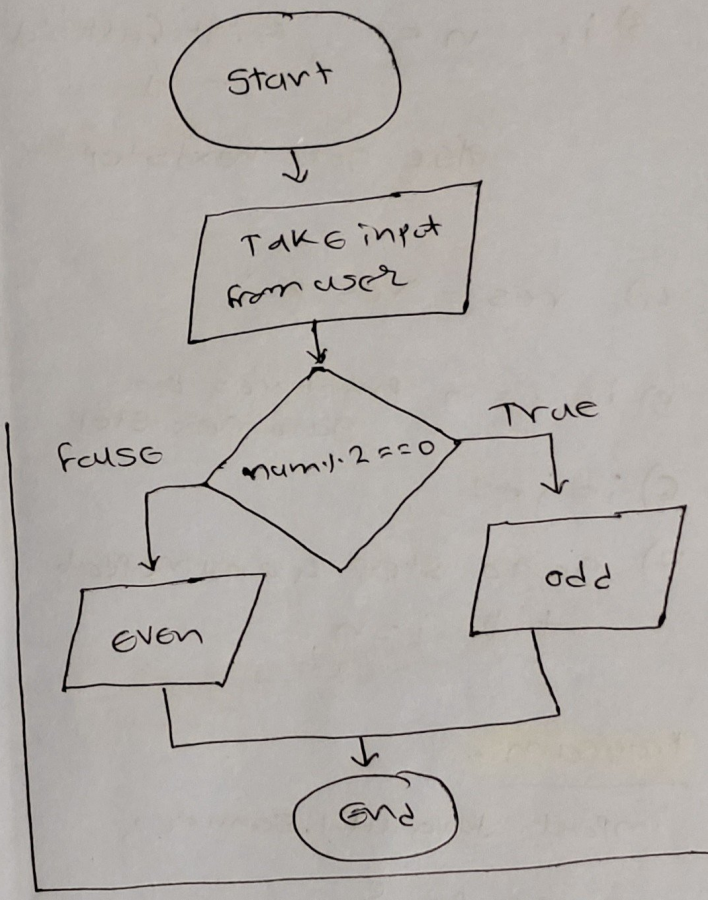
* Assignment-1

Q.1) check if the no. is even or odd.

Algorithm -

1) Take input from user
2) if num%2 == 0
   print Even
   else
   print odd

Flowchart -



Program -

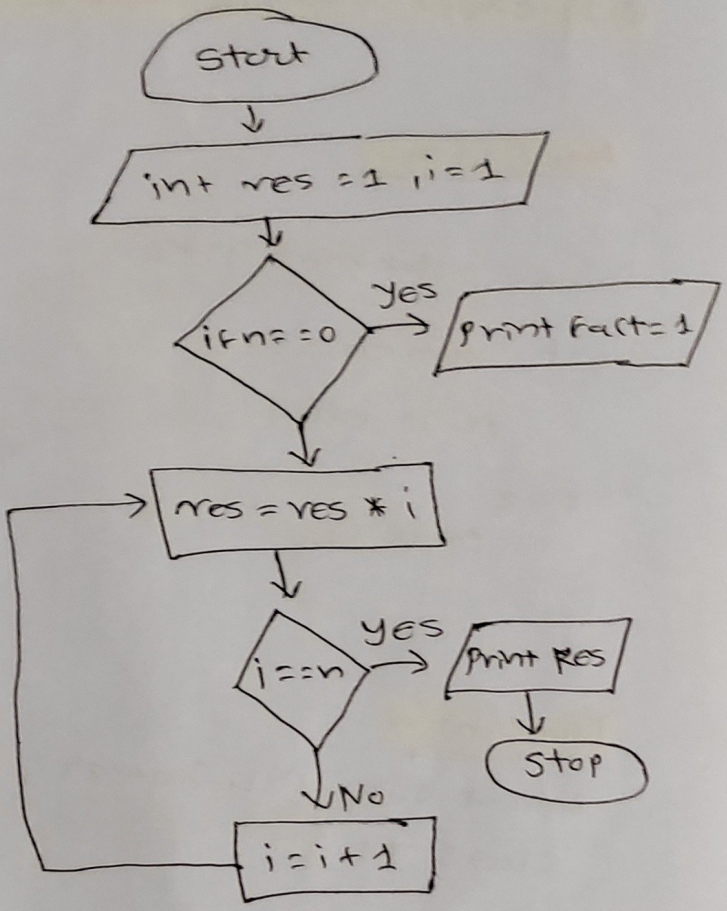
```
Import Java.util.Scanner;
 Class A1 {

Public static void main(String
                    args[]) {

Scanner sc = new Scanner (System.in);

~~int num = sc.nextInt();~~

System.out.println("Enter any no.");
    int num = sc.nextInt();
if (num%2 == 0)
 { system.out.println("Even");

   }
else { System.out.println("odd");

   }
}
}
```

Q. 2) WAP for to find Factorial of given no.

Algorithm -

1) input from user 'n'
2) intiate
   res = 1 , i = 1
3) if n = 0 , Print Factorial
   = 1
   else goto nextstep
4) res = res * i
5) if i = n print res or
   goto nex step
6) i = i + 1
7) go to step 4 and repeat
   till i = n.

