

3

II. Check the given number is Even or Odd

4

5

HPI: Take input

HPI: take mod

End number $1, 2 \neq 0$

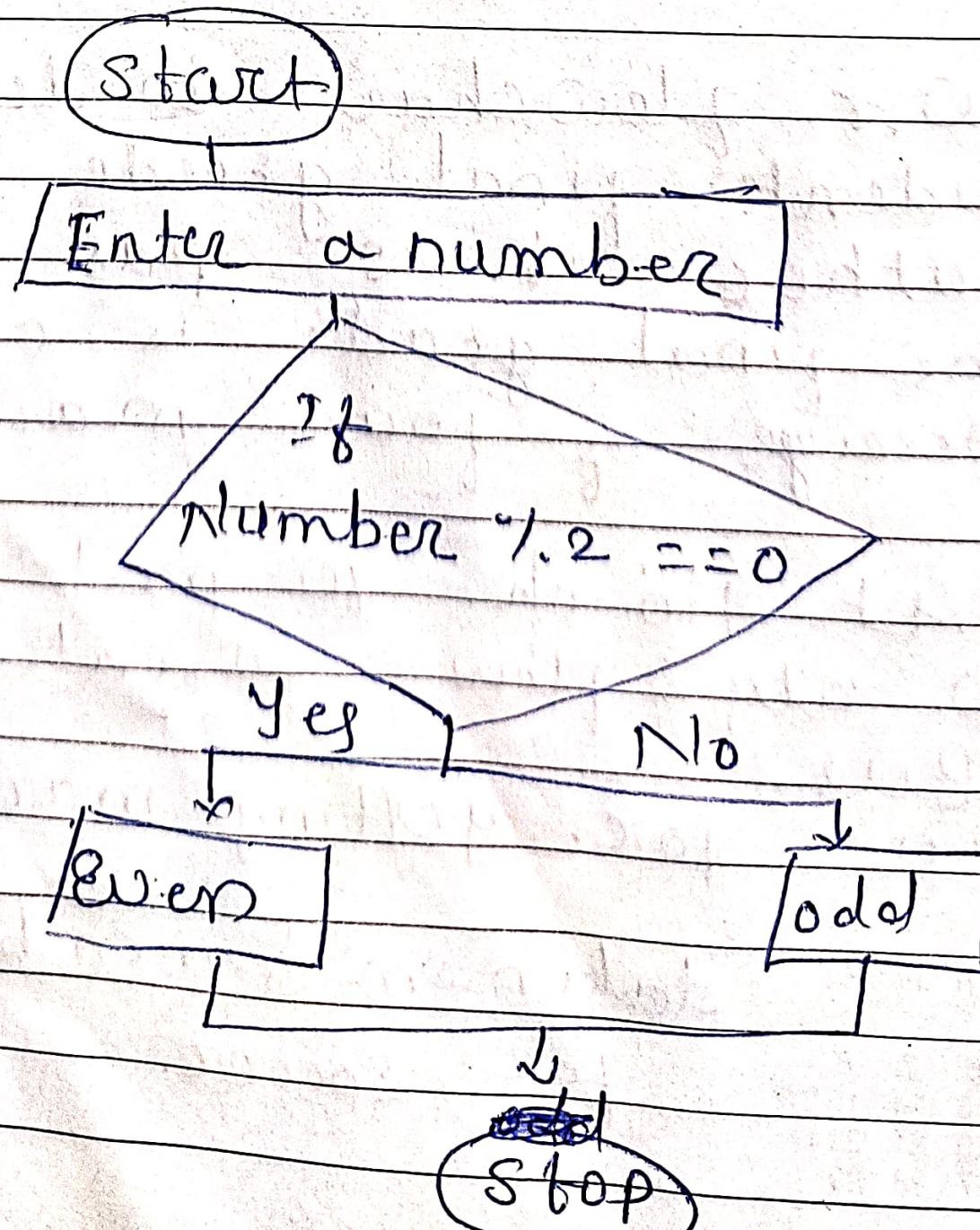
If $\text{mod} = 0$

Print "Even"

Else

23 Sunday

Print "Odd"



FEBRUARY
2020

Week 09 | 055-311

24
Monday

a w.r.t find grade.

(Start)

| Input P, C, M, B |

$$A = \frac{(P + C + M + B)}{4} \cdot 100$$

Is $A \geq 85$?

Yes | Print A |

end

No

Is $A \geq 75$?

Yes | Print B |

end

No

Is $A \geq 65$?

Yes | Print C |

end

Fail

end

FEBRUARY

2020

25

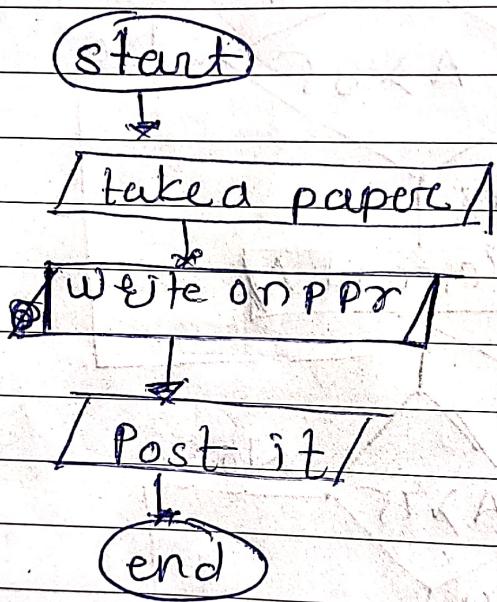
056-310 | Week 09

FEBRUARY
2020

Tuesday

- 9 Step 1: start
- Step 2: take input P, C, M, B.
- 10 Step 3: take average of PCM_B
- Step 4: check the condition
- 11 Step 5: print result
- Step 6: End

12



- 5 Step 1: start
- Step 2: take a paper
- 6 Step 3: write a letter
- Step 4: put letter on post office
- 7 Step 5: end.

