

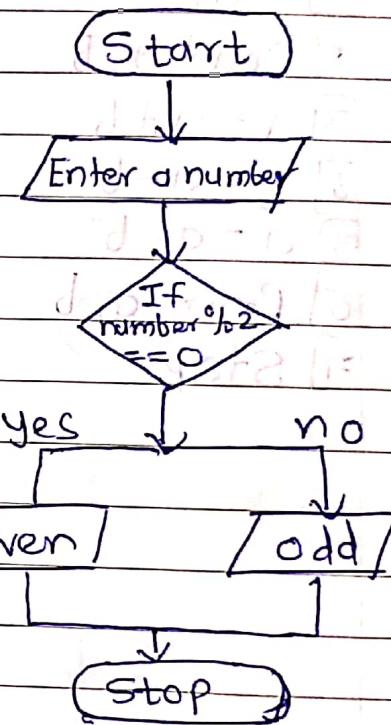
Write Algorithm & Flowchart for the following programs.

1) Check if the given number is Even or Odd.

→ Algorithm

- 1] Start the program
- 2] Input the number
- 3] If $n \% 2 == 0$ then
it is even.
- 4] and the number is odd.
- 5] Stop.

Flowchart

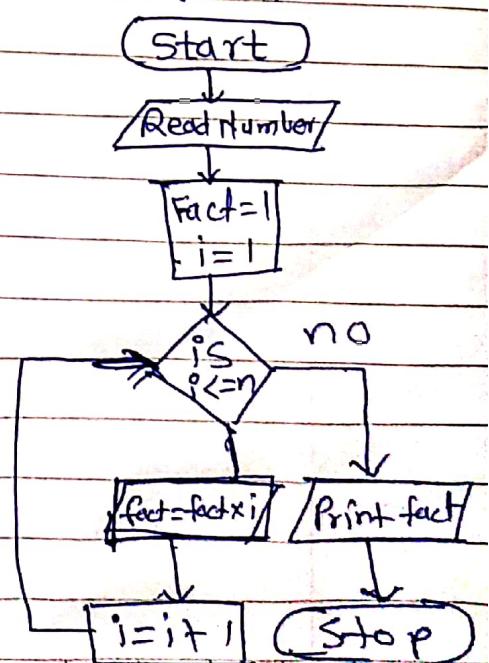


2) Write a Java Program to find the Factorial of a given number.

→ Algorithm

- 1] Start
- 2] Input number
- 3] Set Fact=1, i=1
- 4] Check condition is number,
if false goto step 2
- 5] Fact=Fact × i
- 6] Update i = i+1 go to step 4
- 7] Display Fact
- 8] Stop.

Flowchart



③ Factorial using recursion

Algorithm

Factorial function

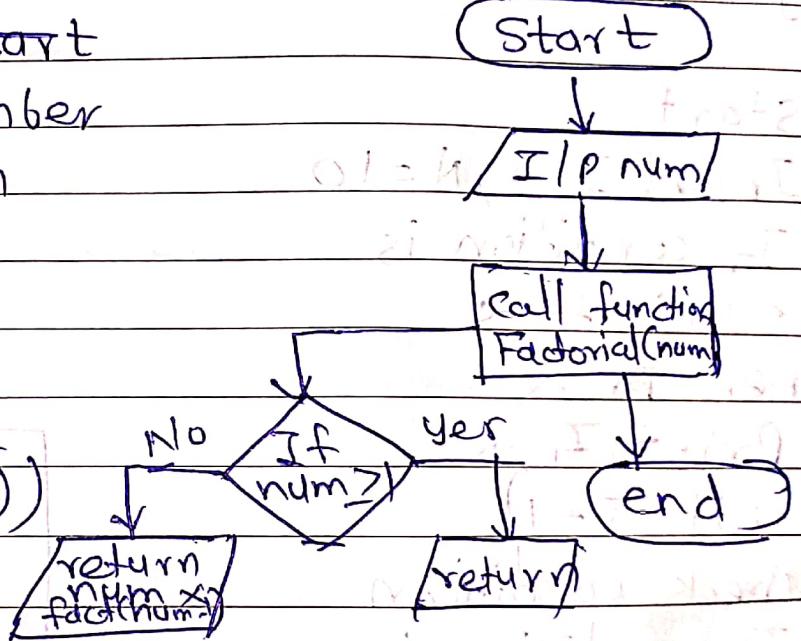
- 1) Check condition Start
- 2) Input a number
- 3) Check condition
if ($\text{num} \geq 1$)
return 1
- 4) else
return ($\text{num} \times \text{fact}(\text{num} - 1)$)

Cell function itself

main function

- 1) Start
- 2) take I/p num
- 3) Call function factorial num
- 4) Print that function.