Flowchart -



Program -

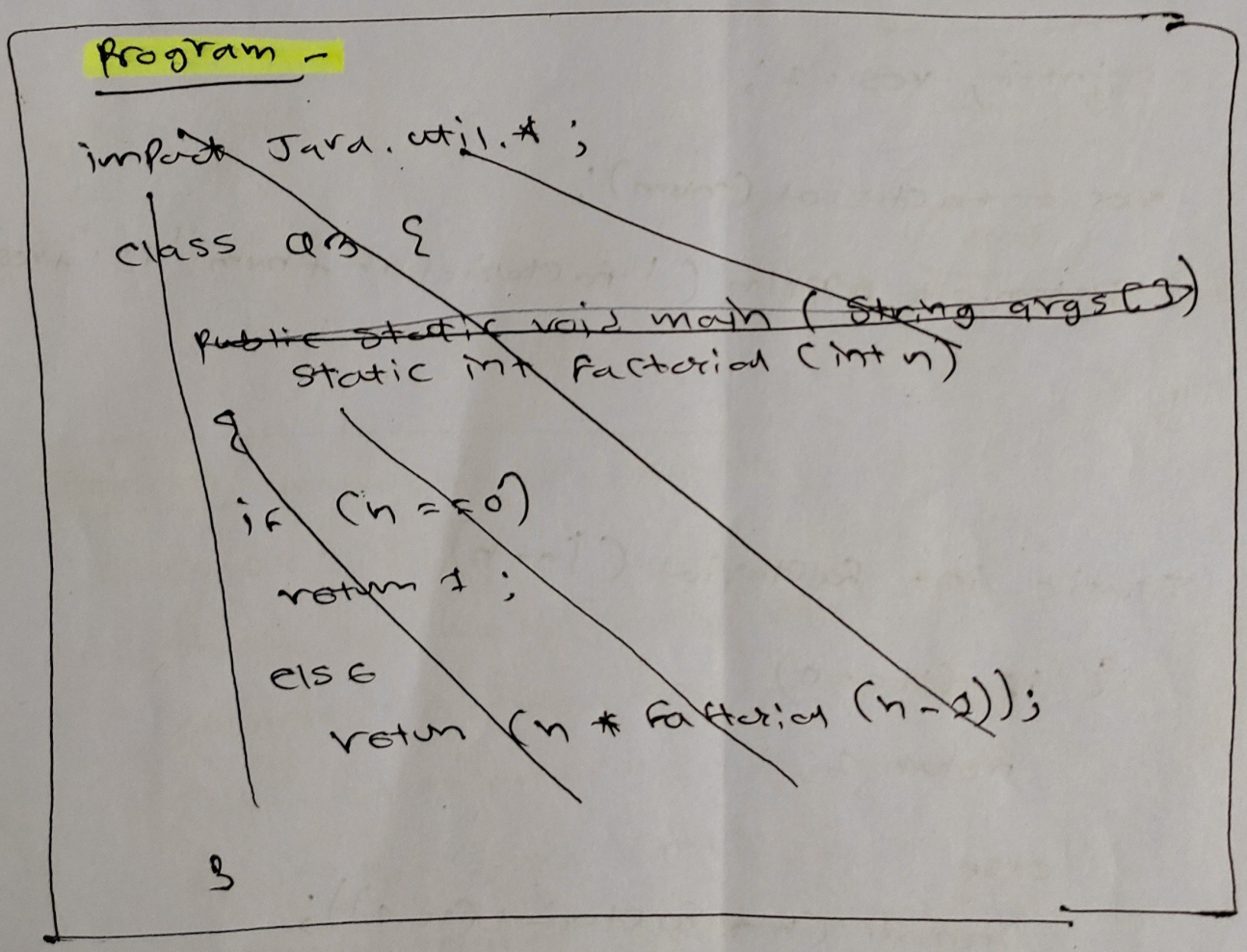
```
import java.util.Sanner;
  class A2 {
Public static void main (String args[]) {
   Scanner sc = new scanner (System.in);
   system.out.println ("Enter a number");
      int num/n = sc.nextInt ();
      int res = 1 , i = 1;
if (n = = 0)
   { system.out.println ("Factorial = 1");
   } else { while (i <= n)
          { res = res * i;
```

```
        i++;
    }
    System.out.println("Factorial = " +res);
}
```

Q.3) WAJP to find Factorial using Recursion.

