stop



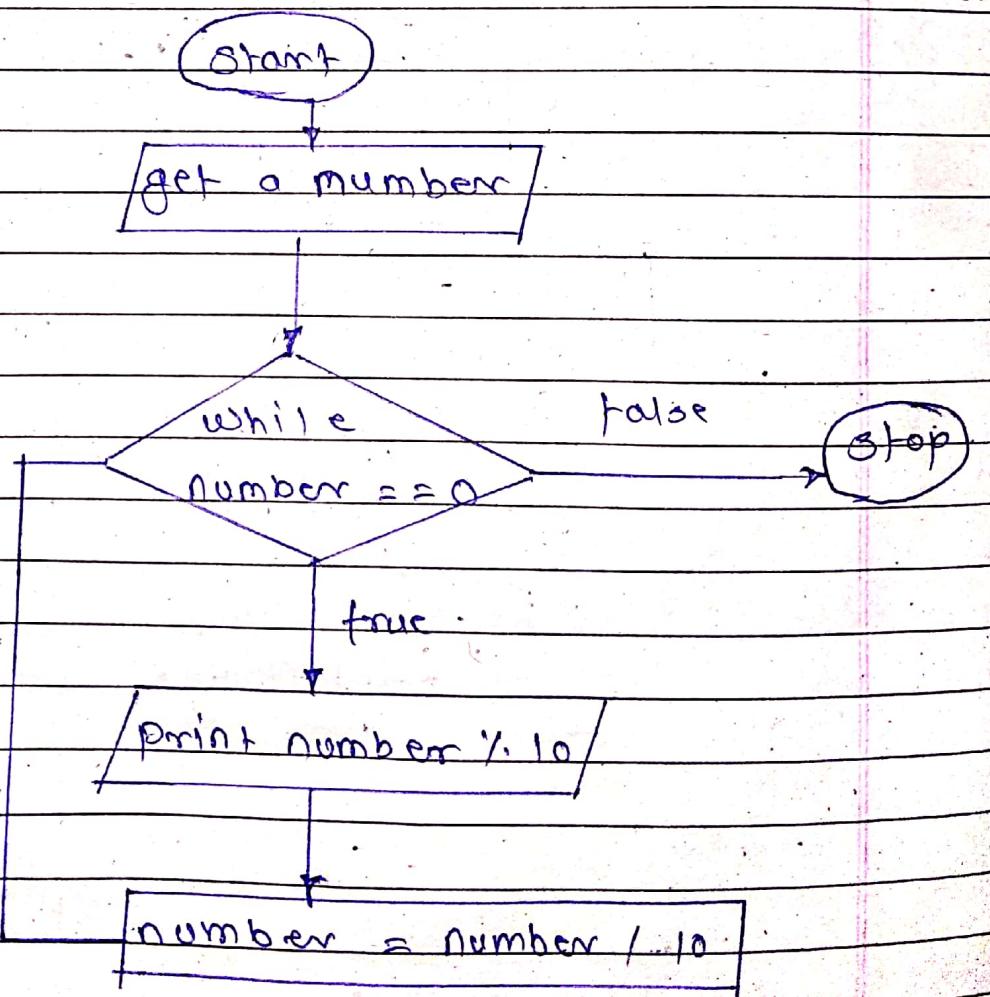
- Q.T) print 1 to 10 without loop.
- 
- 1) start.
  - 2) call print method.
  - 3) define a method print.
  - a. check num <= 10 If true print & recursively call print method with num-1  
else exit.
  - 4) stop.