Flowchart

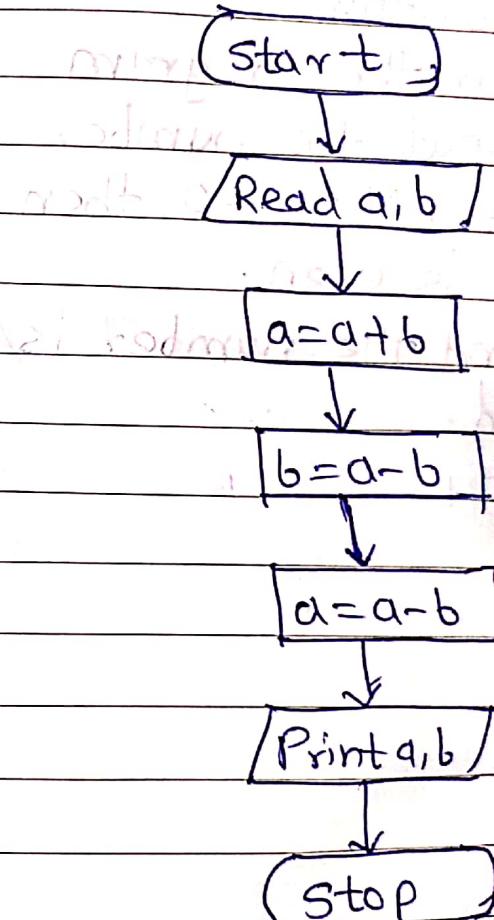


Q.4] Swap two number without using the third variable. approach.

Algorithm

- 1] Start
- 2] Read a, b
- 3] a = a + b
- 4] b = a - b
- 5] a = a - b
- 6] Print a, b
- 7] Stop

Flowchart

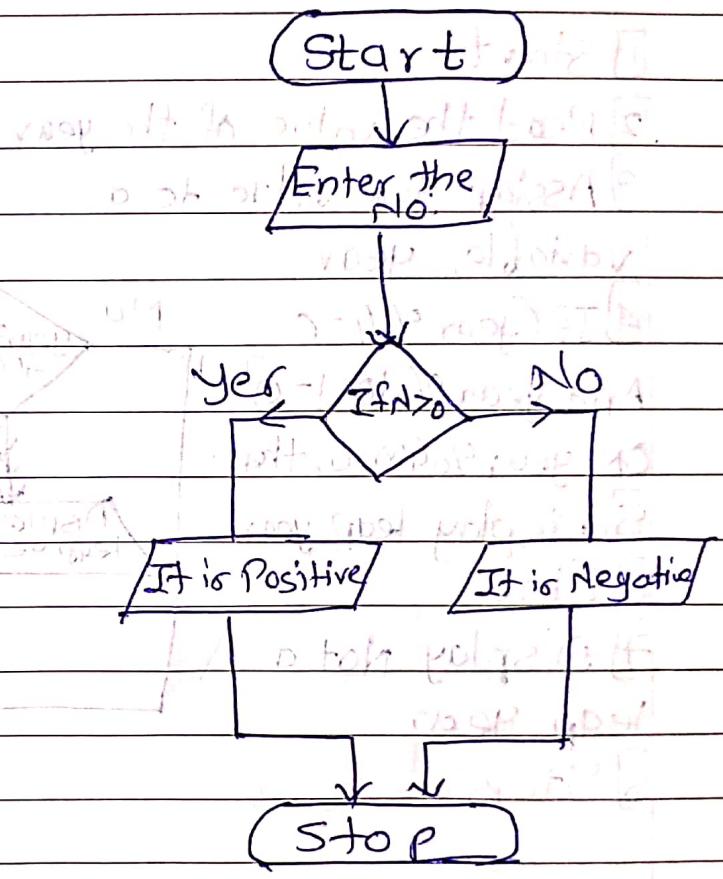


5] How to check whether the given number is Positive or Negative in Java in simple.

