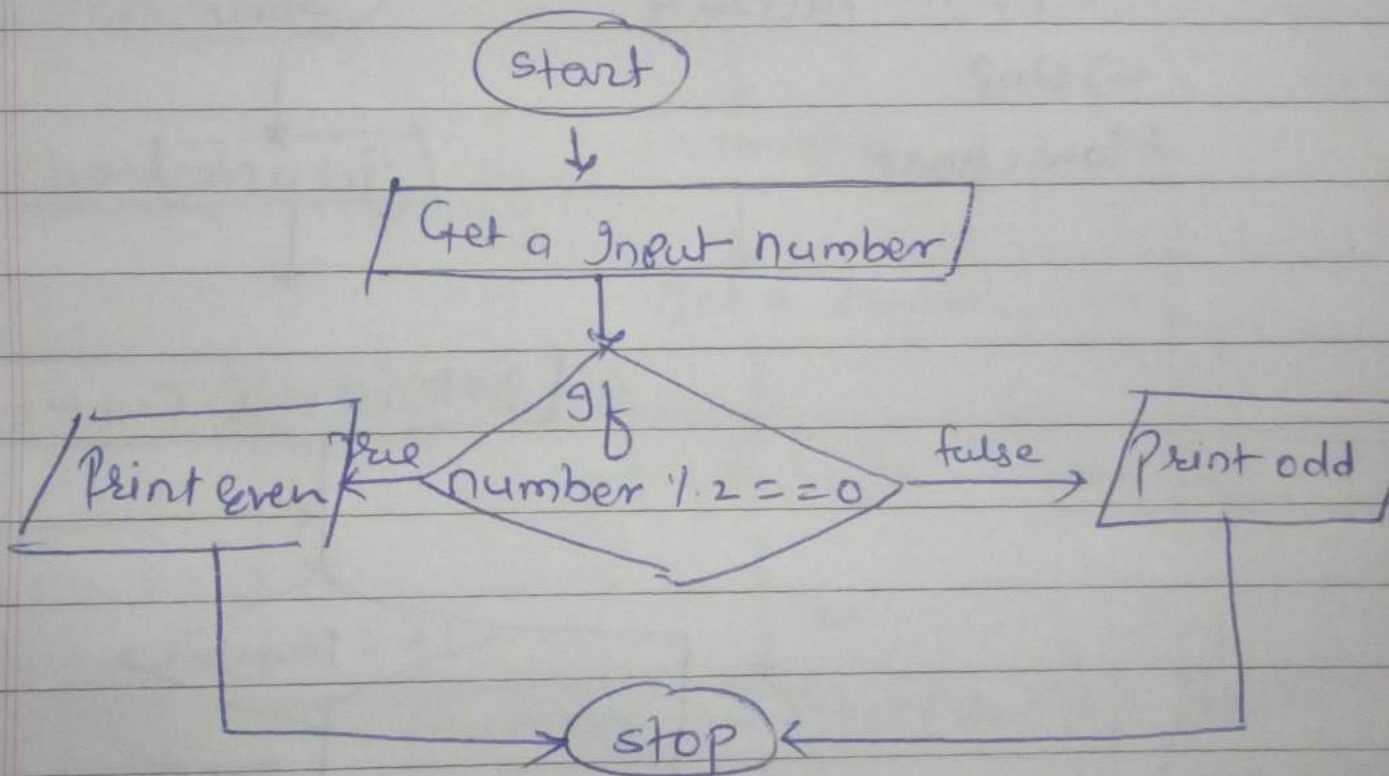Assignment 1

Q1) check no is even or odd

→ Algorithm -

1) start

2) Get input

3) Check whether it is odd or even using num %.2 == 0

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

5) stop

```
            ( start )
                ↓
    | Get a input number |
                ↓
               if
    true     number %.2==0     false
 | Print even |              | Print odd |
                ↓
             ( stop )
```
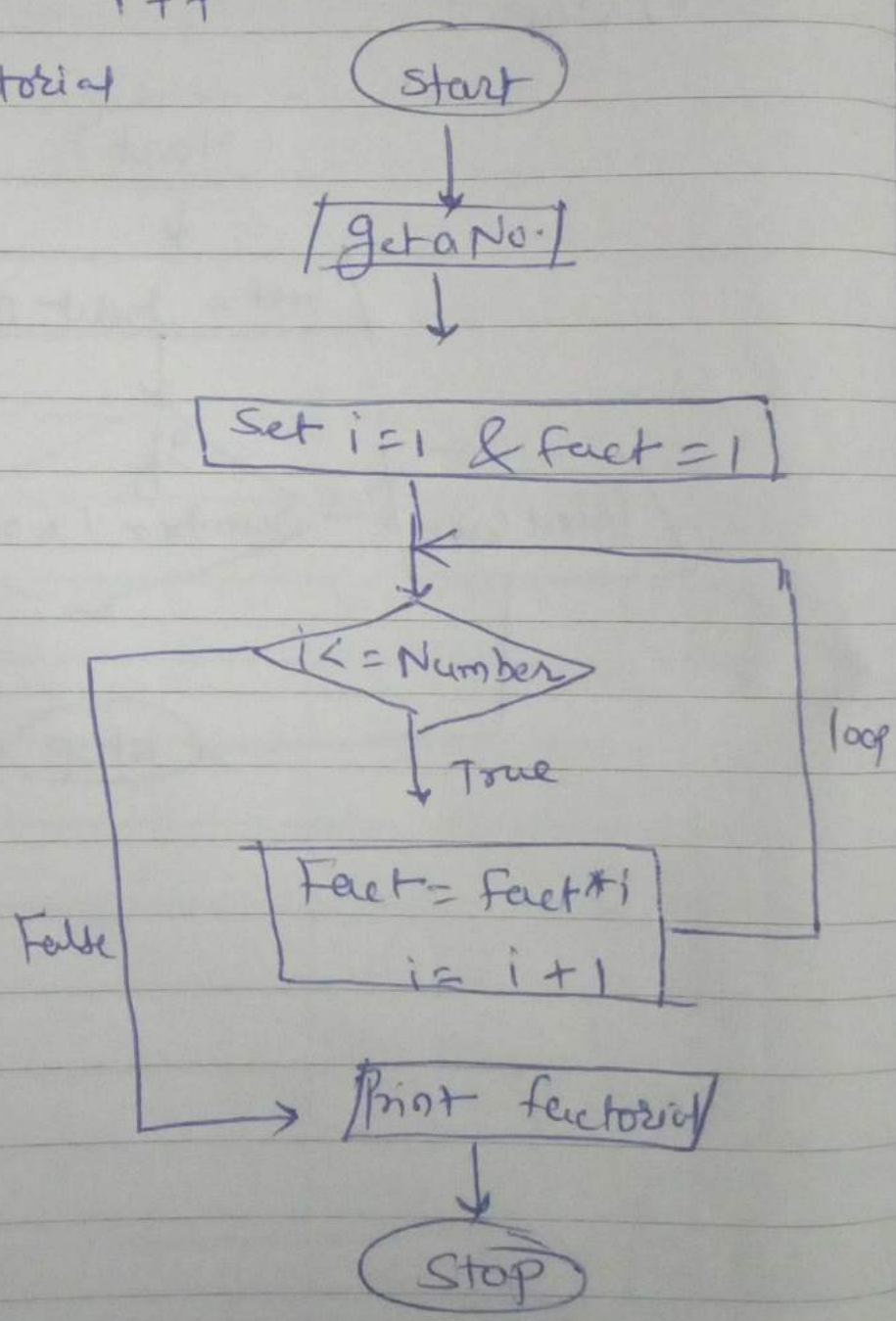
Q2) factorial of given Number.