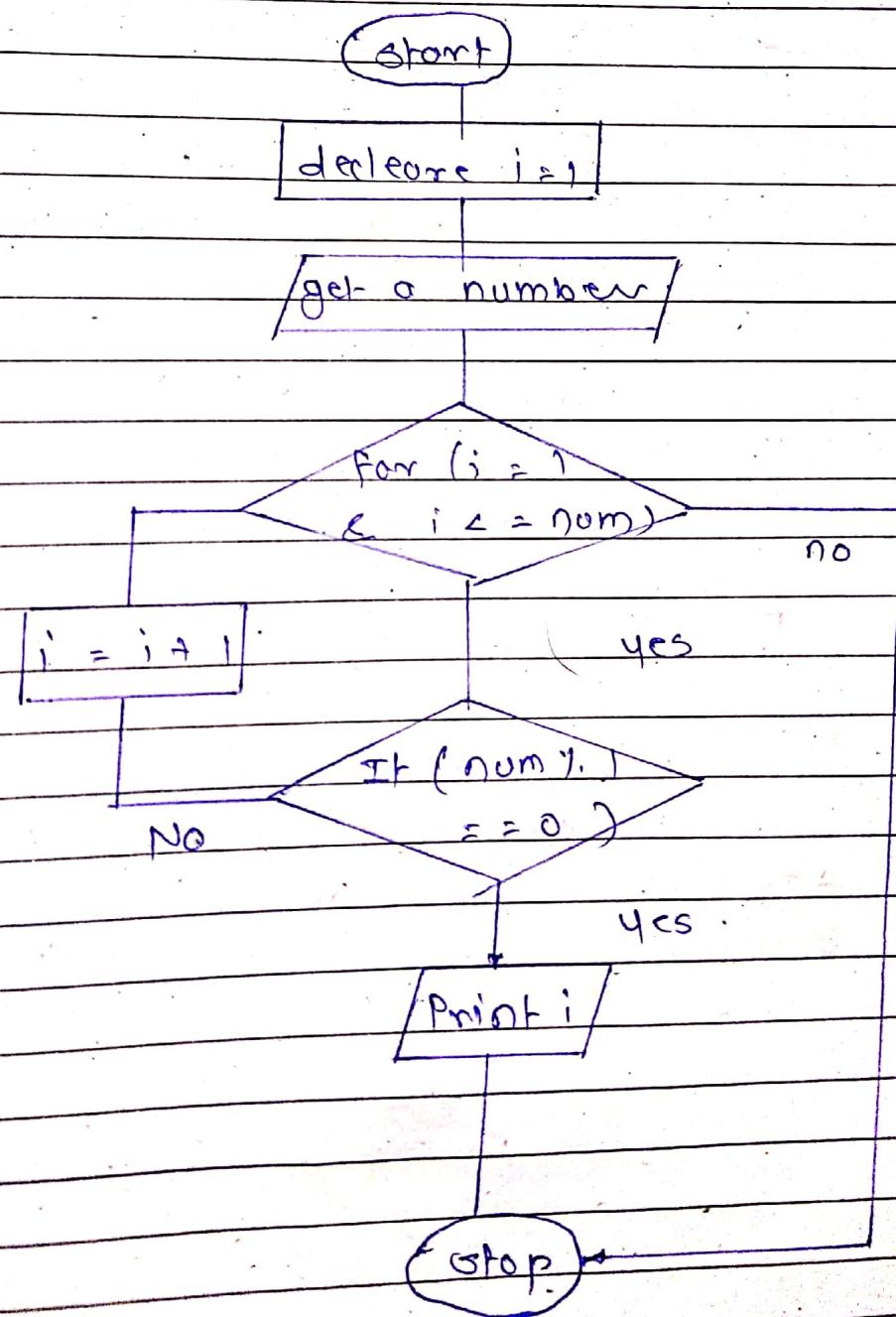
Q.8) Print digit at given number.

→ Algorithm :-

- 1) start .
- 2) get a number ,
- 3) print the value of num % 10
- 4) number = number / 10 ;
- 5) repeat steps 3 & 4 until number is not equal to zero .
- 6) stop



- Q. 9) Factor of given number.
- - 1) Start.
    - 2) get a number.
    - 3) Declaare  $i = 1$
    - 4) Check number  $i : i == 0$  If true - print  $i$  & increment the value of  $i$ ;
    - 5) Repeat Step 4 until  $i \leq \text{number}$
    - 6) Stop.



Q.10) sum of digit of given number.

→ Algorithm :

1) start

2) get a number

3) set num = 1

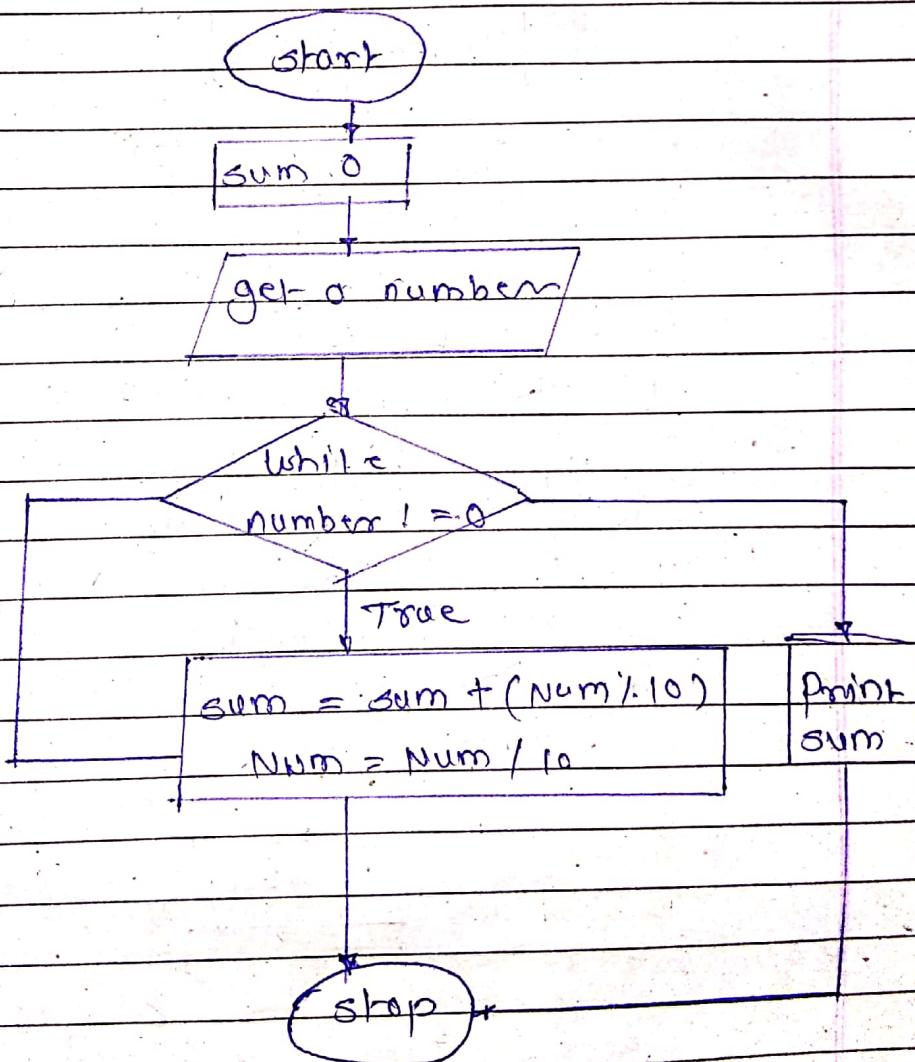
4) while (number != 0)

    sum = sum + (Number % 10)

    num = Num / 10

5) print sum

6) stop



Q.11) smallest of three numbers.