Algorithm

Flowchart

- 1] Start
- 2] Enter the Number.
- 3] If the Number is $N > 0$ then it is Positive OR
- 4] If the Number is $N < 0$ then it is Negative.
- 5] Stop



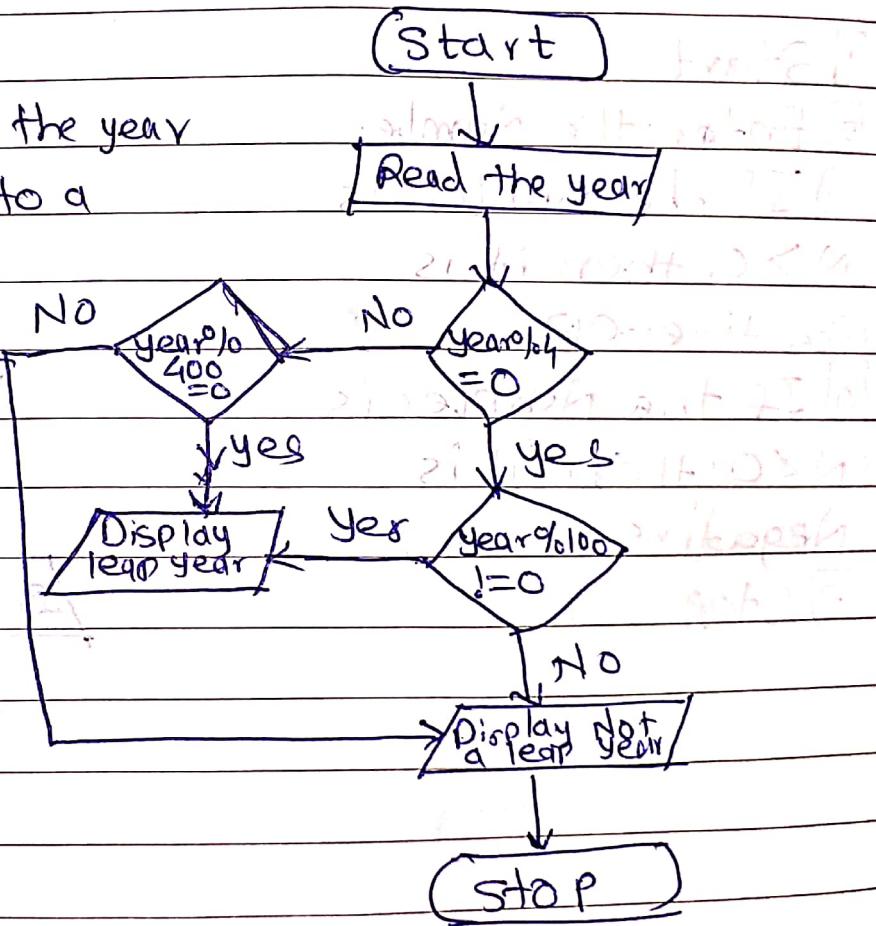
6]

Write a Java program to find whether a given number is leap year or NOT.

Algorithm

- 1 Start (Input)
- 2 Read the value of the year
- 3 Assign the value to a variable, year
- 4 If ($\text{year} \% 4 = 0$ And $\text{year} \% 100 \neq 0$) OR $\text{year} \% 400 = 0$, then:
 - 5 Display leap year
 - 6 Else
 - 7 Display Not a leap year
- 8 Stop. (Output)