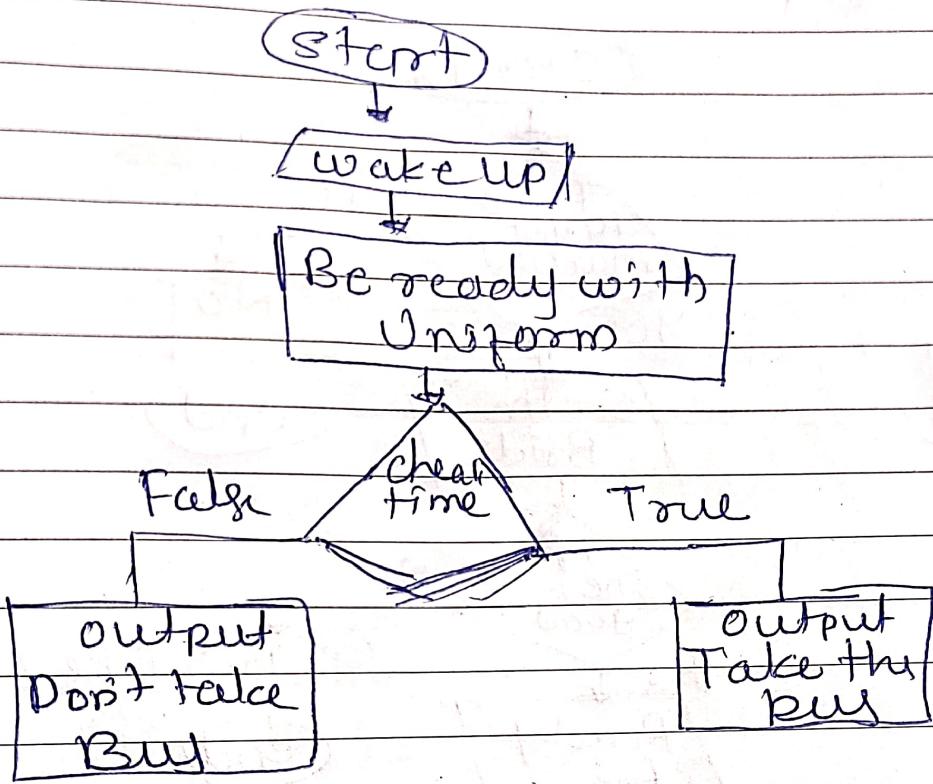
FEBRUARY
2020

Week 09 | 057-309

26

Wednesday

CBS



Algorithm

Step 1: Start

Step 2: wake up early

Step 3: Ready with all your stuff

Step 4: check time of bus
which is arrived.

Step 5: Output

Step 6: end

FEBRUARY 2020

Su	Mo	Tu	We	Th	Fr	Sa
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22

FEBRUARY
2020

27

058-308 | Week 09

Q4

Thursday

9

10

11

12

1

2

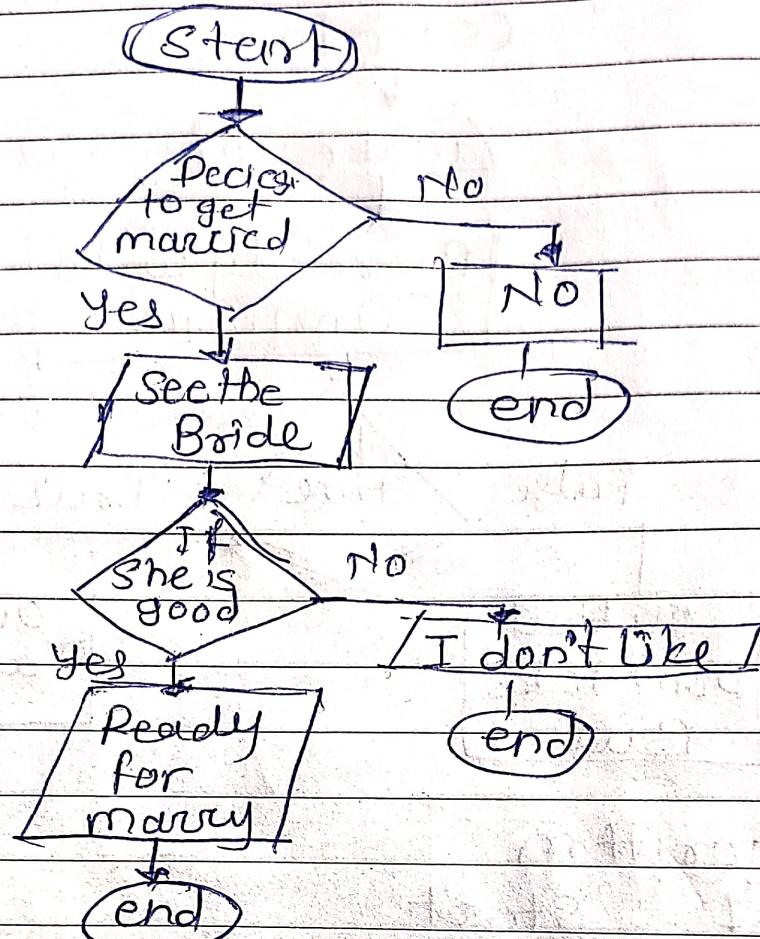
3

4

5

6

7



Algorithm

Step 1: Start

Step 2: if married is decision == True
 Print 'Yes'
else
 Print 'No'

Step 3: if Bride == good
 Print "Ready for marry"
else
 Print "I don't want marry."