Algorithm :-

1) Start

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

3) get a input number

4) Repeat until i <= num

   $Fact = fact * i$

   i++

5) Print factorial

6) Stop

Flowchart

```
        ( Start )
            |
            v
       / get a No. /
            |
            v
  | Set i = 1 & fact = 1 |
            |
            v
        <i <= Number>  -----> loop
            |
          True
            |
            v
     | Fact = fact * i |
     |    i = i + 1    |

  False
            |
            v
      / Print factorial /
            |
            v
        ( Stop )
```
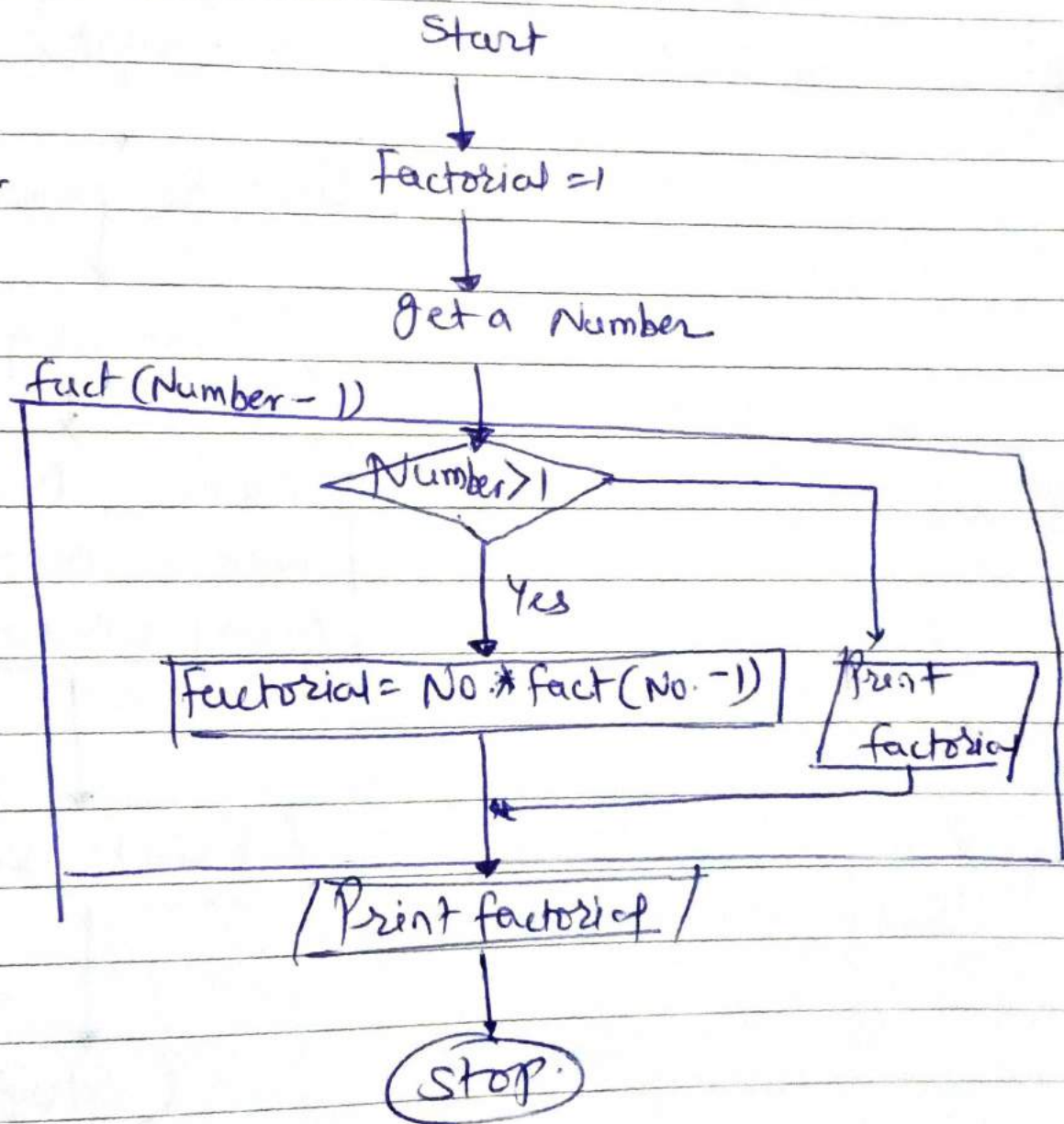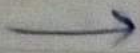
Q3) Factorial using recursion.

→ Algorithm.

1) Start

2) Declare variable fact = 1

3) Get a number from user

4) Call method facto
   Recursively Until value of Number >1

5) Print factorial

6) Stop.

Flowchart :-

```
                              Start
                                |
                                v
                          Factorial = 1
                                |
                                v
                          Get a Number
                                |
   fact (Number - 1)           |
   +----------------------------+
   |                        Number >1 ----------------+
   |                            |                      |
   |                           Yes                     v
   |                            |                   Print
   |                            v                   factorial
   +----> Factorial = No * fact (No. - 1)              |
                                |                      |
                                v <--------------------+
                          Print factorial
                                |
                                v
                             Stop
```

Q4)   Swap 2 Numbers without using third variable

→ 1) start

2) get two numbers Num1, Num2;

3) print unswap numbers

Num1 = Num1 + Num2

Num2 = Num1 - Num2

Num3 = Num1 - Num2

4) print swap Number

5) stop

Start

Get two Numbers Num1, Num2

Print original Numbers

Num1 = Num1 + Num2
Num2 = Num1 - Num2
Num1 = Num1 - Num2

Print Swap Numbers
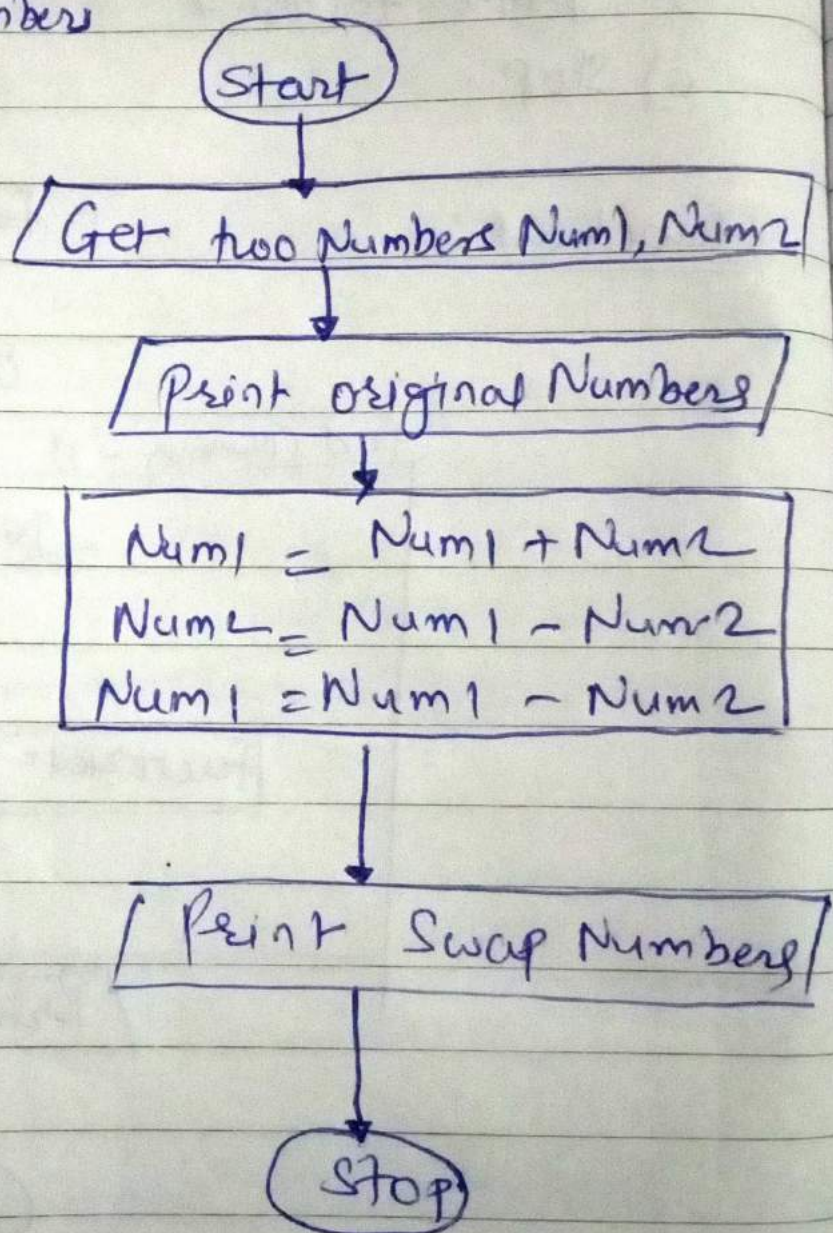
Stop

Q5) cheek given No. whether it is Positive or Negative

→ Algorithm;

1) stop

2) get a Number

3) cheek Number %2 == 0

☺ 9f true, Print Positive

Else Print negative

4) stop

```
        ┌─────────┐
        │  Start  │
        └─────────┘
             │
             ▼
      ┌──────────────┐
      │ get a Number │
      └──────────────┘
             │
             ▼
          ╱──────╲
         ╱   9f    ╲      False
        ╱  Number %2 ╲─────────────►  Print negative
        ╲   == 0    ╱
         ╲─────────╱
             │
             ▼
      ┌────────────────┐
      │ Print Positive │
      └────────────────┘
             │
             ▼
        ┌─────────┐
        │  stop   │ ◄──────────────
        └─────────┘
```
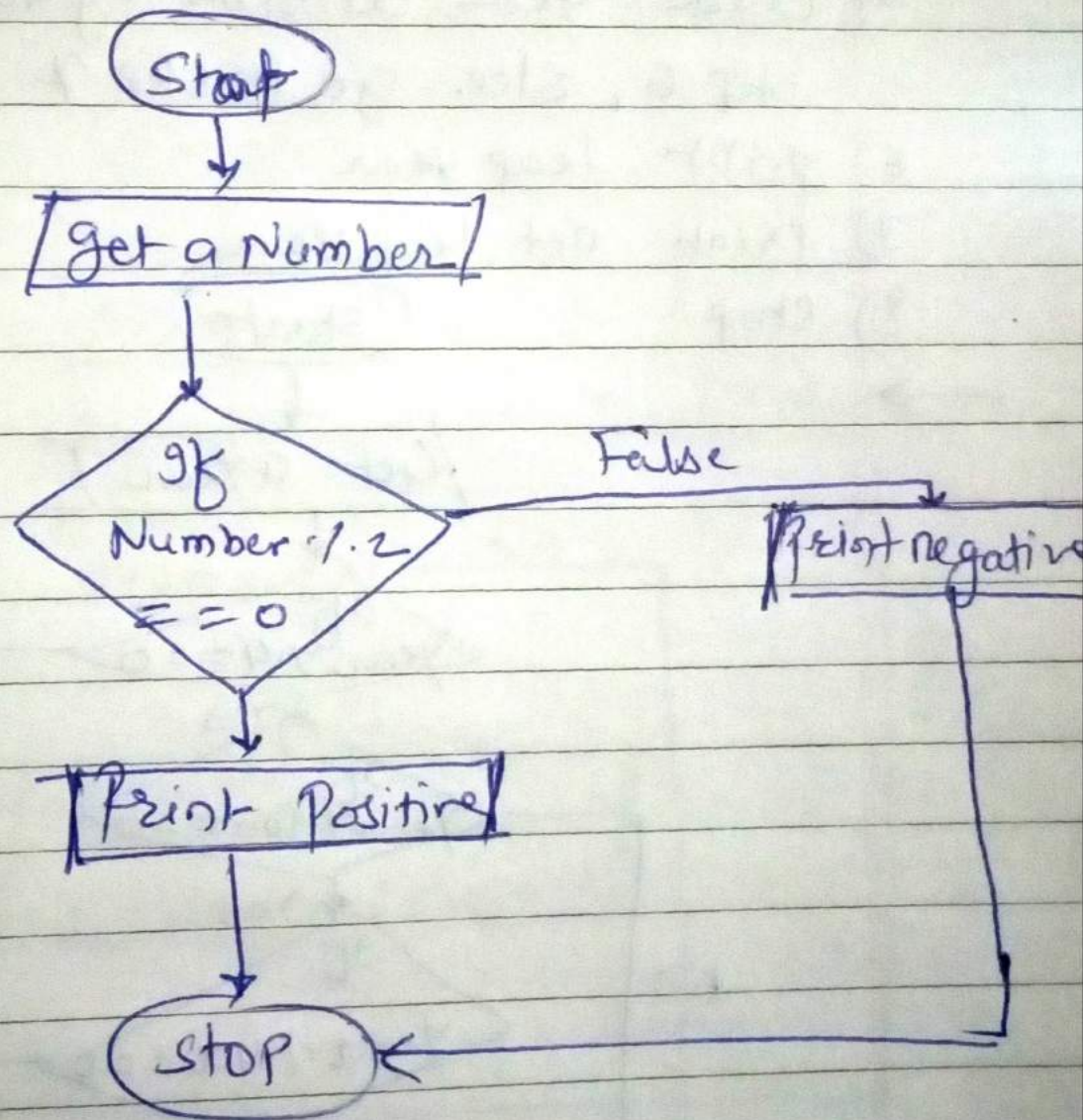
Q6) Leap Year