→ 1) start -

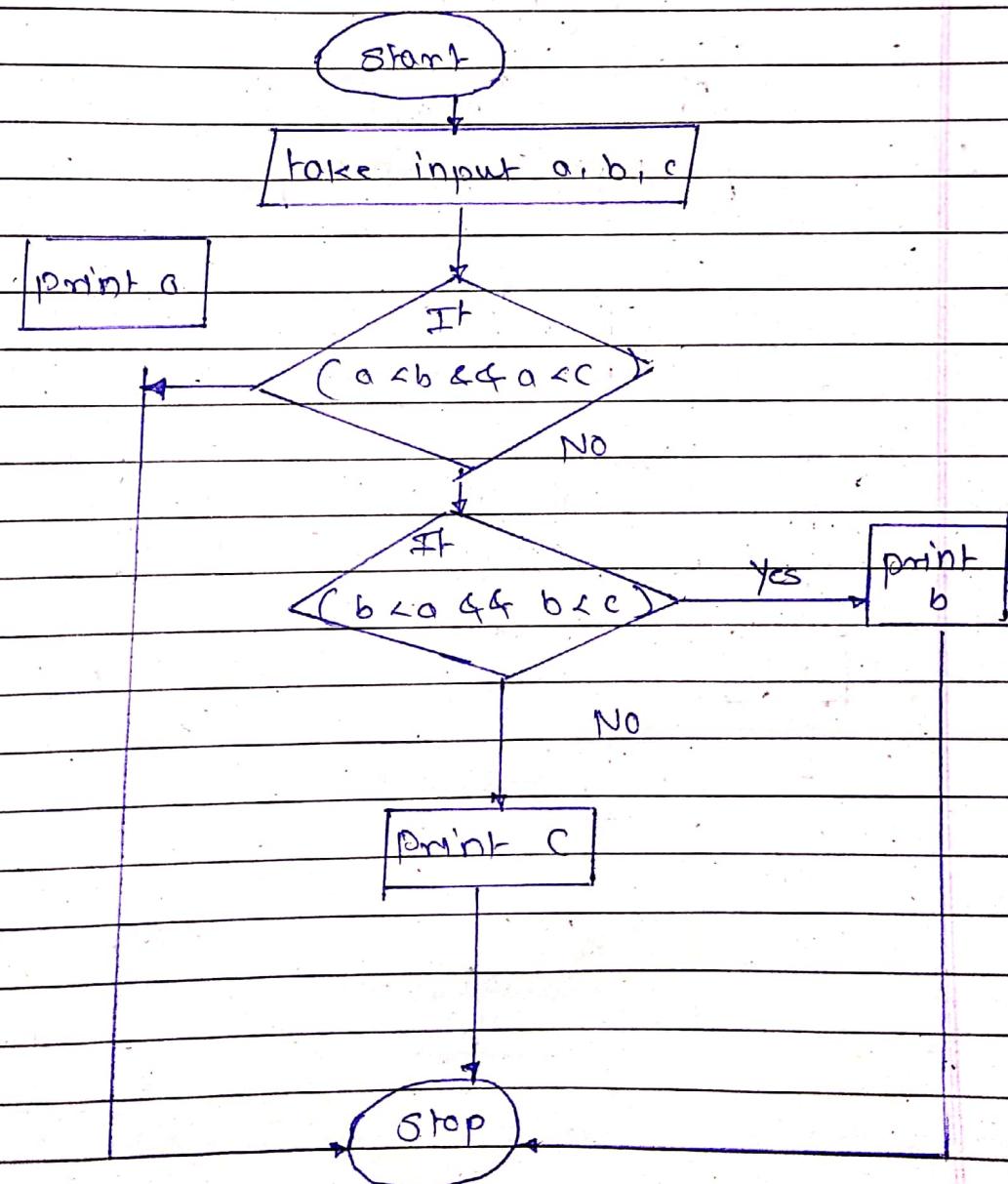
2) get three number from user.

3) check if  $a < b$  &  $a < c$ . If true print a & exit else go to step 4.

4) check if  $b < a$  and  $b < c$ . If true print b and exit else go to step 5.