Step 4 end.

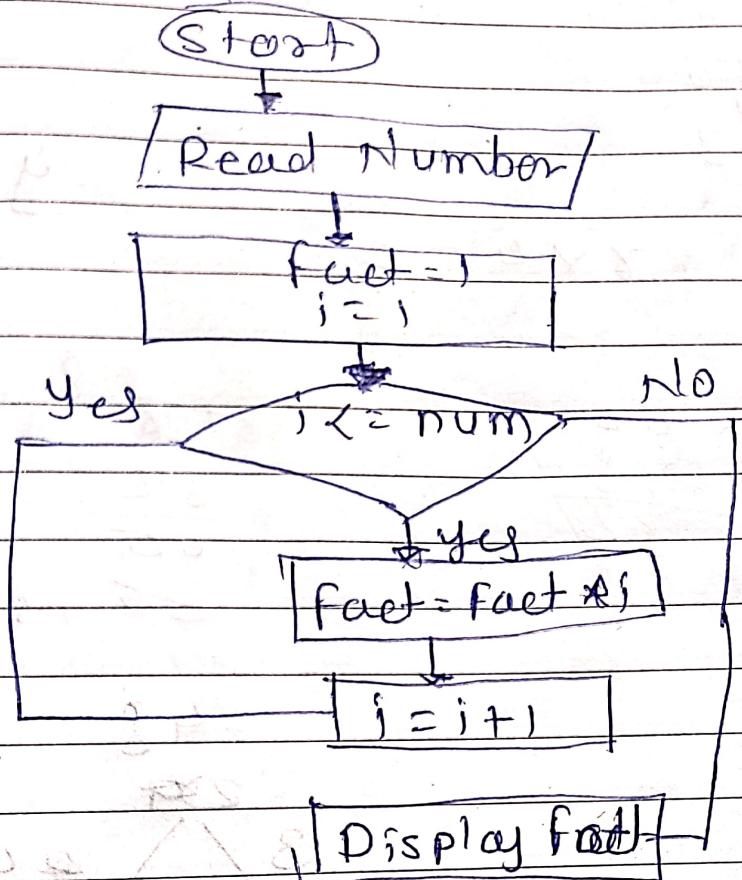
FEBRUARY
2020

Week 09 | 059-307

28

Friday

a Factorial



Algorithm

Step 1: Take input

Step 2: declare fact = 1 & i = 1

Step 3: check the

number is greater than equal

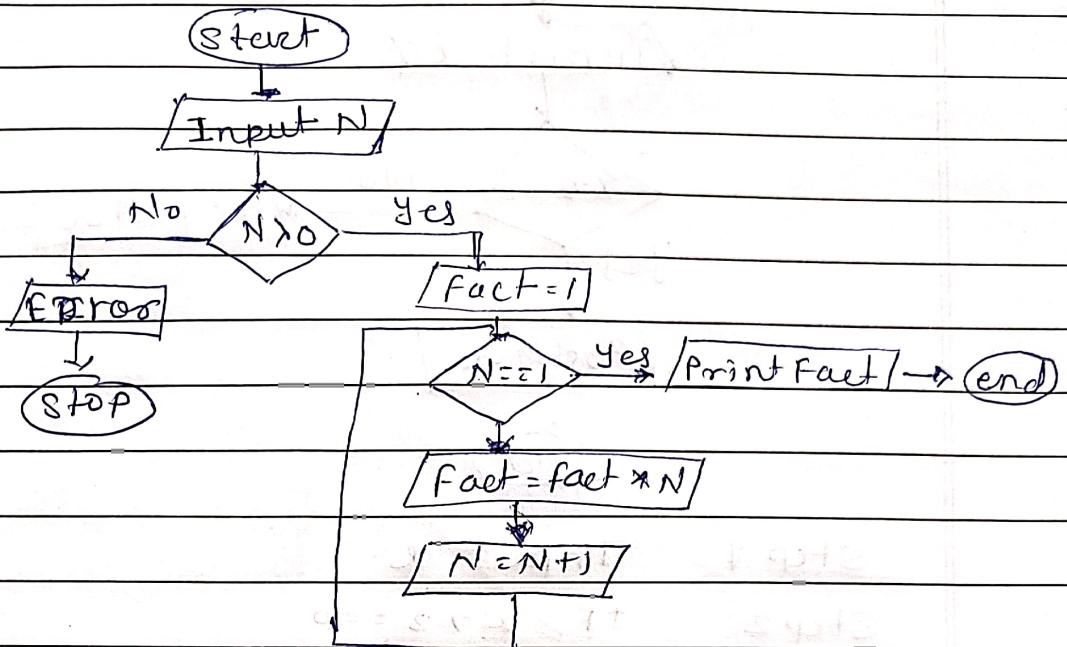
to 1

Step 4: Output

FEBRUARY 2020

Su	Mo	Tu	We	Th	Fr	Sa
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

a) find the factorial of a number using recursion.