→ 1) Start
2) get a input year
3) check Year divisible 4, If true go to step 4
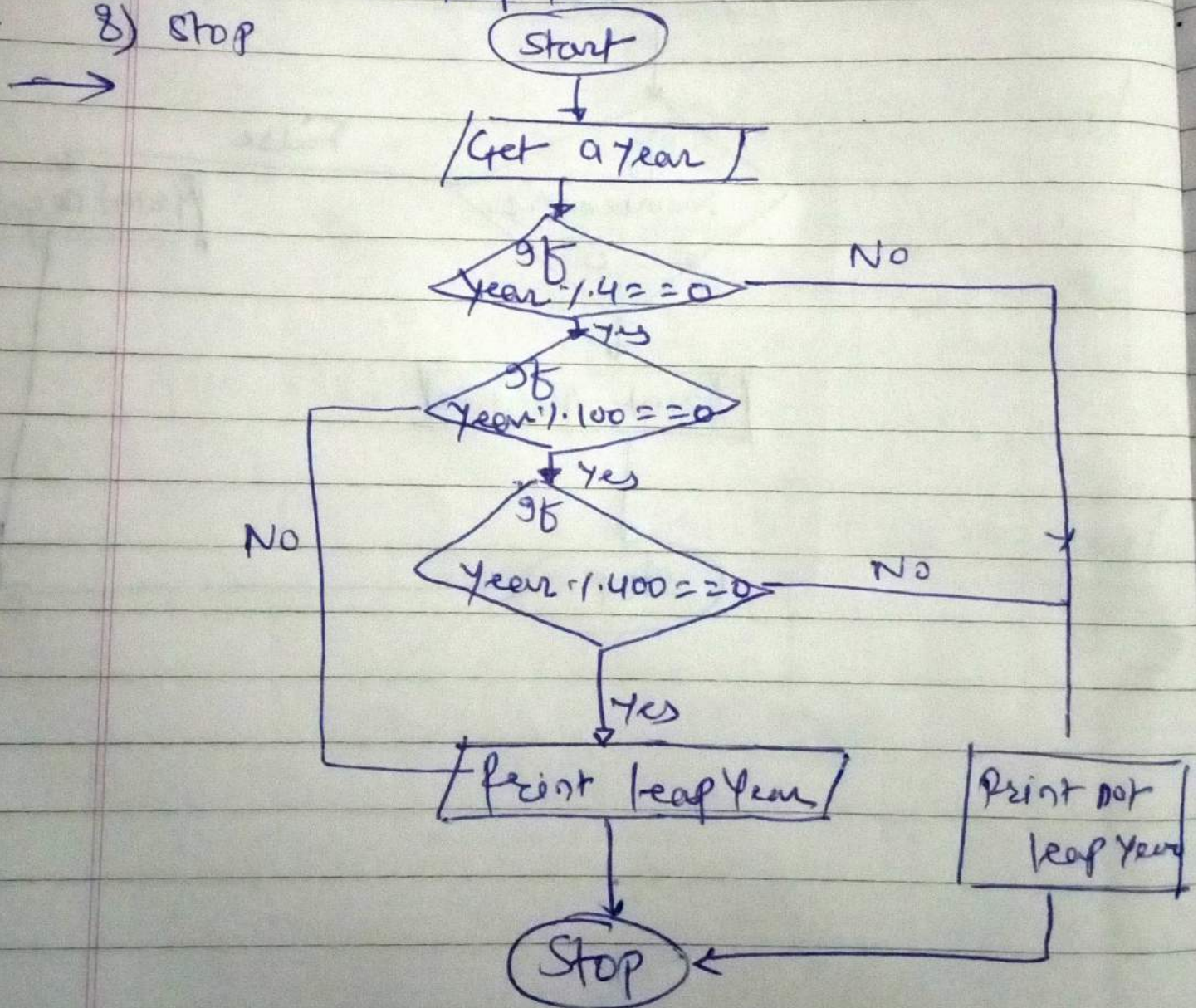   Else go to step 7
4) check year divisible by 400, If true go to
   step 5   Else go to   step 6
5) check year divisible by 400, If true go to
   step 6, Else go to step 7
6) print leap year
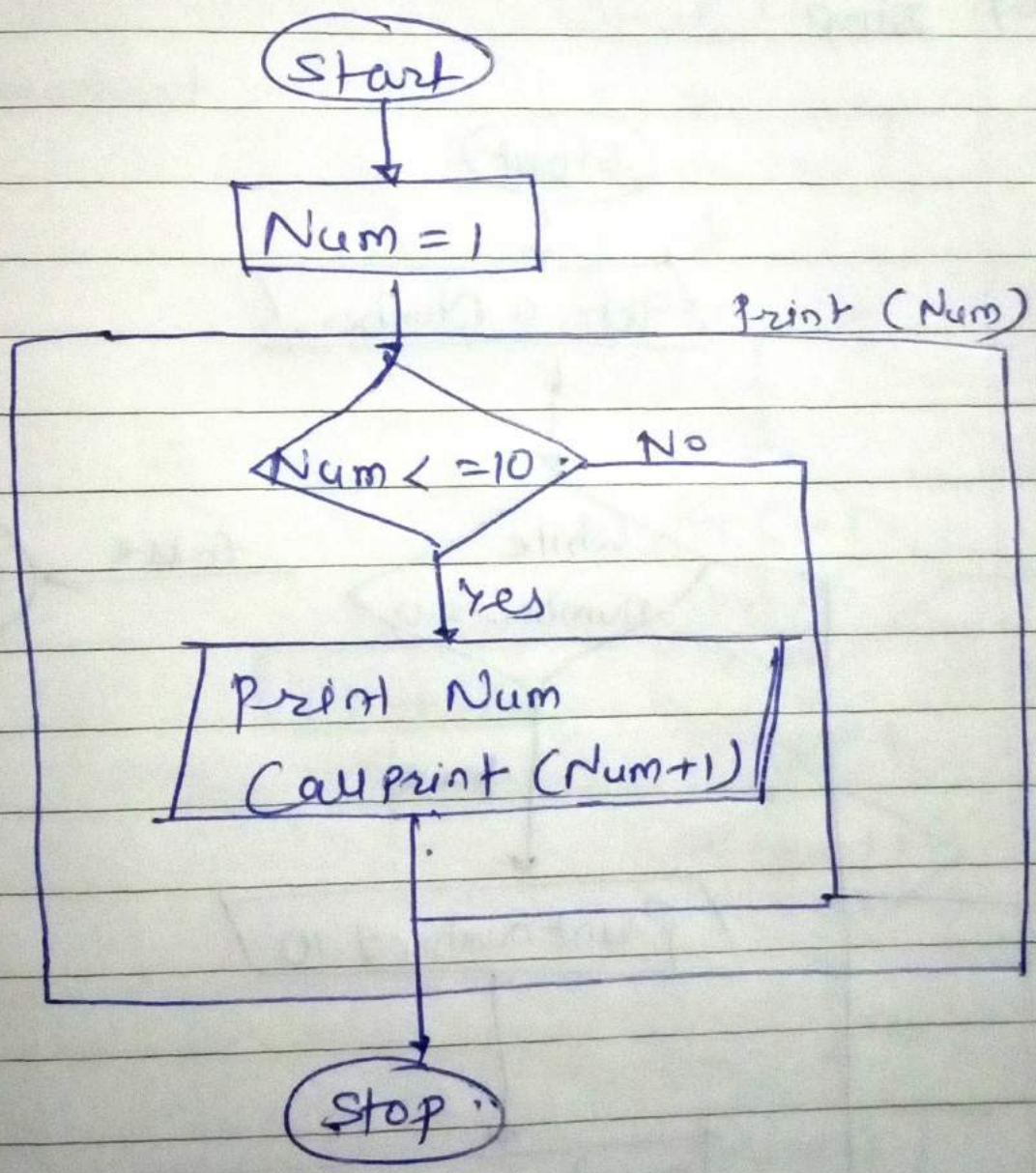7) print not leap year
8) stop

→

Q7) Print 1 to 10 without loop.