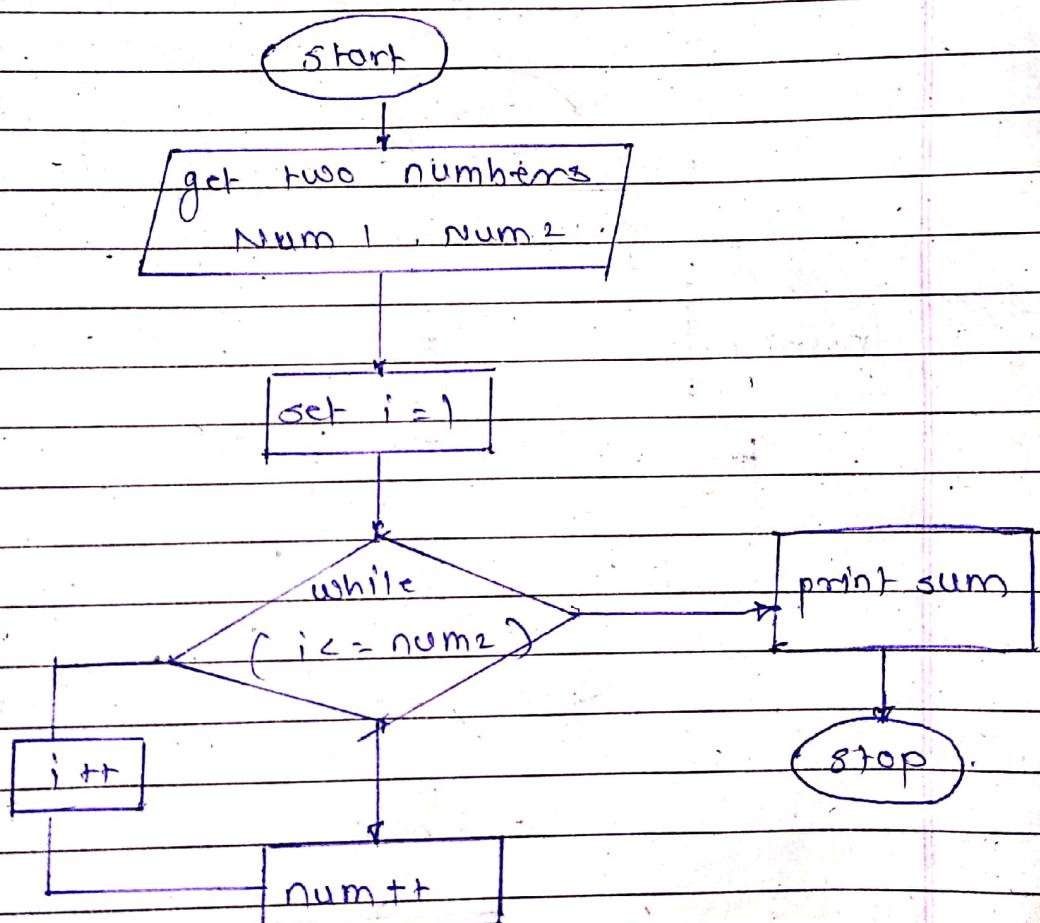
5) print c

6) stop .



Q.12) Addition without arithmetic operators.

- 1) start
- 2) get two numbers
- 3) call add\_num (num1, num2) method
- 4) for ( $i=1$ ;  $i < num2$ ;  $i++$ )  
    a)  $num++$
- 5) Print sum
- 6) stop



Q. 10) Reverse a given number.

→ Algorithm

1) start.

2) get a number.

3) set rev = 0, rem = 0

4) while (number != 0)