Flowchart

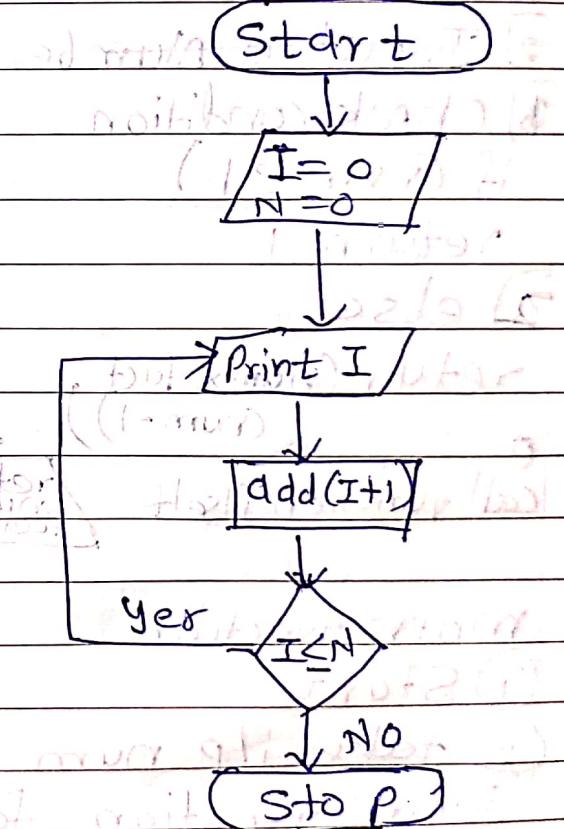


Q) Write a Java program to print 1 to 10 without using loop.

Algorithm

- 1) Start
- 2) Input $I=1, N=10$
- 3) If condition is $I < N$ is true
then
- 4) Print I
- 5) add ($I+1$)
- 6) check condition
- 7) If condition is False then
- 8) Stop.

Flowchart

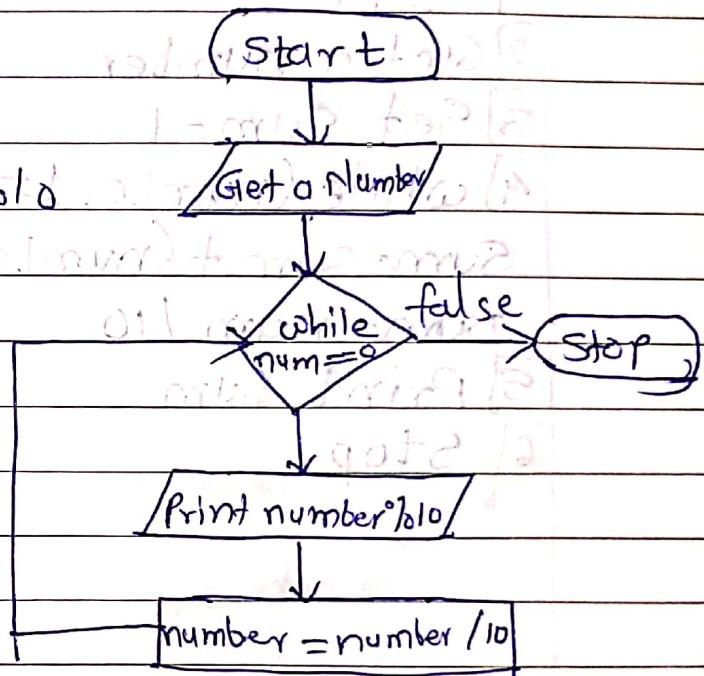


8] Write a Java Program to Print the digits of a given number.

Algorithm
(+ note)

- 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.