algorithm

Step 1: Start

Step 2: Read number n; Step 3 call funct(n)

Step 4: Print Factorial R

Step 5: Stop

funct(n)

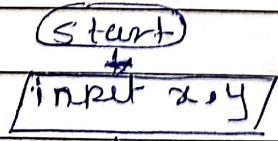
Step 1: If $n = 1$ then return 1

Step 2 - Else

$F = n * \text{factorial}(n-1)$

Step 3 : Return F

a) Swap two numbers without using third variable



Step 1: Start

Step 2: Enter x,y

Step 3: print x,y

Step 4: $x = x + y$

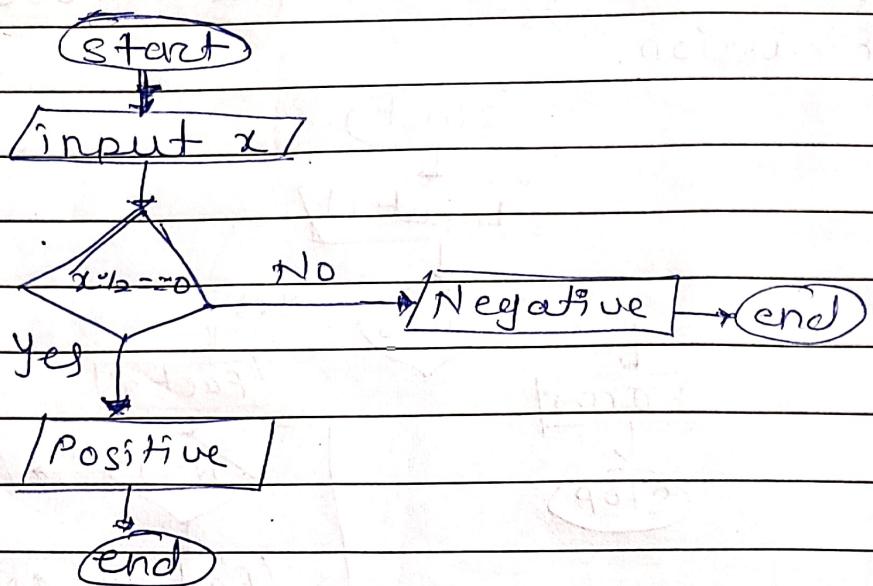
Step 5: $y = x - y$

Step 6: $x = x - y$

Step 7: Print x,y

Step 8: End.

Q5 How to check given number is positive or negative



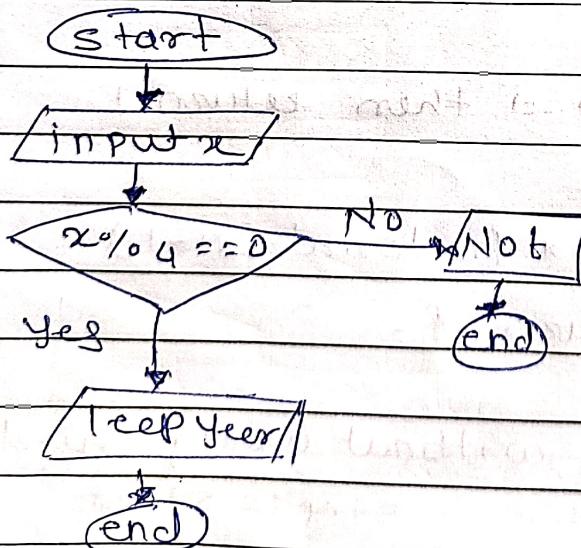
Step 1: Input x

Step 2: If $x \cdot 1.2 = 0$
Print "Positive"

Else
Print "Negative"

Step 3: end

Q6. Find whether given number is leap or not



Algo.

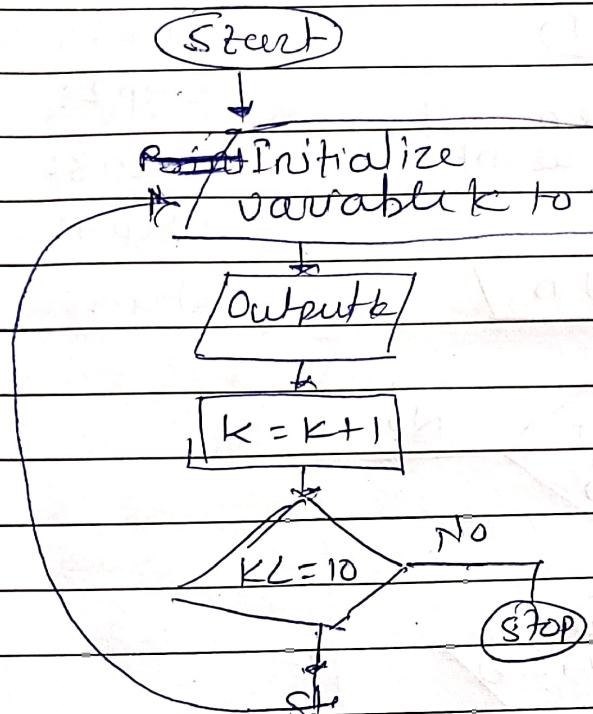
Step 1. Input x

Step 2: If $x \% 4 = 0$
Print "Leap"

Else
Print "Not"

Step 3: end.