→ Start

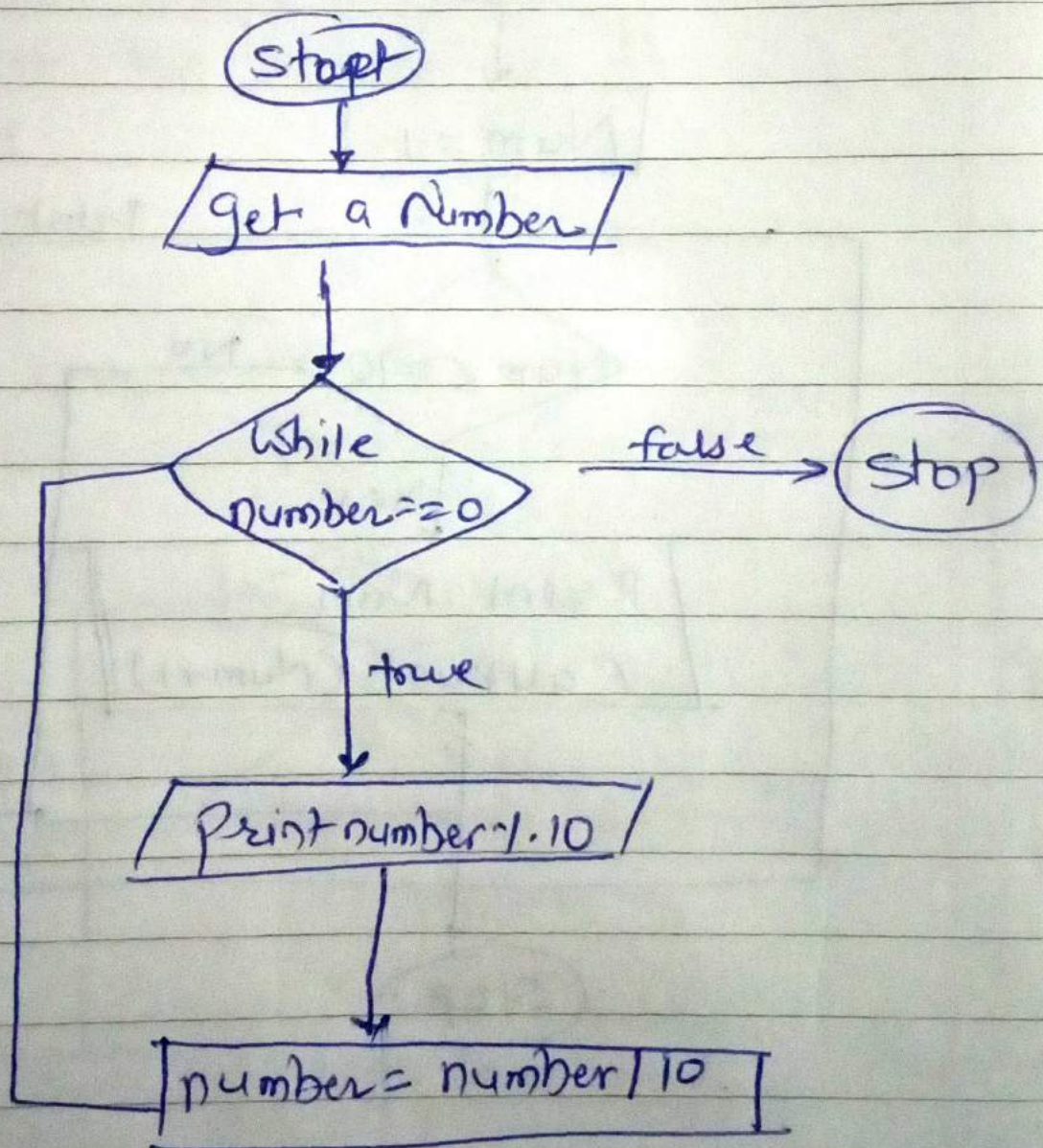2) Call print method

3) define a method print

   a. check num <= 10 If true print and recursively Call print method with num-1 else exit

4) Stop

```
              ( Start )
                 |
                 v
           +-----------+
           |  Num = 1  |
           +-----------+
                 |                          Print (Num)
                 v
            / Num <=10 ? \------ No ----+
            \            /               |
                 |                       |
                yes                      |
                 v                       |
        +----------------------+         |
        |   Print Num          |         |
        |   Call print (Num+1) |         |
        +----------------------+         |
                 |                       |
                 v                       |
              ( Stop )
```

Q8) Print digit of given number

→ Algorithm :-

1) Start
2) Get a number
3) Print the value of number % 10
4) Number = number/10;
5) Repeat Step 3 to 4 Until number is not equal to zero
6) Stop

```
         ( Start )
            |
            v
   / Get a Number /
            |
            v
         < While
         number == 0 >  --- false --->  ( Stop )
            |
          true
            |
            v
   / Print number % 10 /
            |
            v
 | number = number / 10 |
```
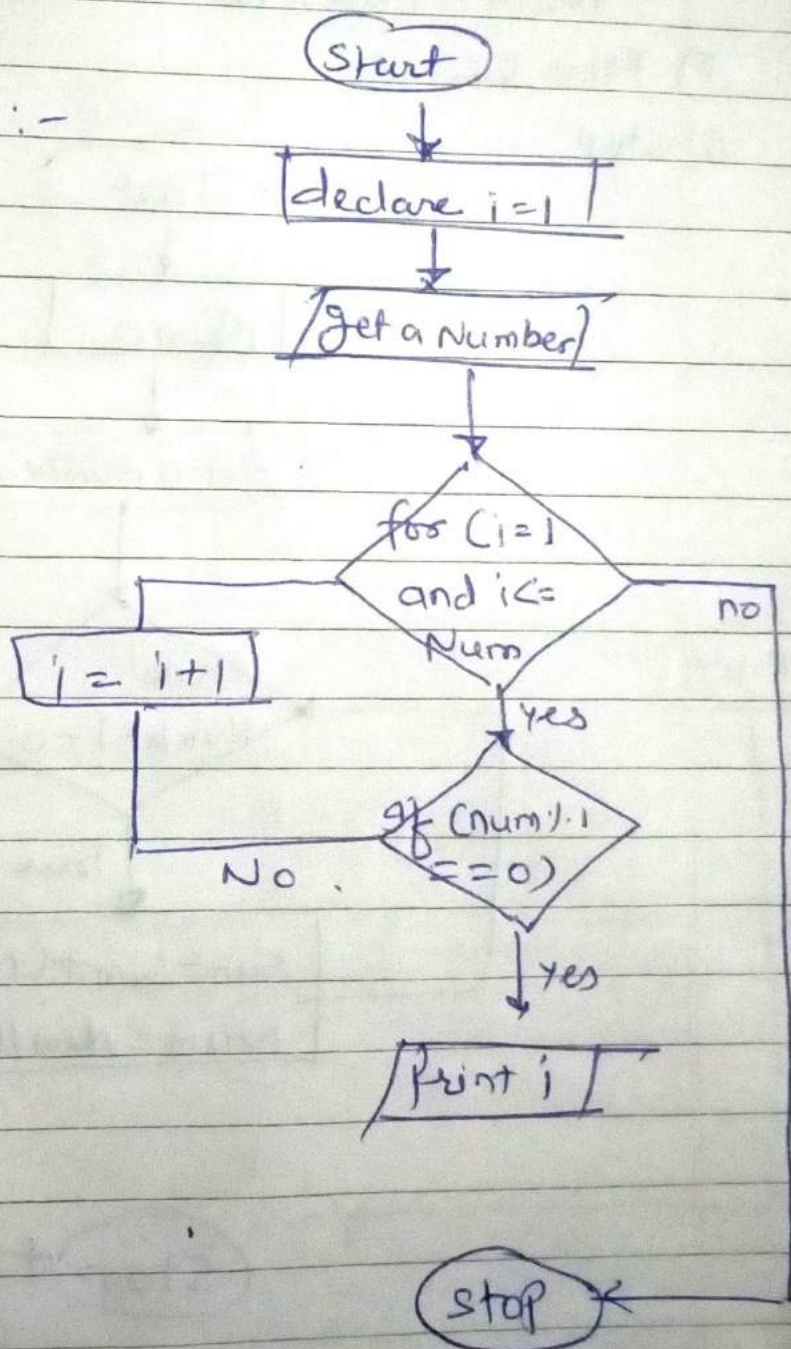
(09) Factor of given number

→1) Start

2) get a number

3) Declare i = 1

4) check number % i == 0 if true print i and increament the value of i

5) Repeat step 4 until <= number

6) Stop.