Flowchart

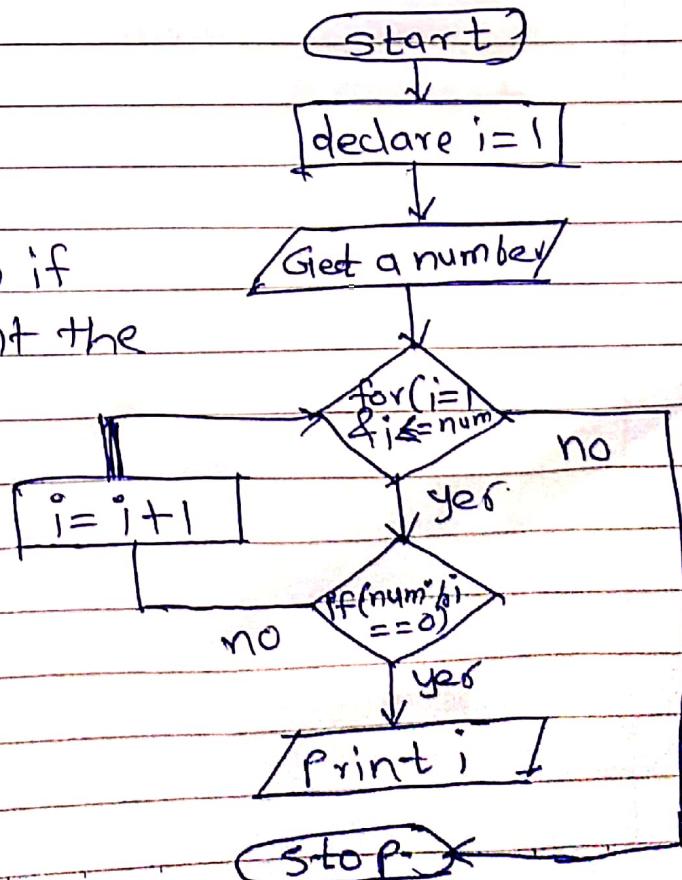


9] Factor of given number

Algorithm

- 1] Start
- 2] Get a number
- 3] Declare i=1
- 4] Check number%6 i==0 if true print i & increment the value of i
- 5] Repeat step 4 until i<=number
- 6] Stop

Flowchart



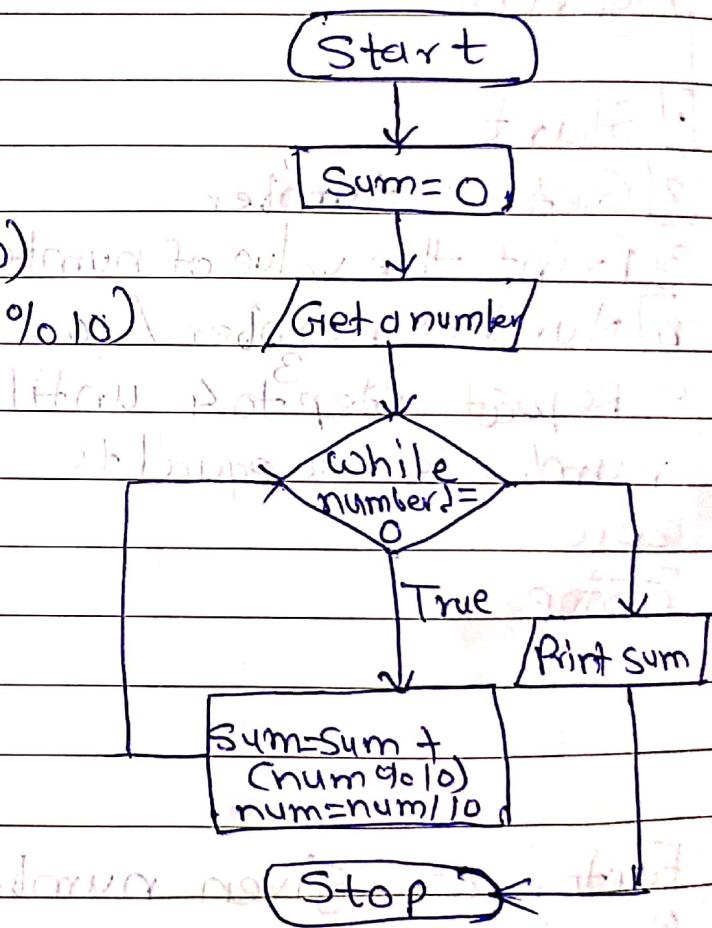
10)

Sum of digit of given number

Algorithm

- 1] Start
- 2] Get a number
- 3] Set Sum = 0
- 4] while (number != 0)
 Sum = Sum + (number % 10)
 number = number / 10
- 5] Print Sum
- 6] Stop.

Flowchart



Algorithm

Flowchart

1, 2, 3, 4, 5, 6

11) Smallest of three numbers

Algorithm

1] Start

2] Get three numbers from user

3] Check if $a < b \& a < c$, if true print a & exit else go to step 4

4] Check if $b < a \& b < c$,

if true print b & exit else go to step 5.

5] Print c .

6] Stop.

Flowchart

Start

Take input a, b, c

if ($a < b$ & $a < c$)

Print a

if ($b < a$ & $b < c$)

Yes

Print b

Print c

Stop

12) Addition without arithmetic operator.

Algorithm :-

Flowchart

1] Start

2] Get two numbers

3] Call addNum($num1, num2$) method

4] For $i = 1; i \leq num2; i++$

$a = num1 + +$

5] Print Sum

6] Stop.