Q7)



Step 1: Start

Step 2: Initialize k to 1

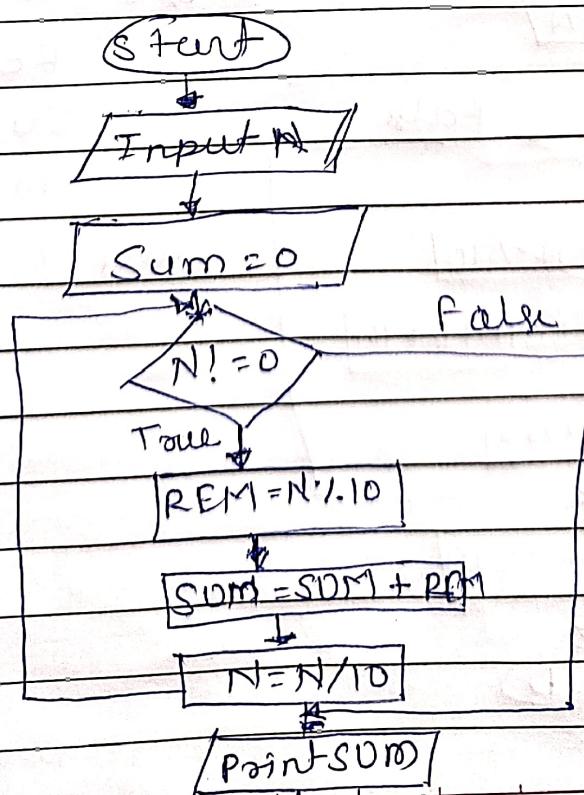
Step 3: Output k

Step 4: Increment k by 1

$$k = k + 1$$

Step 5: Stop

Q8.



Algorithm

Step 1: INPUT N

Step 2: SUM = 0

Step 3: while (N != 0)