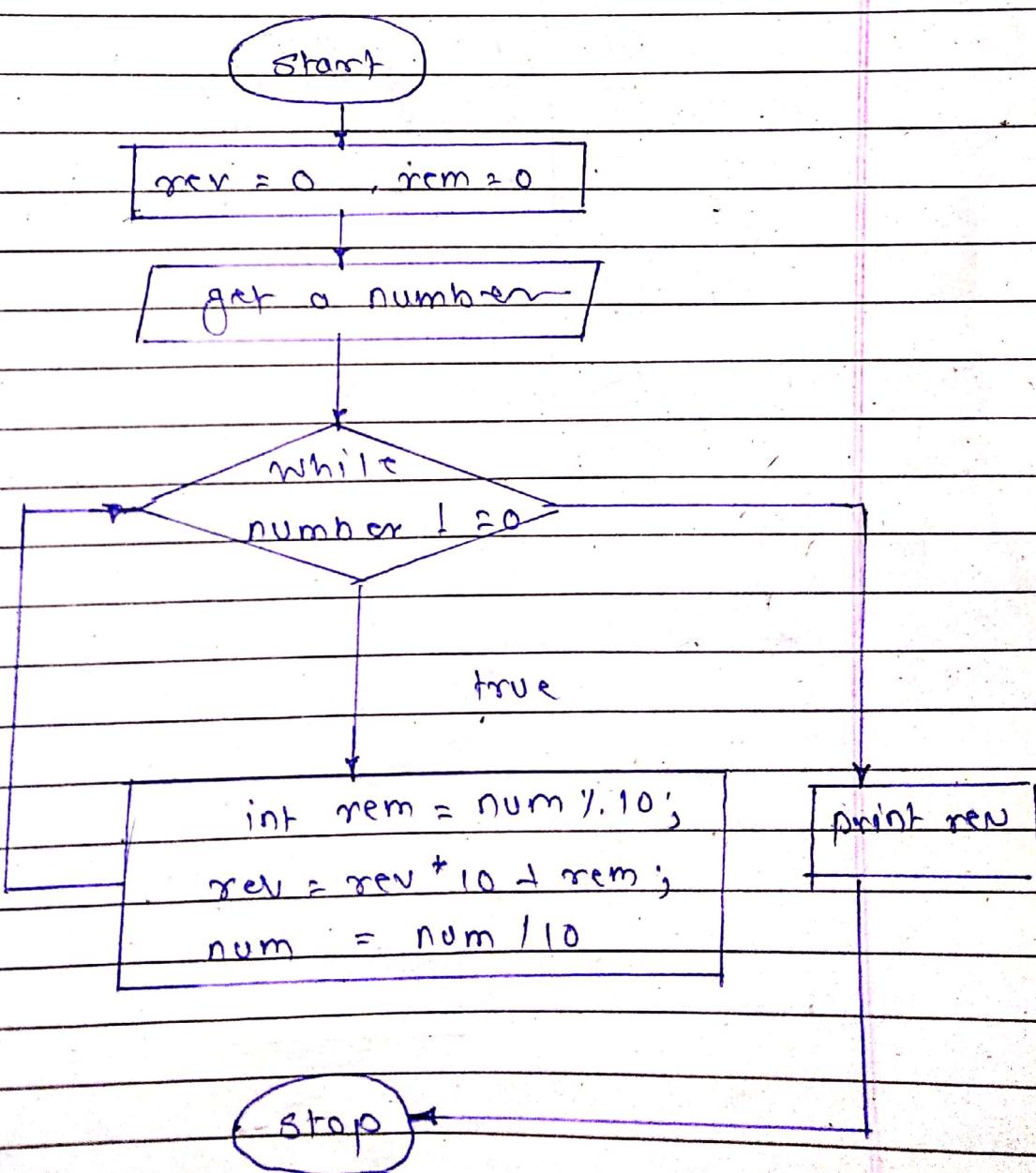
a. int rem = num % 10

b. rev = rev \* 10 + rem

c. num = num / 10

5) print rev.

6) stop.



Q.14) GCD of two numbers.

→ 1) Start --

2) get two no. num1, num2.

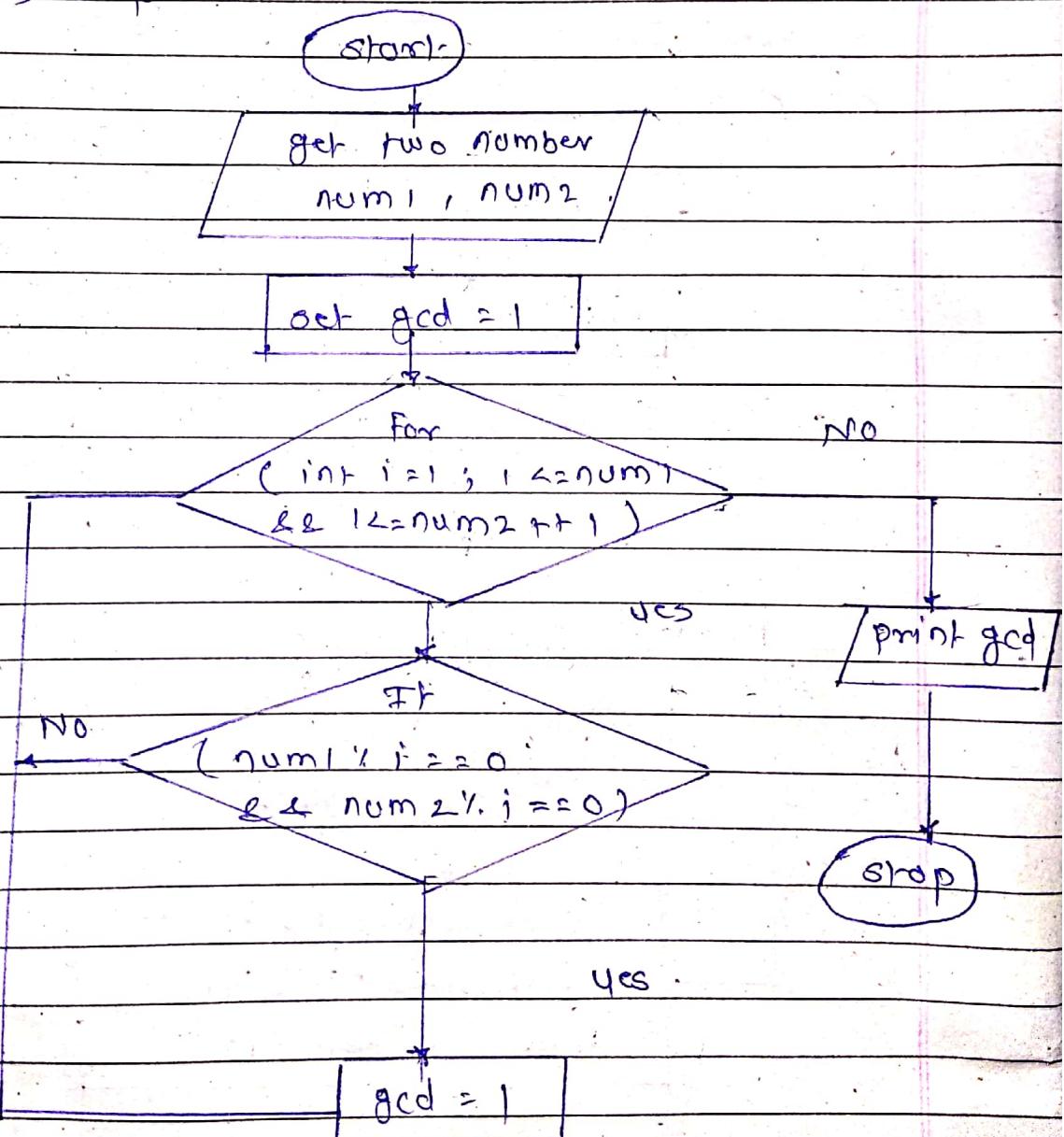
3) get gcd = 1

4) for (int i=1; i<=num) &&

i <= num2, ++i. IF (num1 % i == 0 && num2  
== 0)

5) Print GCD.