Start

Get two numbers $num1, num2$

set $i = 1$

while ($i \leq num2$)

$i++$

$num++$

Print sum

Stop

13

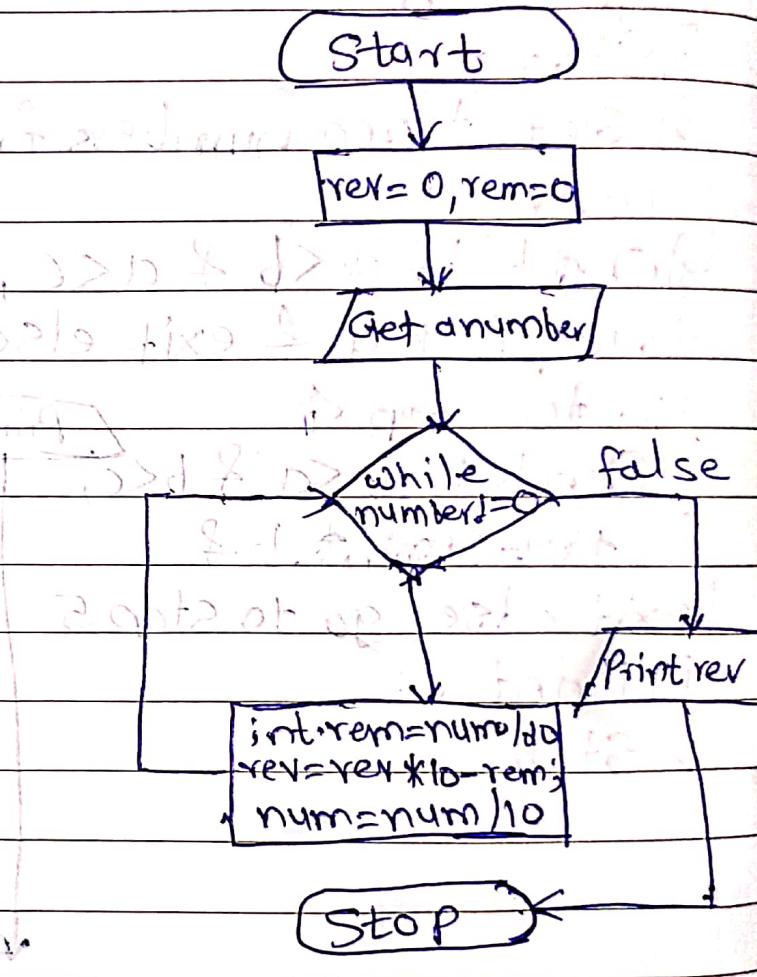
Reverse a given number.

Algorithm

Flowchart

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 + rem
 - c. num = num / 10
 - 5] Print rev
 - 6] Stop



14] GCD of two number using division method

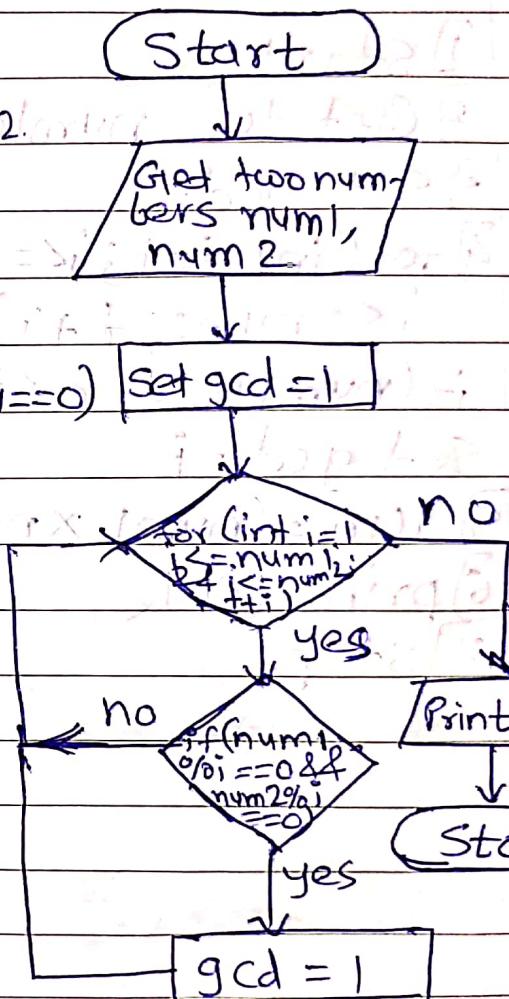
Flowchart

Algorithm :-

- 1] Start
- 2] Get two numbers num1, num2
- 3] Set gcd = 1
- 4] for (int i=1; i<=num1 && i<=num2; ++i)

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

 Set gcd = i
- 5] Print GCD.
- 6] Stop.