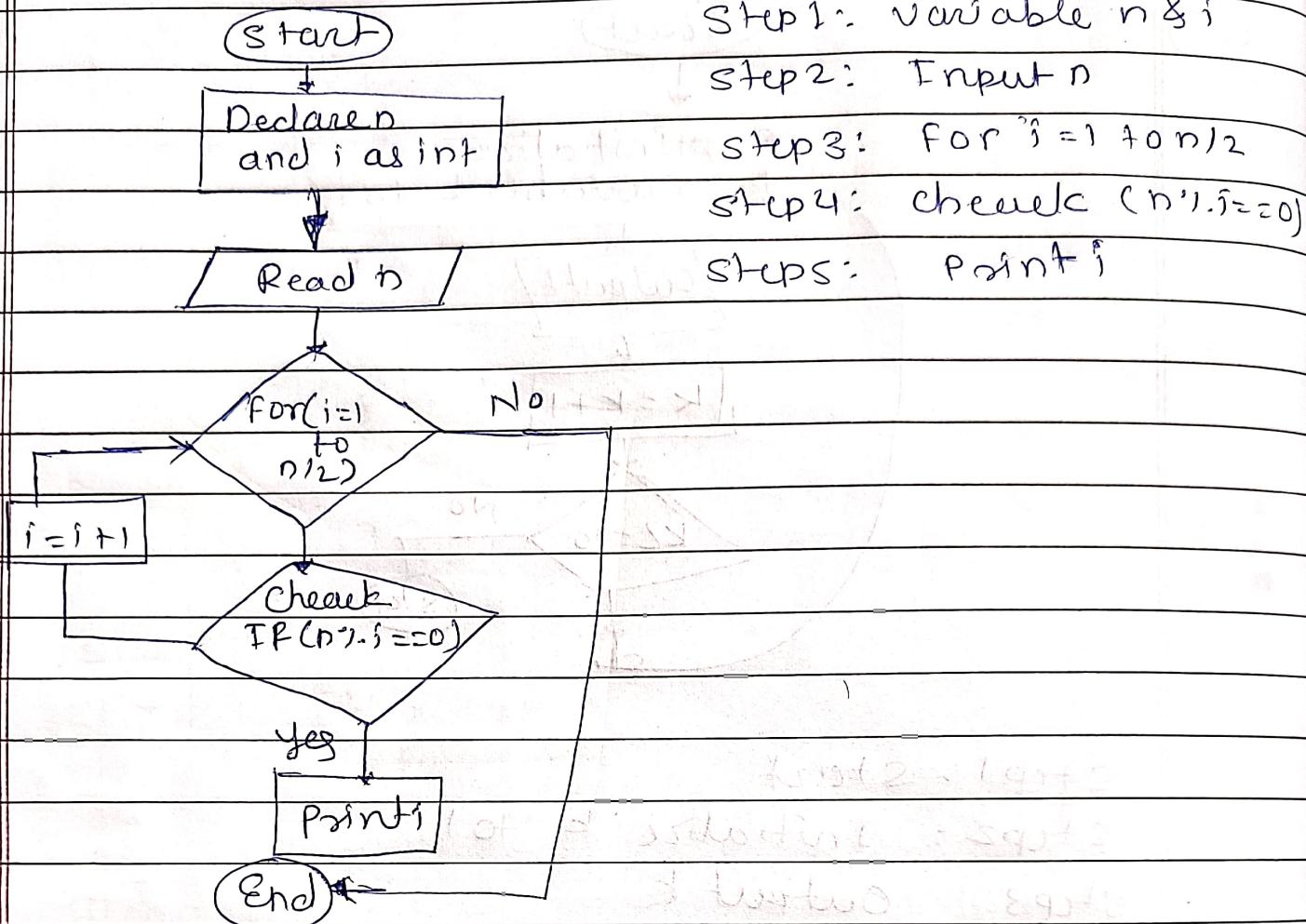
$$\text{Rem} = N \% 10$$

$$\text{SUM} = \text{SUM} + \text{Rem}$$

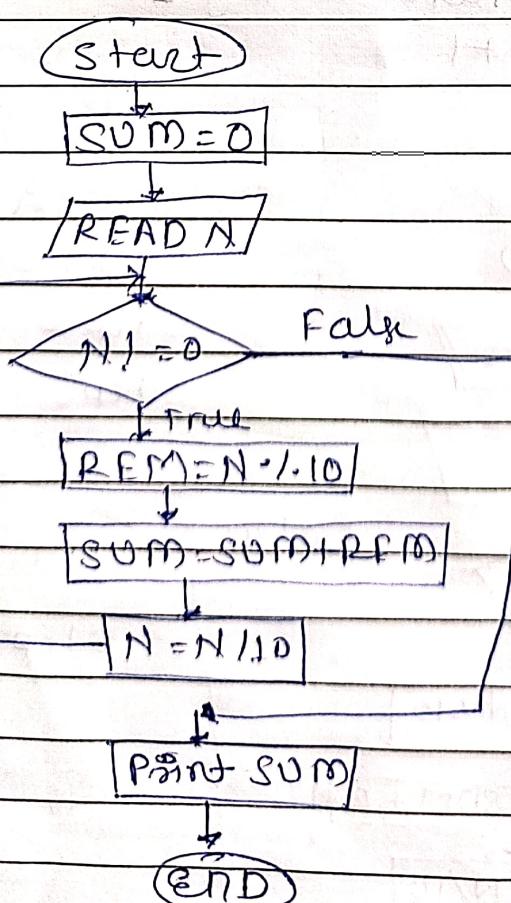
$$N = N / 10;$$

Step 4: Print SUM.

eg.



Q10



Step 1: variable n & i

Step 2: Input n

Step 3: For ?i = 1 to n/2

Step 4: check if (n-i,i=0)

Steps: Print i

Step 1: Input N

Step 2: Sum = 0

Step 3: while (N != 0)

Rem = N % 10;

Sum = Sum + Rem

N = N / 10

Step 4: Print sum

Q11.

Step 1: Start

Step 2: Read three num.

a, b, c

Step 3: If $a > b$ and $a > c$

then display "a is small"

If $b > a$ and $b > c$

then "b is small"

If $c > a$ and $c > b$

then "c is smallest"

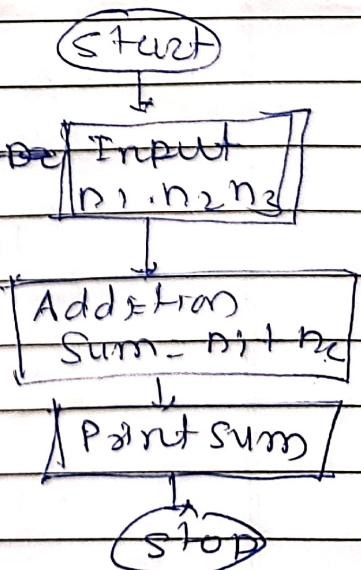
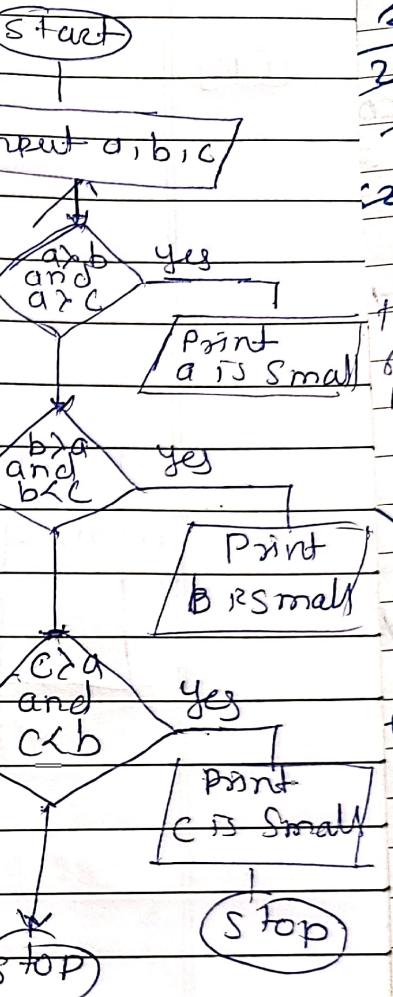
Step 4: Stop.