Flowchart :-
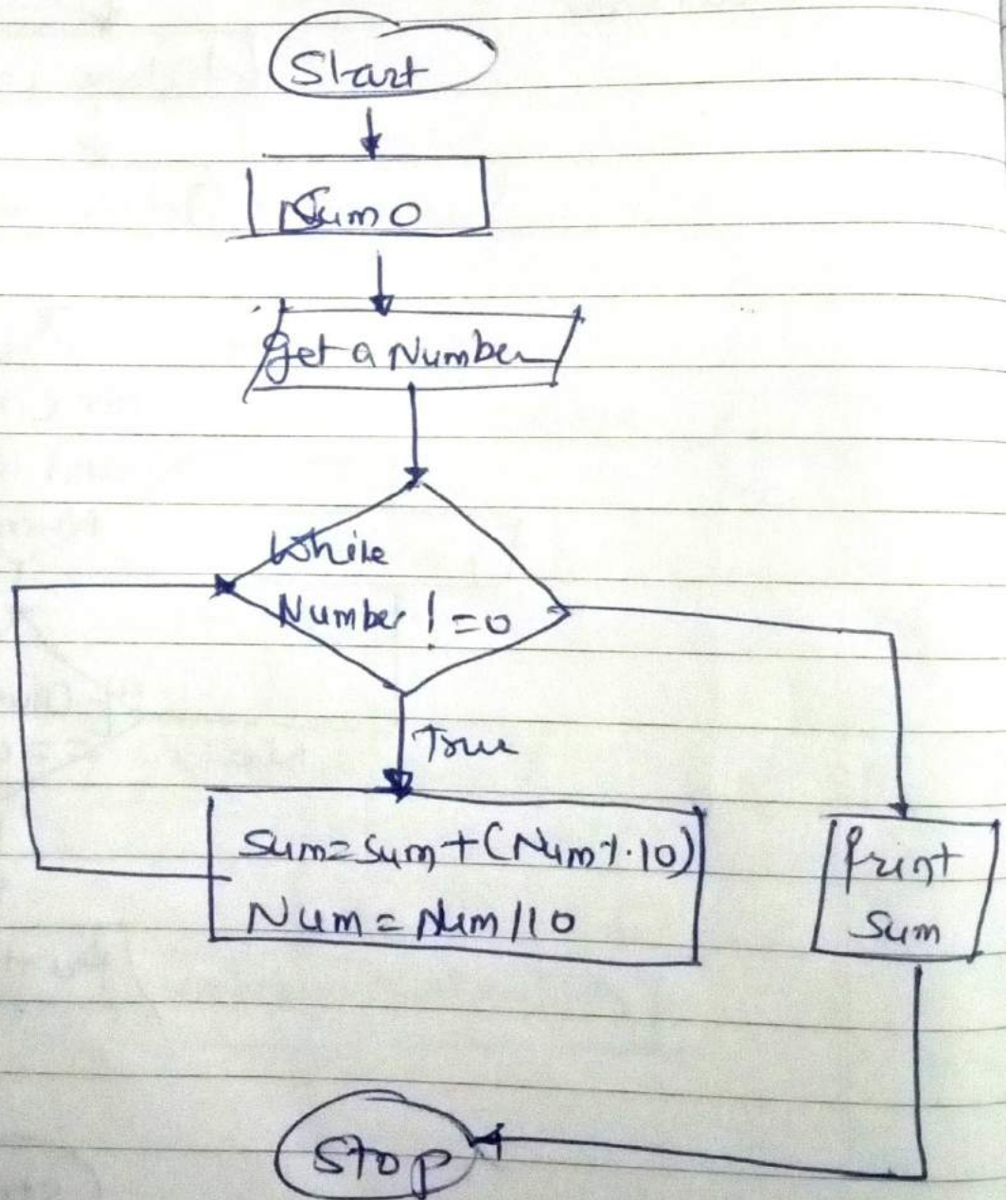
```
                        ( Start )
                            |
                            v
                    | declare i=1 |
                            |
                            v
                   / get a Number /
                            |
                            v
                          / \
                         /   \
            for (i=1    <     >  no
            and i<=      \   /
            Num           \ /
                           |
                          yes
                           |
                           v
                          / \
    | i = i+1 |          /   \
        ^              if (num%i
        |       No.     == 0)
        |               \   /
        |_____ \ /
                           |
                          yes
                           v
                     / Print i /
                           |
                           v
                       ( stop )
```

Q10) 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

Start

Sum 0

Get a Number

while
Number != 0

True

Sum = Sum + (Num % 10)
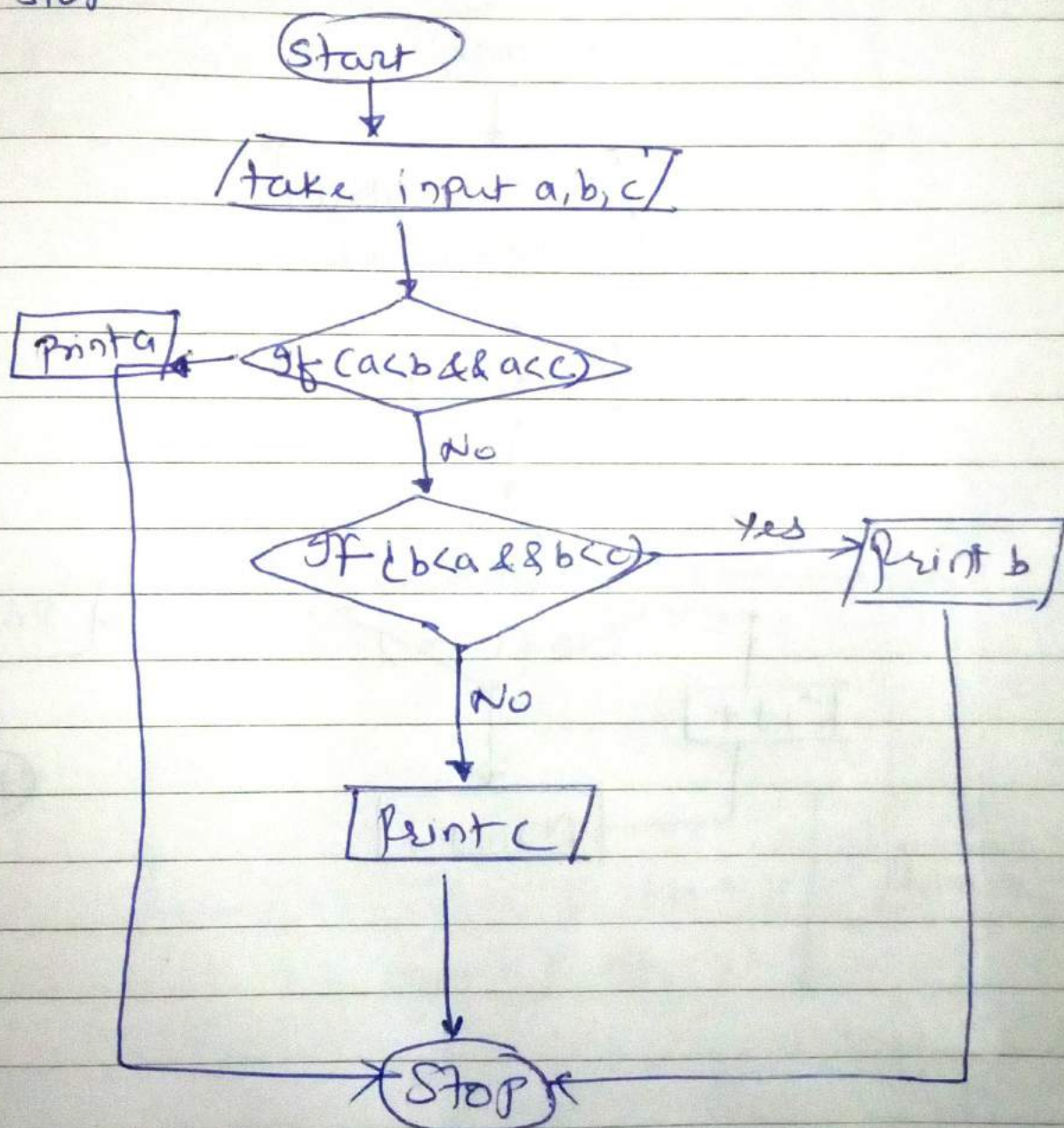Num = Num / 10

Print Sum

Stop

(Q11) Smallest of three Number

→ 1) start

2) get three number from user

3) check if a<b and a<c, If true Print a & exit else go to Stop 4

4) check if b<a and b<c If true Print b and exit else goto stop 5

5) Print c

6) stop

```
                    ┌─────────┐
                    │  Start  │
                    └─────────┘
                         │
                         ▼
              ┌──────────────────────┐
              │  take input a,b,c    │
              └──────────────────────┘
                         │
                         ▼
  ┌─────────┐      ◇─────────────────◇
  │ Print a │◄─────  If (a<b && a<c)
  └─────────┘      ◇─────────────────◇
       │                  │ No
       │                  ▼
       │           ◇─────────────────◇   Yes    ┌─────────┐
       │             If (b<a && b<c)  ─────────► │ Print b │
       │           ◇─────────────────◇          └─────────┘
       │                  │ No                        │
       │                  ▼                           │
       │           ┌────────────┐                     │
       │           │  Print c   │                     │
       │           └────────────┘                     │
       │                  │                           │
       │                  ▼                           │
       │            ┌──────────┐                      │
       └───────────►│   Stop   │◄─────────────────────┘
                    └──────────┘
```