15

LCM of two numbers

Algorithm

Flowchart

1] Start

2] Get two numbers num1, num2

3] Set gcd=1

4] for(int i=1; i<=num1 && num1 % i == 0 && num2 % i == 0; i++;

i <= num2; i++)

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

Set gcd=i

5] lcm = (num1 * num2) / gcd

6] print LCM

7] Stop.

Start

Get two numbers num1, num2

set gcd=1, lcm=1

for(int i=1; i<=num1 && num1 % i == 0 && num2 % i == 0; i++)

i++;

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

gcd = i

lcm = (num1 * num2) / gcd

Print lcm

gcd = i

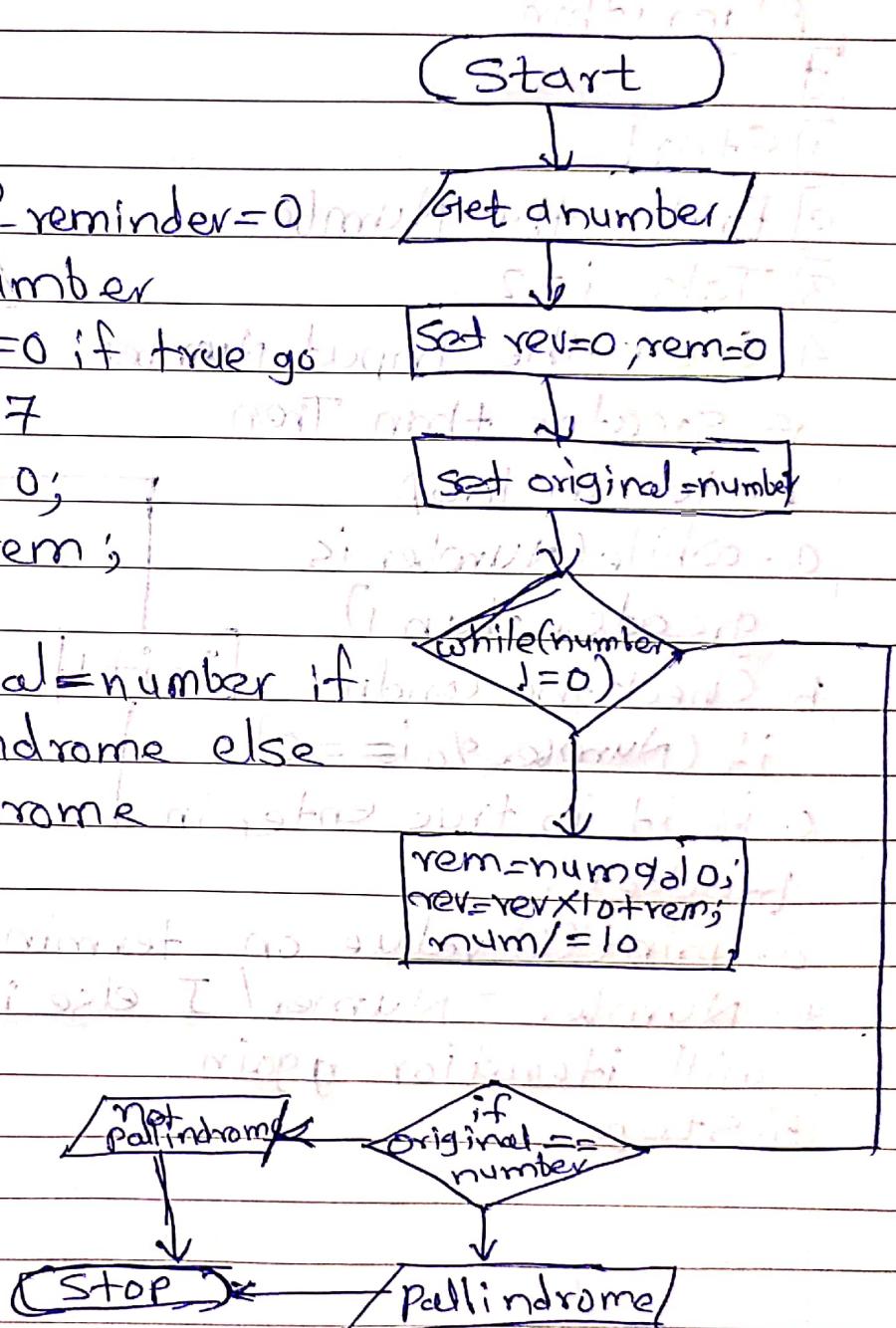
stop

17) Check Palindrome number or not.

Flowchart

Algorithm :-

- 1] Start
- 2] Get a number
- 3] Set reverse = 0 & remainder = 0
- 4] Set original = number
- 5] Check number != 0 if true goto 6
else goto 7
- 6] rem = num % 10;
rev = rev * 10 + rem;
num /= 10;
- 7] Check if original == number if true
print palindrome else print not palindrome
- 8] Stop.



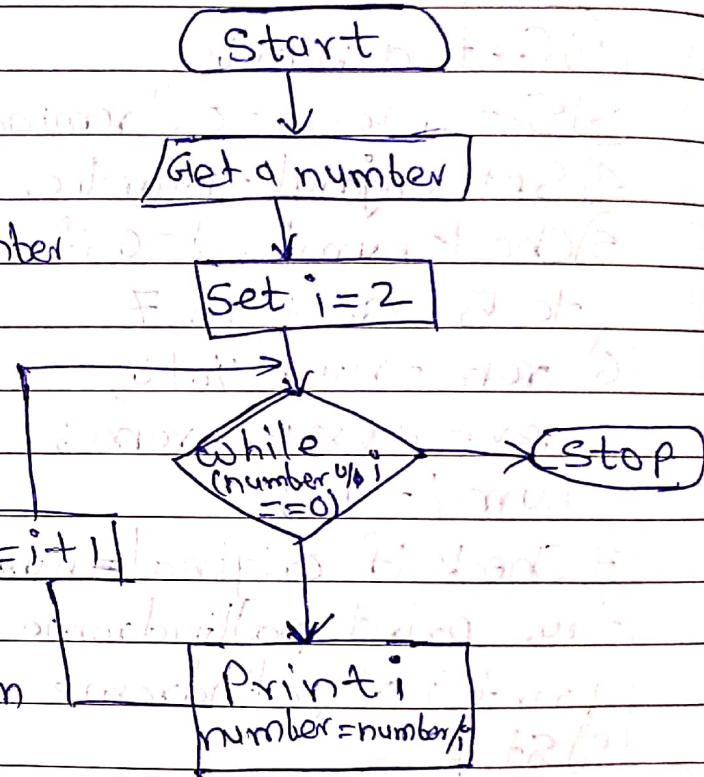
18]

Prime Factor of given number.

Algorithm

1. Start
2. Enter the Number
3. Take $i=2$.
4. Check the Input Number is greater than Then enter in loop.
 - a. while(Number is greater than i)
b. Check the condn if ($Number \% i == 0$)
c. if it is true enter in bracket.
d. print(i) value on terminal.
e. $Number = Number / i$ else i++ then loop will iteration again
5. Stop

Flowchart



19] Even Series

Algorithm :-

[1]

1] Start .

2] Get a number from user upto which they want to print even number

3] Set $i = 2$

4] If $i \leq \text{number}$, print i & $i = i + 2$.
Else go to Step 6.

5] Repeat Step 4 until $i \leq \text{number}$

6] Stop.

Flowchart

Start

Get a number

Set $i = 1$

if $i \leq \text{num}$

Stop

Print i

$i = i + 2$

20] Odd Series

Algorithm :-

[1]

1] Start

2] Get a number from user upto which they want to print odd number

3] Set $i = 1$

4] If $i \leq \text{number}$, print i & $i = 2i + 1$
Else go to Step 6

5] Repeat Step 4 until $i \leq \text{number}$

6] Stop .

Flowchart

Start

Get a number

Set $i = 1$

if $i \leq \text{num}$

Stop

Print i

$i = 2i + 1$