6) Stop.



a.15) Lcm of two numbers:

→ Algorithm :

1) Start.

2) get two number num1, num2.

3) set gcd = 1.

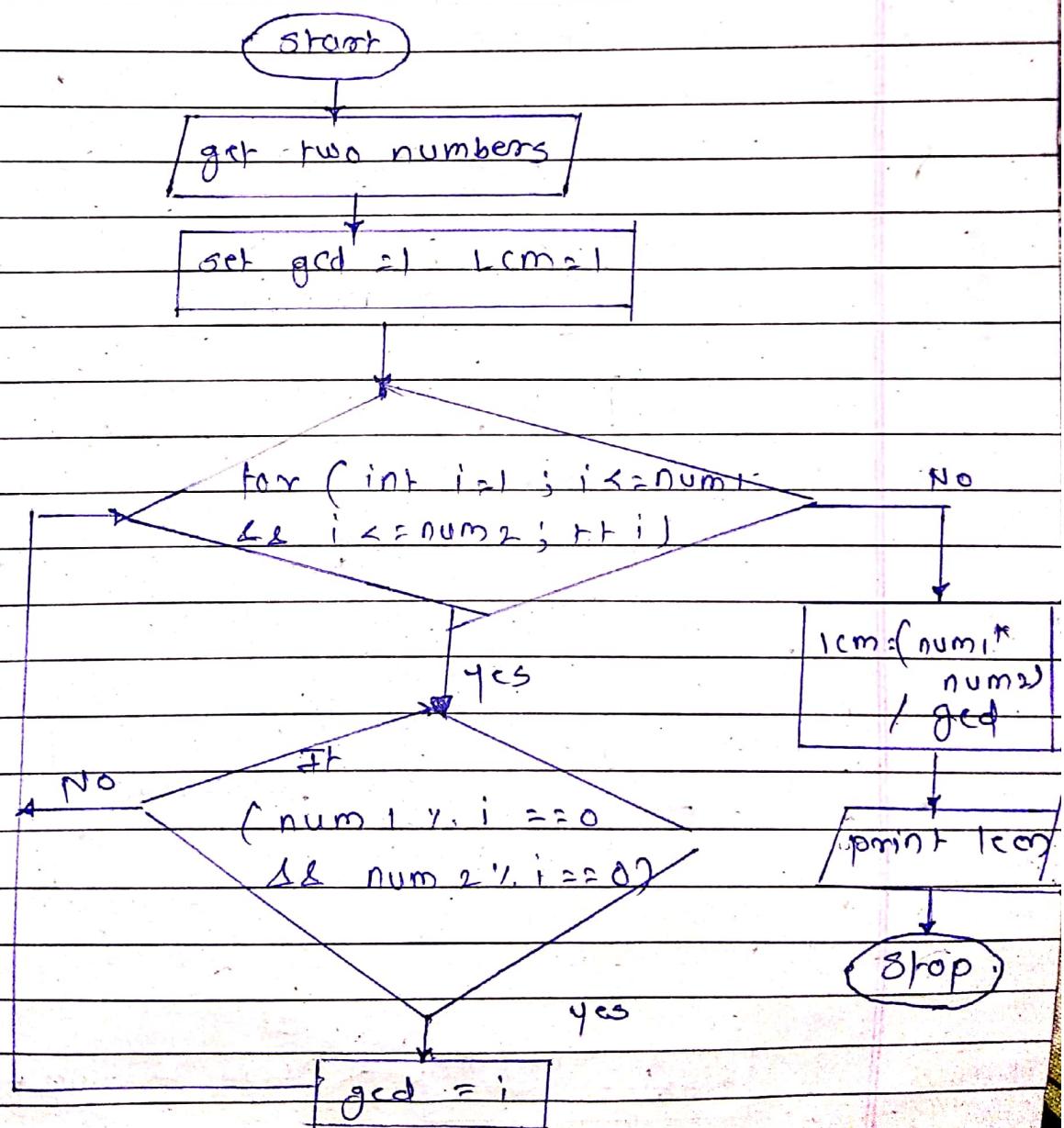
4) for ( int i=1, i<=num1 && num2 ; ++i )

If ( num1 % i == 0 && num2 % i == 0 )

5) lcm = ( num1 \* num2 ) / gcd.

6) print lcm.

7) Stop.



check palindrome or not



1) start.

2) get a number.

3) reverse = 0 & rem=0

4) set original number.

5) check if number 1=0 if true go to 5 else go to 7.