Q7 Addition without arithmetic operator

→ 1) start
2) get two number
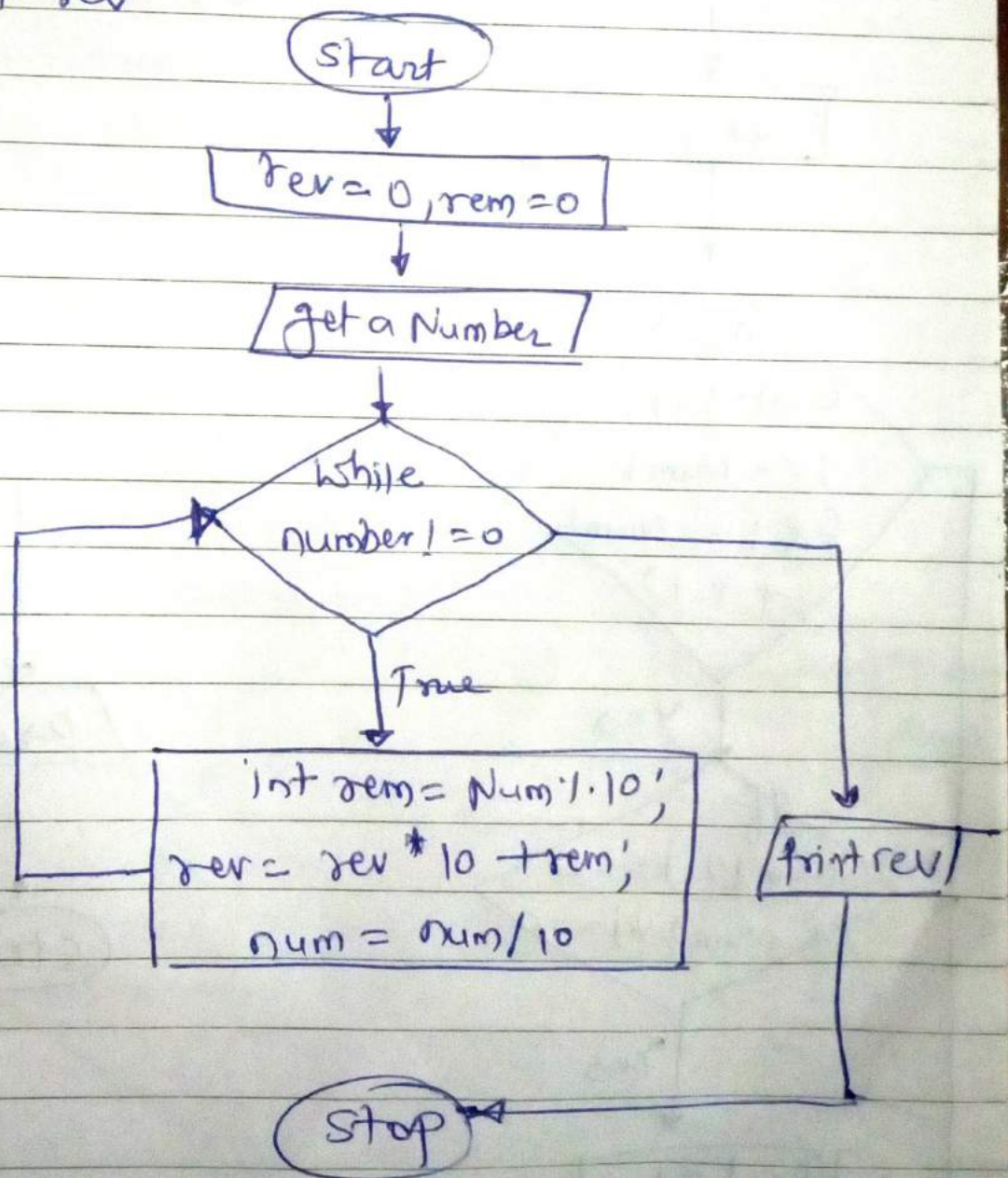3) Call add num(Num1, Num2) method
4) for (i=1; i<= Num2; i++)
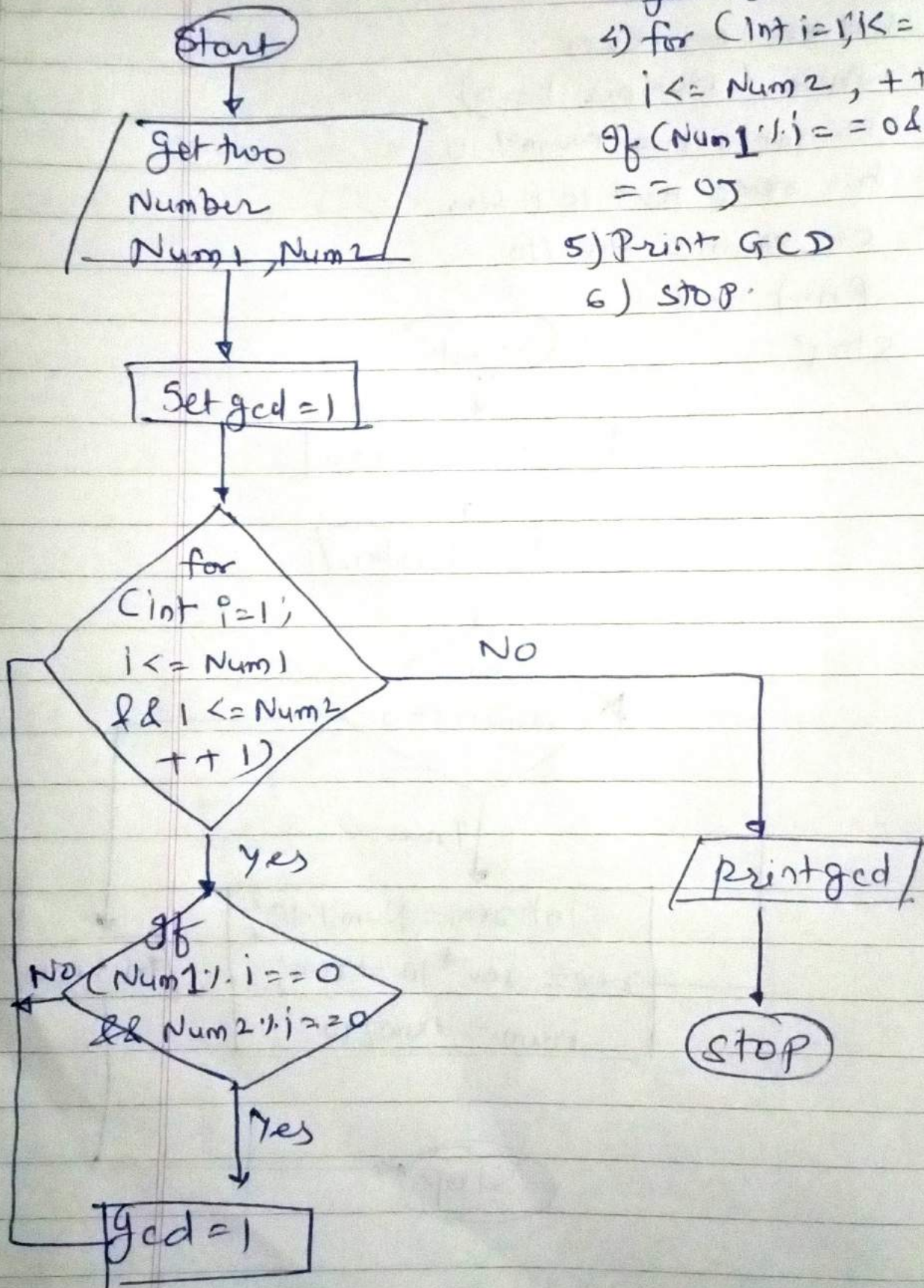   a Num1++
5) Print Sum
6) Stop

```
        ( Start )
            |
            v
  / Get two Numbers /
  /  Num1, Num2    /
            |
            v
    [ set i=1 ]
            |
            v
        <While
         i<= Num2>  ──────────>  [ Print Sum ]
        /        \                     |
  [ i++ ]       v                      v
     ^    [ Num++ ]                 ( Stop )
     |_____|
```
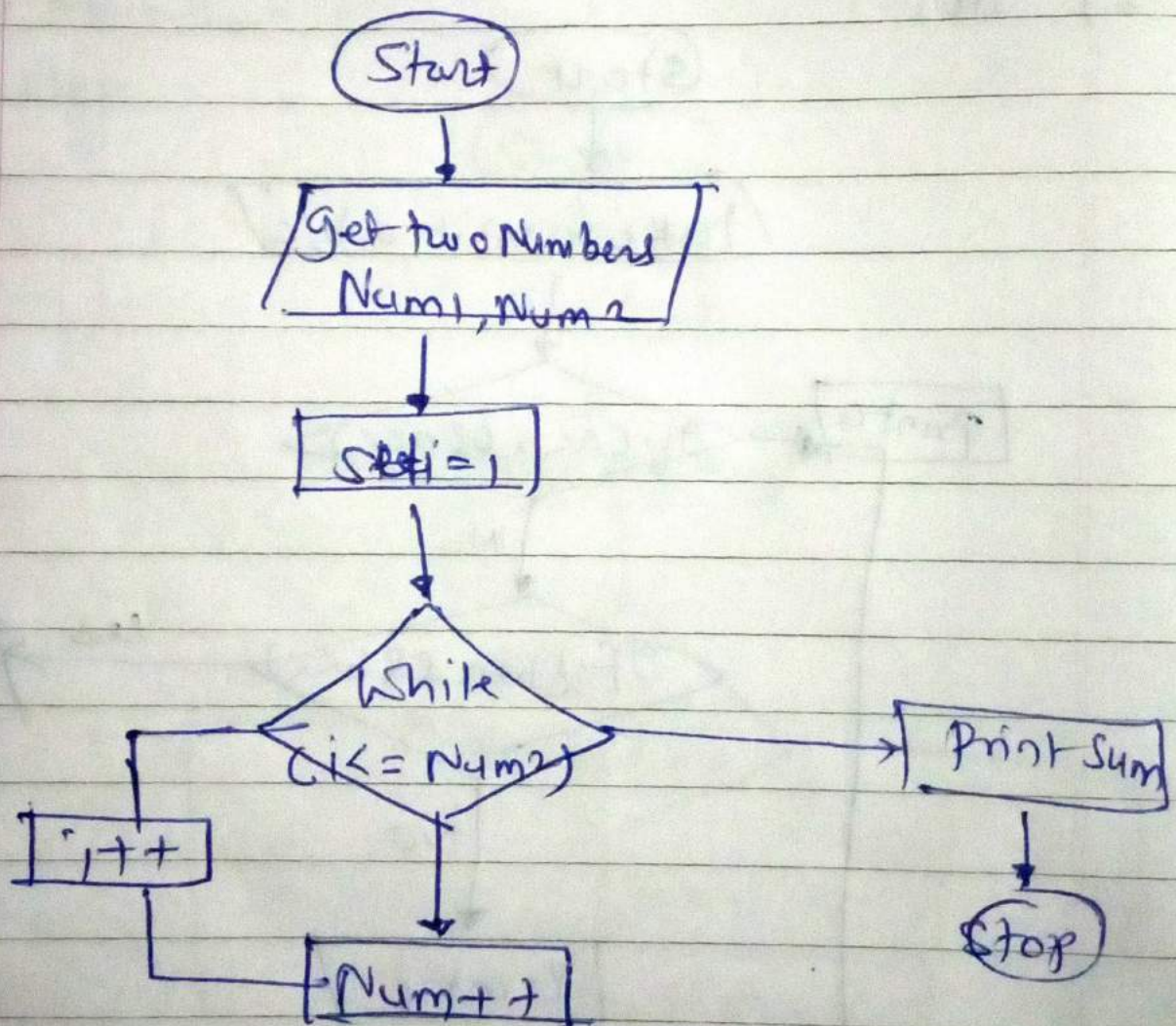
13) Reverse a given Number

→ Algorithm.