Q12 Step 1: Start

Step 2: Input n₁, n₂, n₃Step 3: Add n₁ and n₂

Step 4: Display Sum

Step 5: Stop



Algorithm

Q13 Start

Declare n, rev, rem

Read n

while
(n!=0) $n = n/10$
 $rem = n \% 10$
 $rev = rev * 10 + rem$
 + rem

Point reverse

End

Step 1: Declare n, rev, rem

Step 2: Take input

Step 3: while n not equal 0

 $rem = n \% 10$ $rev = rev * 10 + rem$ $n = n / 10$

Step 4: Print reverse

Q14 Start

Start

Declare variable

 $n_1, n_2, gcd = 1, i = 1$ Input n_1 & n_2 $i \leq n_1 \text{ & } i \leq n_2$

No

Yes

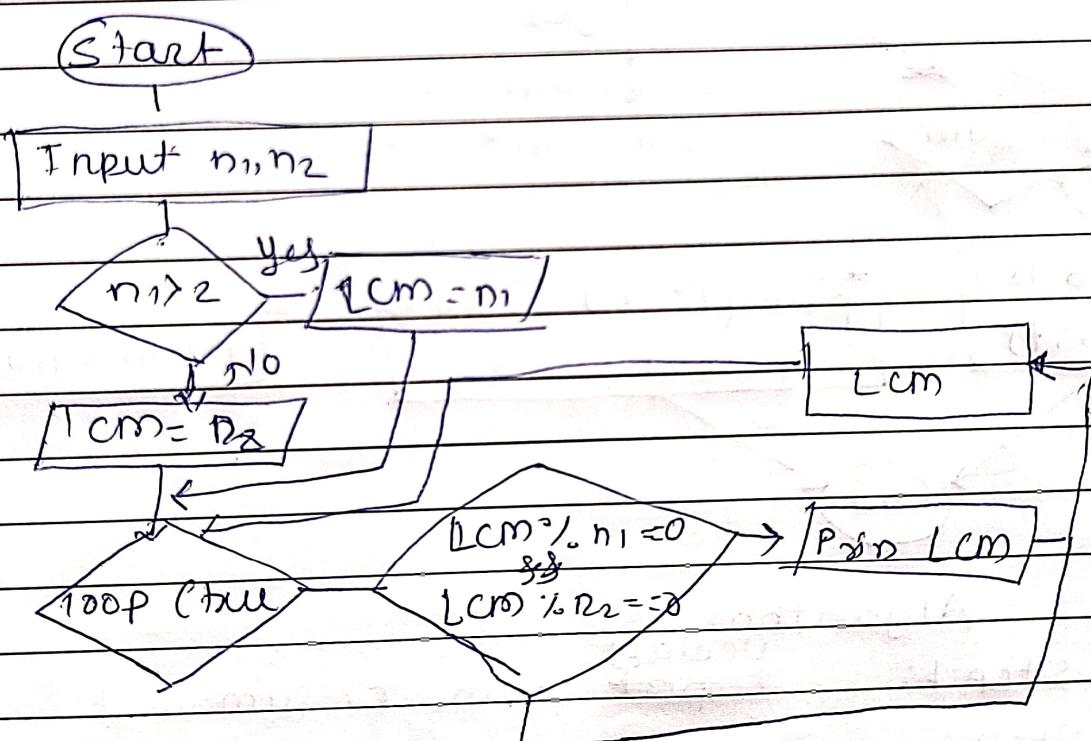
 $n_1 - i = 0$
 and
 $n_2 - i = 0$
 $gcd = i$ $i = i + 1$

Point gcd

End

- 201
- Step 1: Start
- Step 2: Declare vars $n_1, n_2, gcd = 1, i = 1$
- Step 3: Input $n_1 \& n_2$
- Step 4: Repeat $i \leq n_1 \text{ and } i \leq n_2$
 If $n_1 \% i == 0 \text{ and } n_2 \% i == 0$
 $gcd = i$
- Step 5: Print gcd .
- Step 6: Stop

Q15



Algorithm

Step 1: number n_1, n_2, LCM Step 2: Input n_1, n_2 Step 3: $LCM = (n_1 > n_2) ? n_1 : n_2$

Step 4: while true

 if $(LCM \% n_1 == 0 \text{ and } LCM \% n_2 == 0)$

print LCM

 $LCM += 1$

Step 5: stop

(Start)

16)

int num, r, sum = 0, t

```
printf("Input a number")
scanf("%d", &num);
```

t = num

num != 0

Yes

t == sum

r = num % 10

sum = sum * 10 + r

```
printf("%d is a
palindrome: t")
```

```
printf("%d is not a
palindrome: t")
```

num = num / 10

End

Algorithm

Declare

Step 1: Input num, r, sum = 0, t.

Step 2: Input num

Step 3: Store value of num to t.

Step 4: If num > 0

r = num % 10

sum = sum * 10 + r

num = num / 10.