6) rem = num % 10;

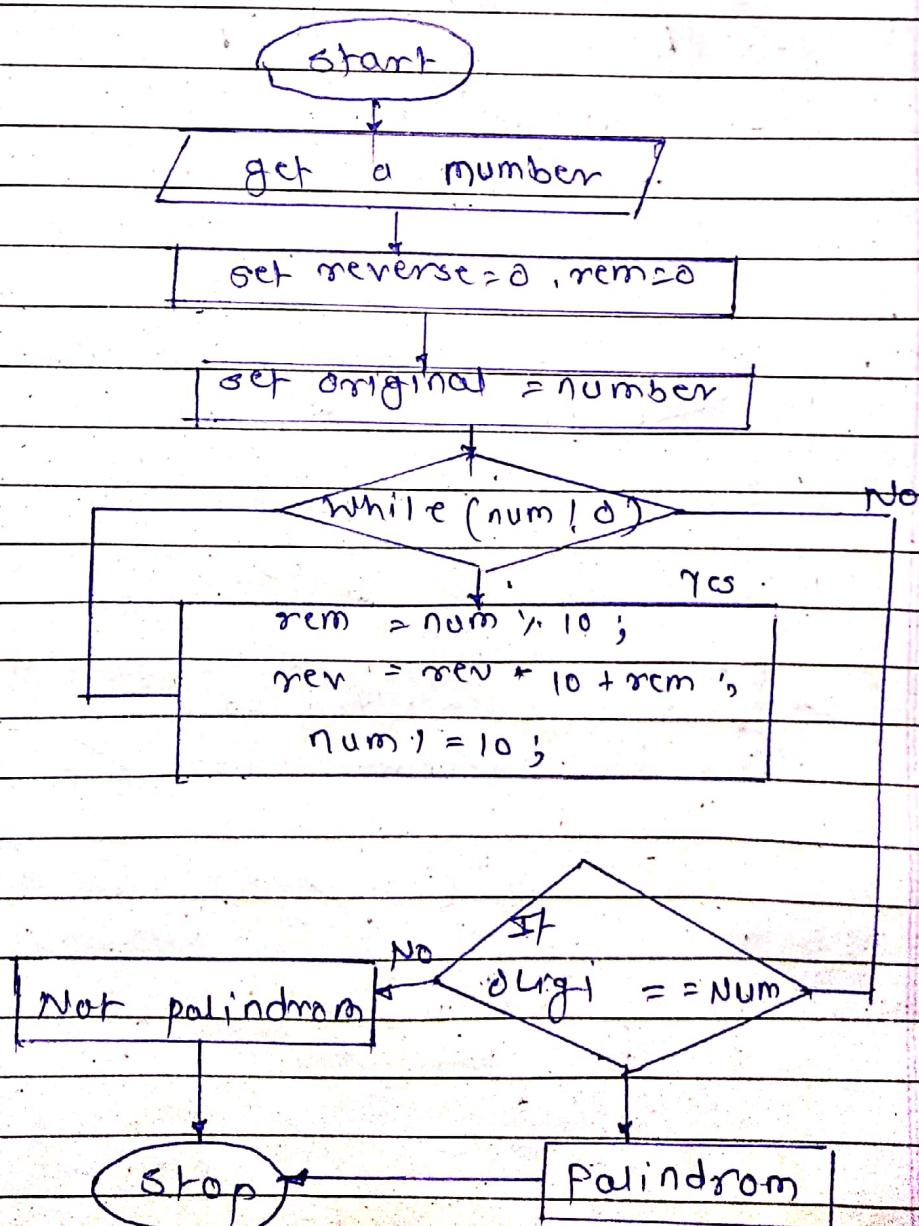
rev = rev \* 10 + rem;

num / = 10 ;

7) check if original == number if true.

print palindrome.

8) stop.



Q.18) Prime factor of given number

→ Algorithm :

1) Start.

2) Enter the num.

3) take  $i = 2$

4) check the input number :

is greater than enter in loop :

a. while ( number is greater than 1 )

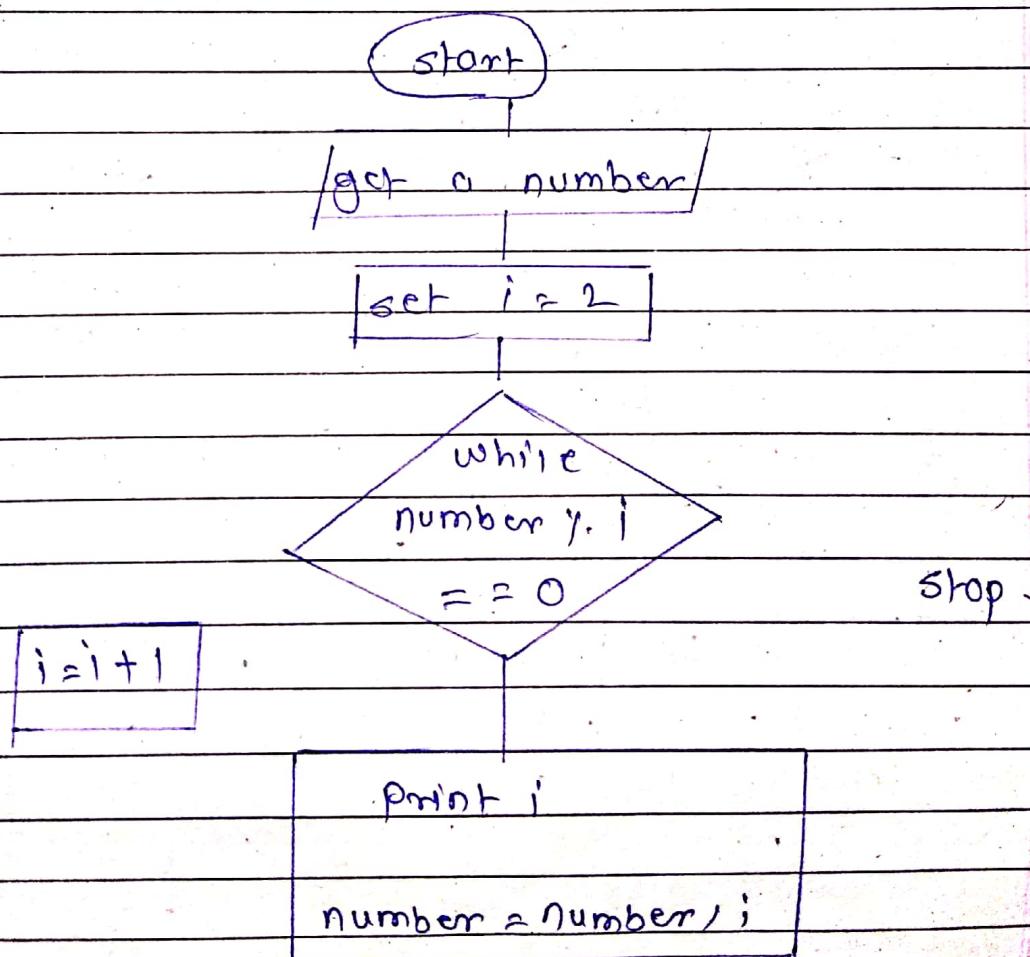
b. check the cond" if ( $num/i == 0$ ) .