1) Start
2) Get a number
3) set rem=0, rev=0
4) While (Number !=0)
   a. int rem = Num % 10
   b. rev = rev * 10 + sem
   c. Num = num/10
5) Print rev
6) Stop

```
        ( Start )
            ↓
   | rev = 0, rem = 0 |
            ↓
    / get a Number /
            ↓
      < While
        number! = 0 >
            |
          True
            ↓
   | int rem = Num % 10;        |        / print rev /
   | rev = rev * 10 + rem;      |
   | num = num/10               |
            
        ( Stop )
```

Q14) GCD of two number

1) start
2) get two no. Num1, Num2
3) get gcd=1
4) for (int i=1; i<= Num1 &&
   i <= Num2, ++i)
   if (Num1%i == 0 && Num2%i
   == 0)
5) Print GCD
6) stop.

Start
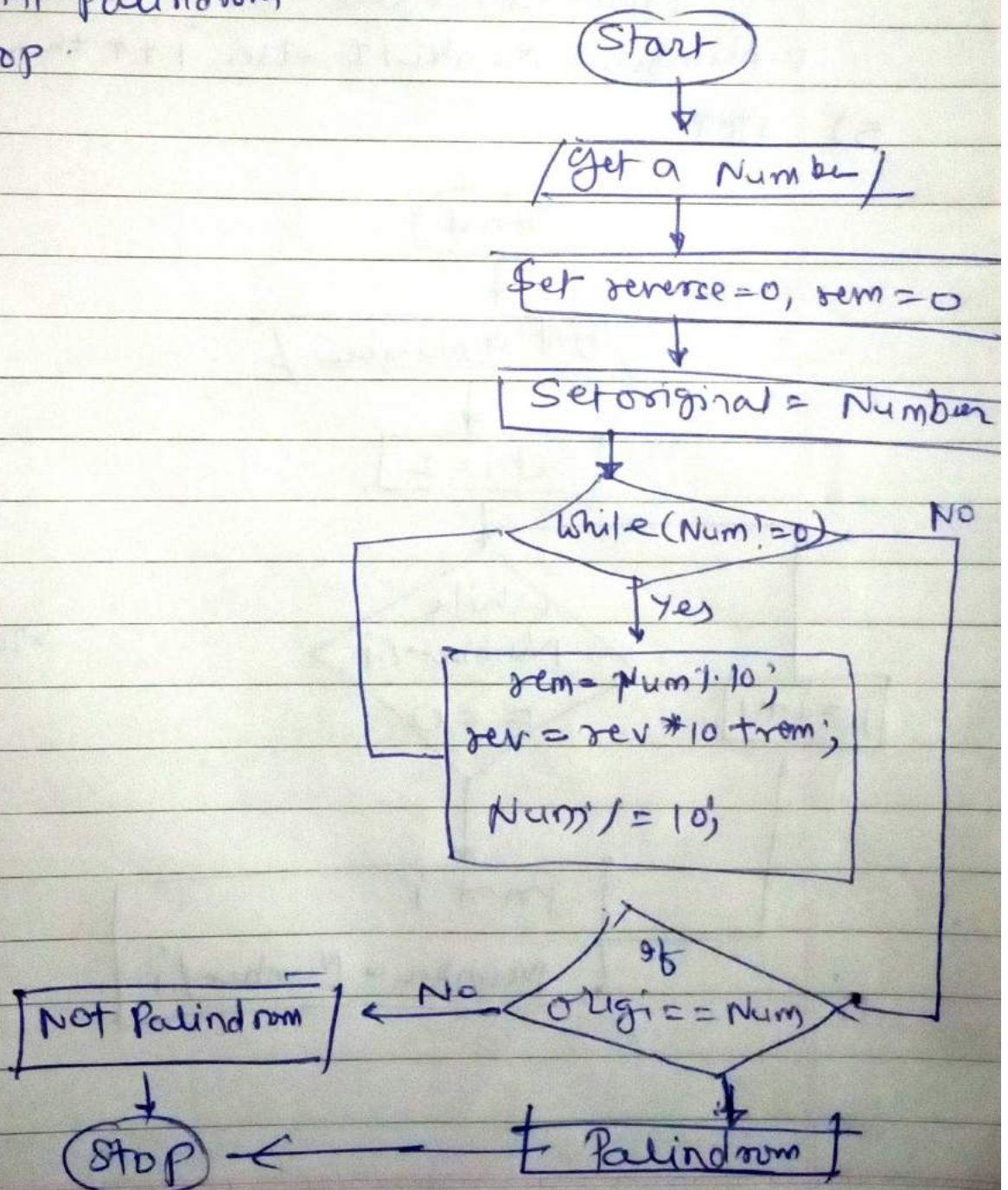
Get two
Number
Num1 , Num2

Set gcd=1

for
(int i=1;
i<= Num1
&& i <= Num2
++ i)