Step 5: If t is equal to sum

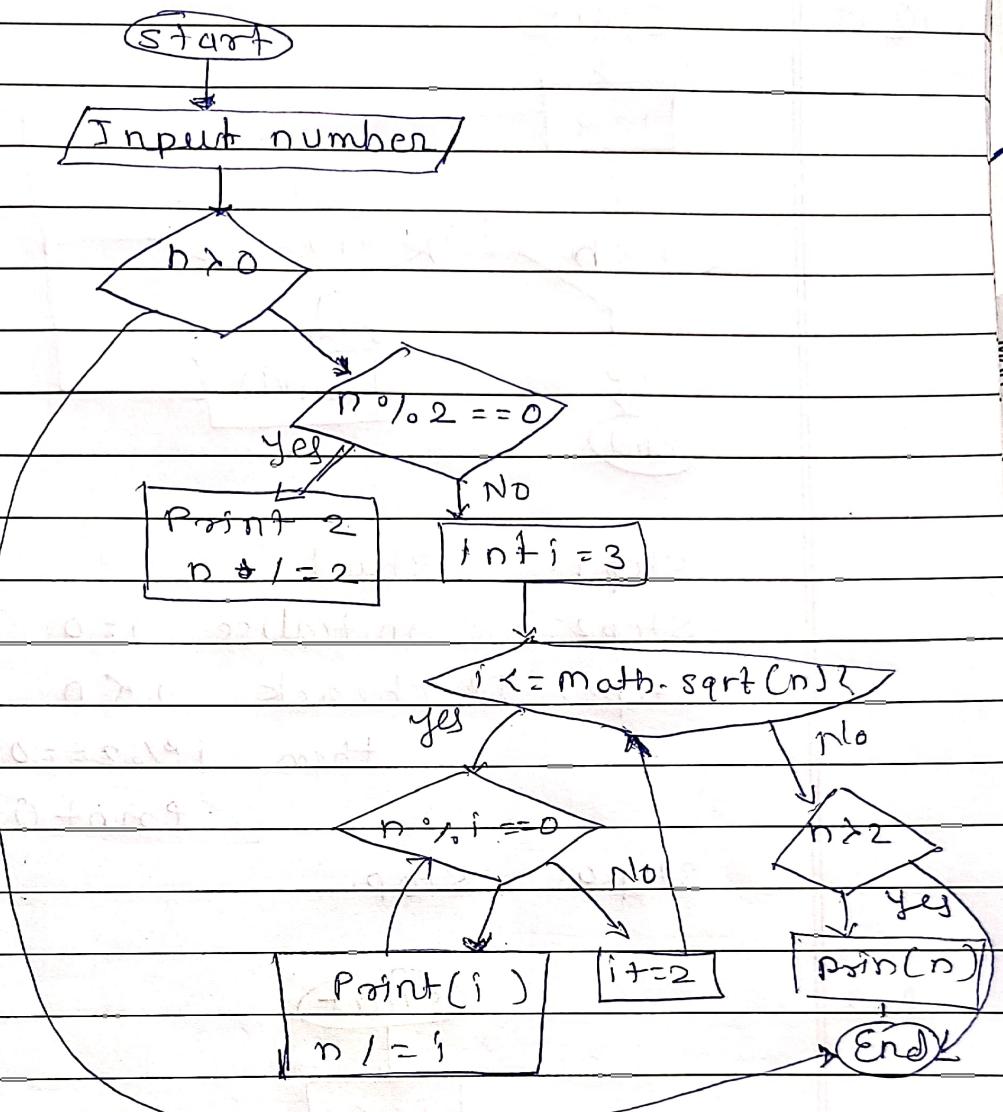
Print Palindrome

Else

Print not palindrome

Step 6: Stop.

Q 18)



Algorithm

Step 1: Input number

Step 2: check $n > 0$

Step 3: if check modulus is equal to zero
then Point 2

ELSE

assign int $i = 3$

Step 4: check $i \leq \text{math.sqrt}(n)$

Step 5: check $n \% i == 0$
then point(i)

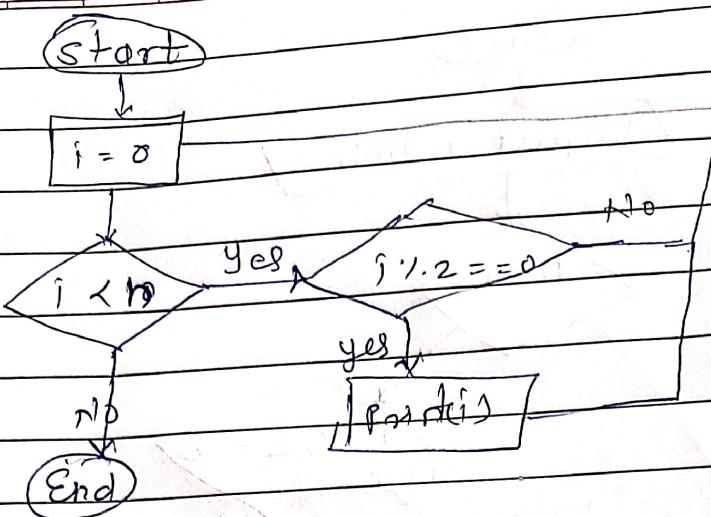
$n / i = i$

Else

$i + 2$

Step 6: STOP.

Q. 19



Step 1: Start

Step 2: initialise $i = 0$

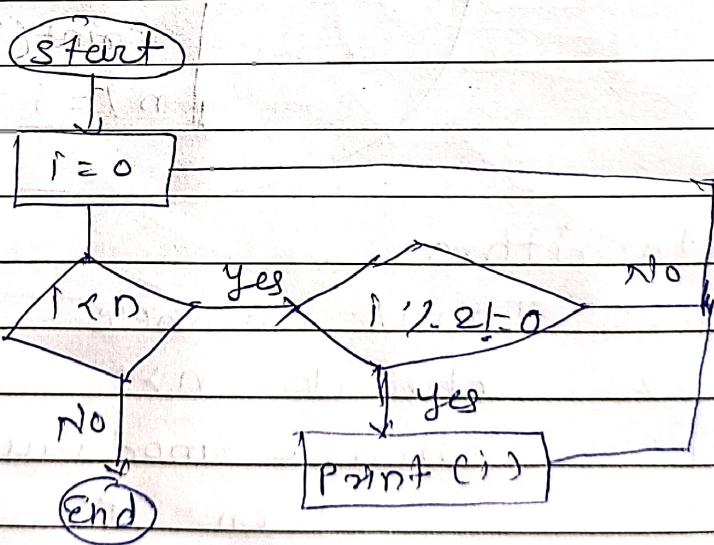
Step 3: IF check $i < n$

then $i \% 2 == 0$

print(i)

Step 4: Stop

Q. 20



Algorithm

Step 1: Start

Step 2: initialize $i = 0$

Step 3: IF check $i < n$

then $i \% 2 == 0$

print(i)

Step 4: Stop