c. If it is true enter in bracket -

d. print ( i ) value on terminated .

e. Number = number /  $i$  else if then loop  
will

5) Stop.



a. 19) Even no. semics.

→ Algorithm

1) start

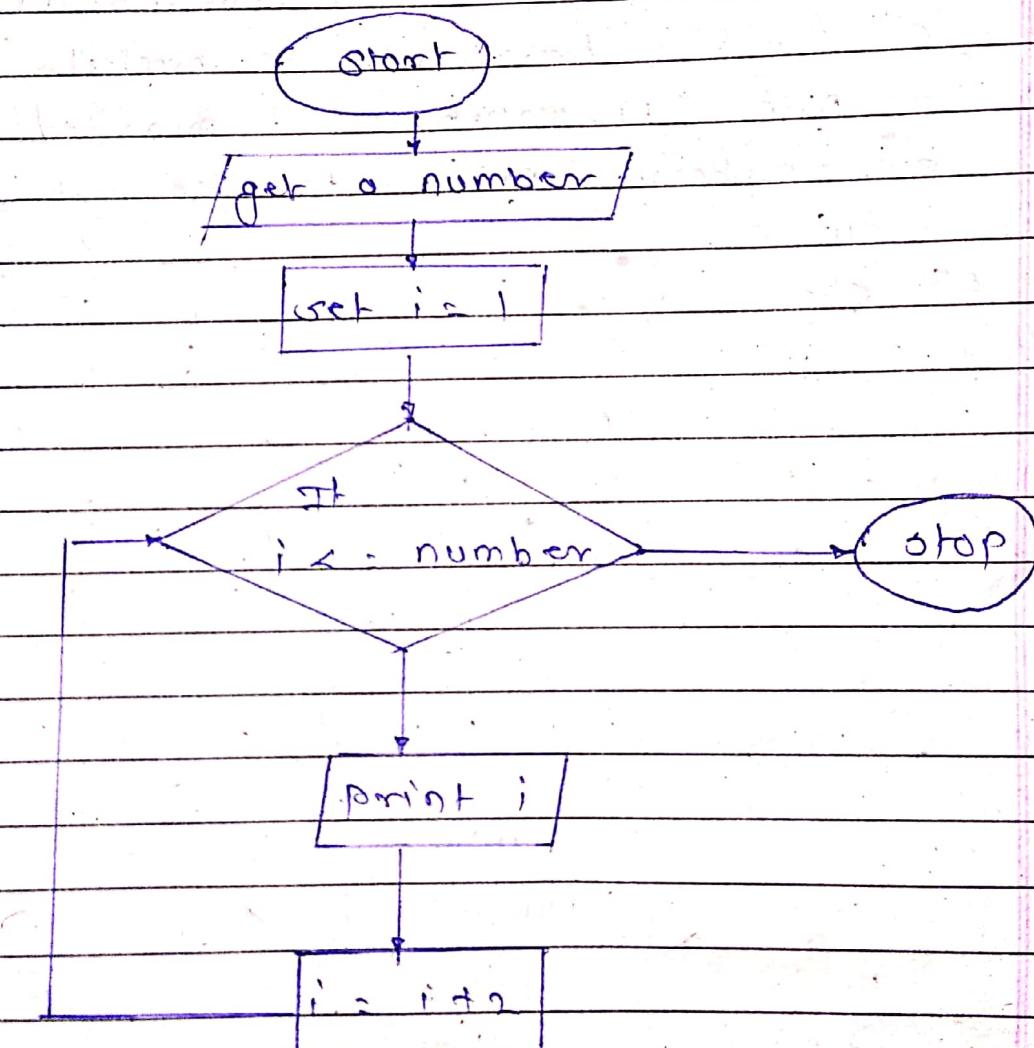
2) get a number

3) set i = 2

4)  $i \leq \text{number}$  print  $i$  &  $i = i + 2$

5) repeat step 4 until  $i \leq \text{number}$

6) stop



Q. 20) odd no series .

→ Algorithm :

1) start .

2) get a number .

3) set i = 1 .

4) If  $i \leq \text{number}$  print i &  $i = i + 2$

5) repeat step 4 until  $i \geq \text{number}$  .

6) stop .