NO

yes

if
No (Num1%.i == 0
&& Num2%.j==0

Print gcd

stop

yes

gcd=1

Q 15) Lcm of two Numbers

Ans Alogorithm:-
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)
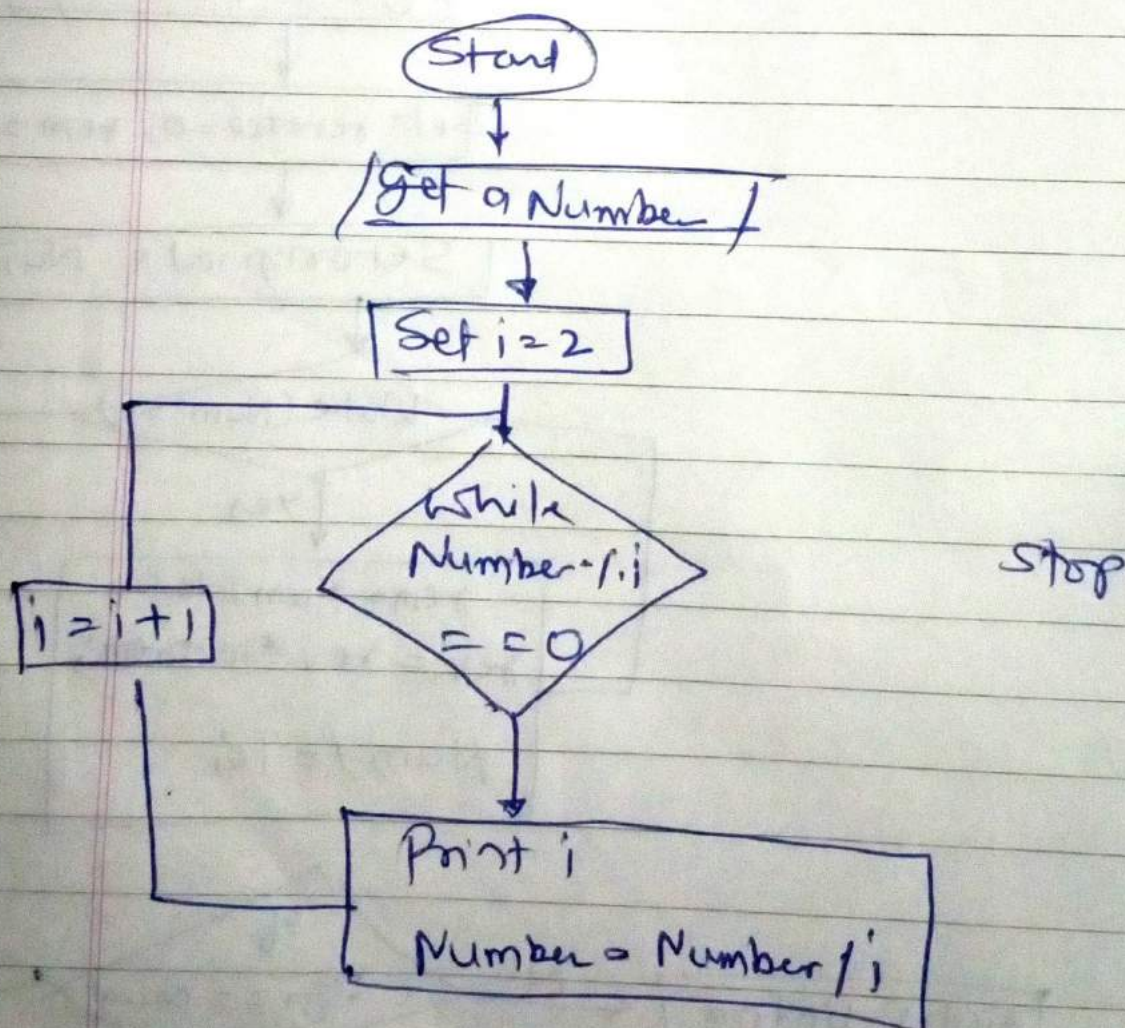5) lcm = (Num 1 * Num2) / gcd
6) Print Lcm
7) Stop
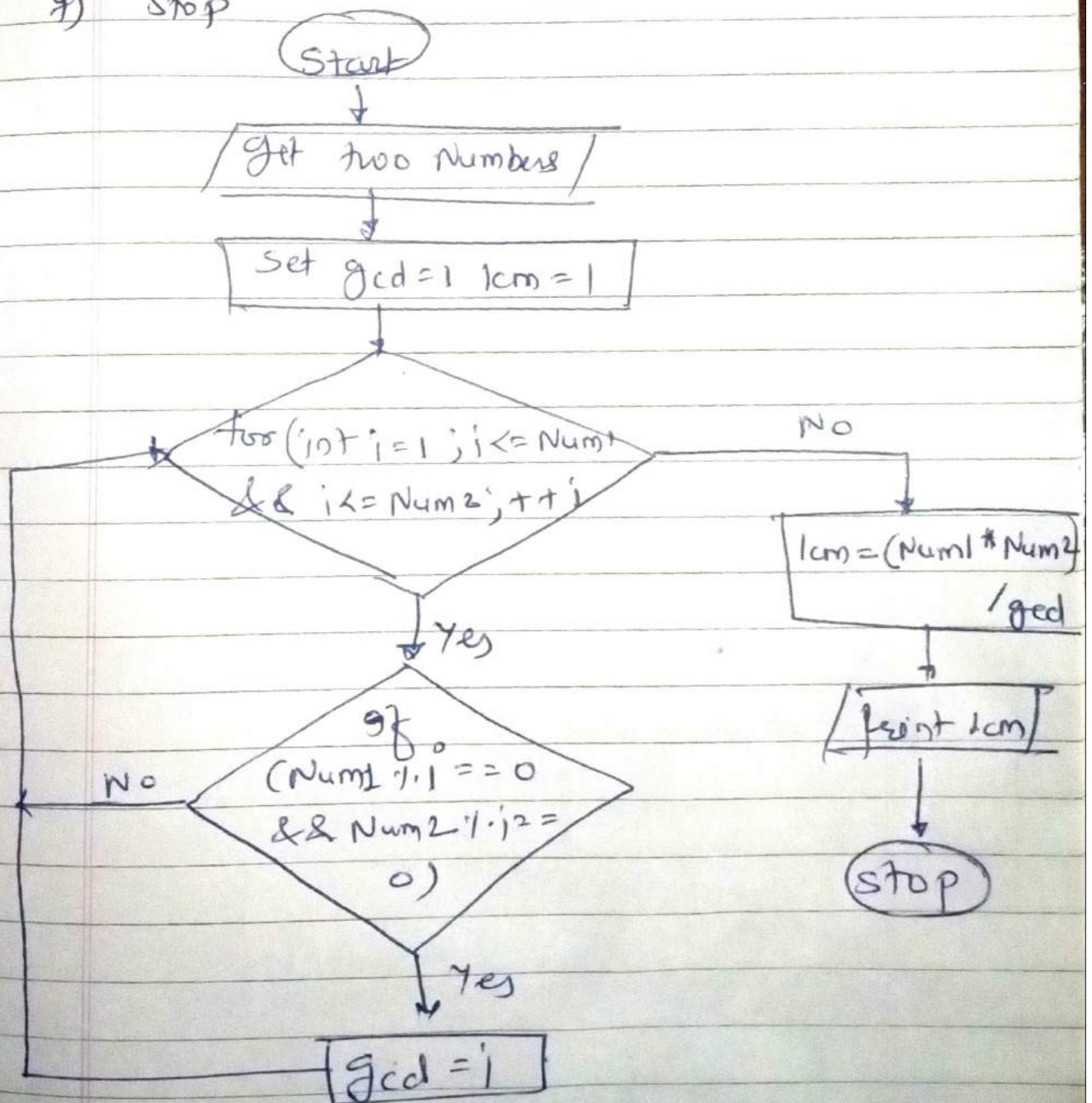
17) Check Palindrome or not
→ 1) Start
2) Get a Number
3) reverse = 0 & rem = 0
4) ~~check~~ Set original = Number
5) Check number != 0 If true go to 5 else goto 7
6) rem = Num % 10;
   rev = rev * 10 + rem;
   Num /= 10;
7) check if original == number If true
   Print Palindrom
8) Stop.

Start

Get a Number

Set reverse = 0, rem = 0

Set original = Number

While (Num != 0)   NO
   Yes

   rem = Num % 10;
   rev = rev * 10 + rem;
   Num /= 10;

if
origi == Num

Not Palindrom   ← No

STOP   ←   Palindrom

19) 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^n If (Num % i == 0)
   c. If it is true enter in bracket.
   d. Print (i) value on terminal
   e. Number = Number / i else i++ then loop will
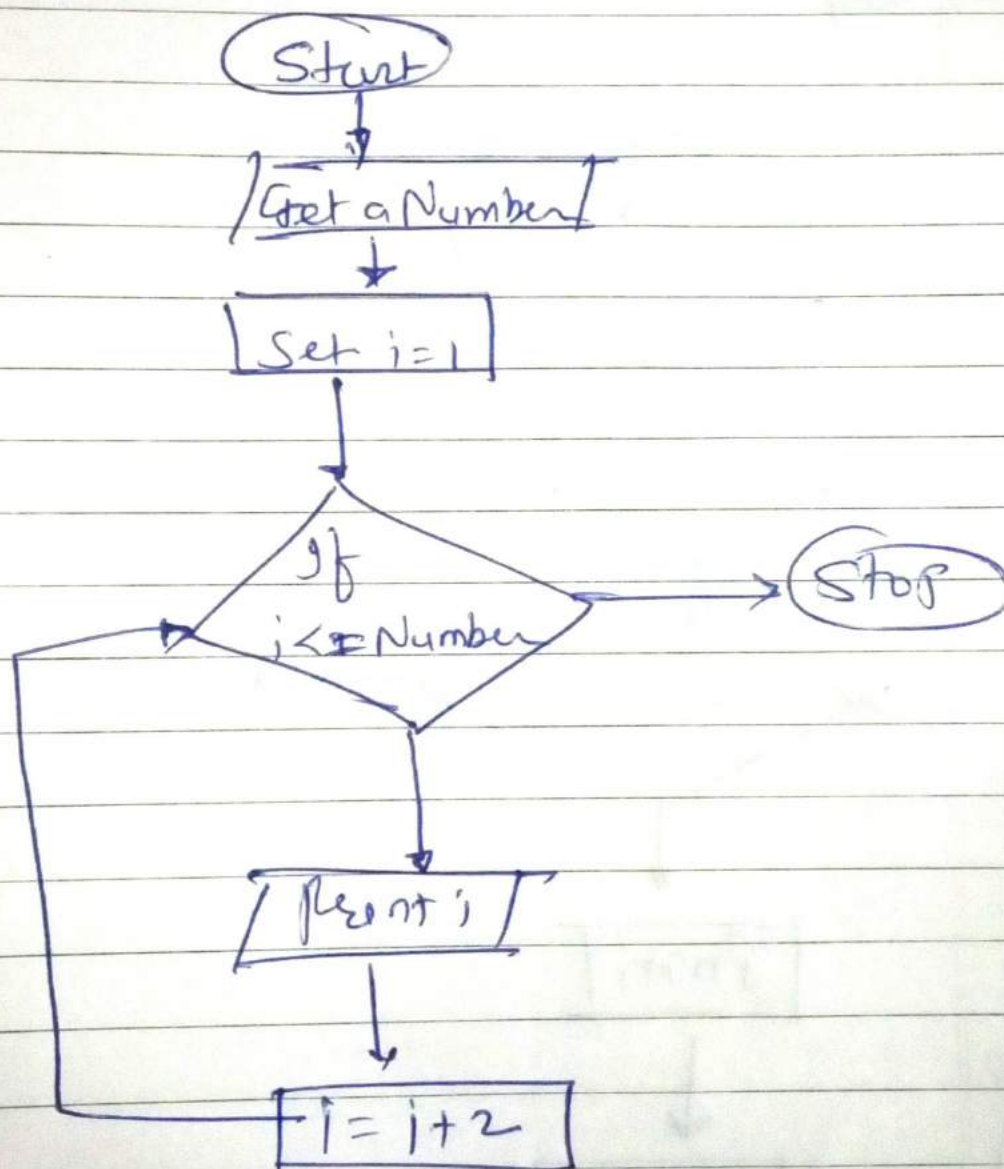
5) Stop

```
                    ( Start )
                       |
                       ↓
              / Get a Number /
                       |
                       ↓
                  [ Set i = 2 ]
                       |
                       ↓
                  < While
                    Number % i          Stop
                    == 0 >
                       |
                       ↓
   [ i = i + 1 ]   [ Print i
                     Number = Number / i ]
```

19) Even Series

→ Start

2) get a Number

3) set i = 2

3I) i <= Number print i & i = j+2

5) repeat step 4 Until i <= Number

6) Stop

20) Odd series
→   Algorithm:
1) Start
2) get a Number
3) Set i=1
4) If i<=Number print i & i=i+2
5) repeat Step 4 Until i<=Number
6) Stop

```
        ( Start )
            ↓
    / get a Number /
            ↓
     | Set i=1 |
            ↓
          / If \
         / i<=Number \ ——————→ ( Stop )
          \        /
            ↓
      | Print i |
            ↓
      | i=i+2 |
```