Algorithm :-

Flowchart :-



Program -

```
import java.util.*;
class q3 {
    public static void main (String args[])
    static int factorial (int n)
    {
    if (n==0)
        return 1;
    else
        return (n * factorial (n-1));
    }
}
```

Q.3) WAP of factorial using recursion

Program -

```
class Q3 {

Public static void main (String args [])

{

    Scanner sc = new Sccanner (System.in);
     System.out.println ("enter any no.");
     int num = sc.nextInt();

    int i, res = 1;

  res = factorial (num);

   System.out.println ("Factorial of" + num " is:" + res);

  }

{  static int factorial (int n)

    { if (n==0)

        return 1;

      else
        return (n * factorial (n-1));

  }

}
```

(Q.4) Swap two no without using 3rd variable Approach.

Algorithm -

1) take input from user A, B

2) Calculation

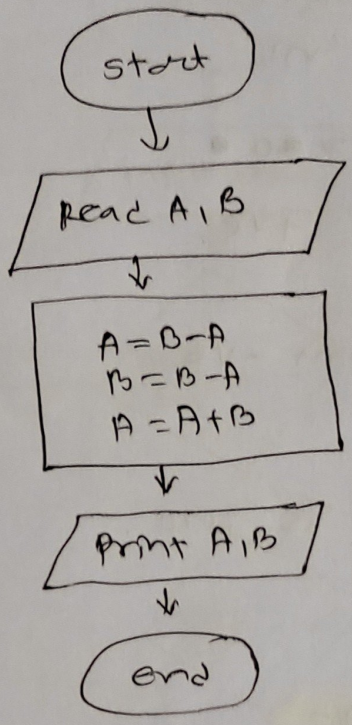
A = B - A

B = B - A

A = A + B

3) Print A, B.

Flowchart -



Program

```
import java.util.Scanner;
class a4 {
Public static void main (String args[]) {
    Scanner sc = new Scanner (System.in);
    System.out.println ("Enter value of A");
    int A = sc.nextInt();
    System.out.println ("Enter value of B");
    int B = sc.nextInt();
        A = B - A
        B = B - A
        A = A + B
    System.out.println ("A = " + A);
    System.out.println ("B = " + B);
}
```

33

(Q.5) WAP to check whether the given no. is +ve or -ve in Jav

Algorithm -

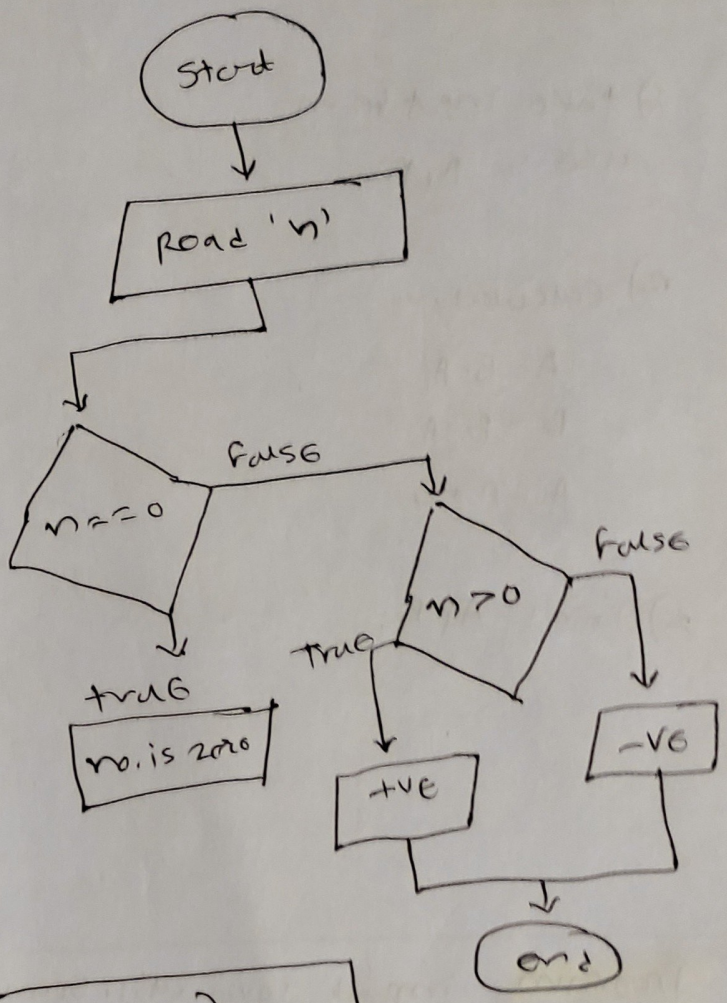
1) take input from user 'n'

2) if n > 0
   print +ve
   else if
   print -ve

3) else
   print no. is zero.

4) end

Flowchart -



Program -

```
Import Java.util.Scanner;

  classas {
Public static void main (String args[]){
  Scanner sc = new Scanner (System.in);
    int n = sc.nextInt();

  if (num>0)
    { System.out.Println ("+ve no.");

    }
  else if (num<0)
    { System.out.println ("-ve no.");

  } else
    { System.out.println ("no. is zero");
  } }.
```

(Q.6) WAP find whether the no. is Leap year or Not.

Algorithm -

1) input year

2) if $\frac{year}{100} == 0$, goto step 3

   else go to step 4

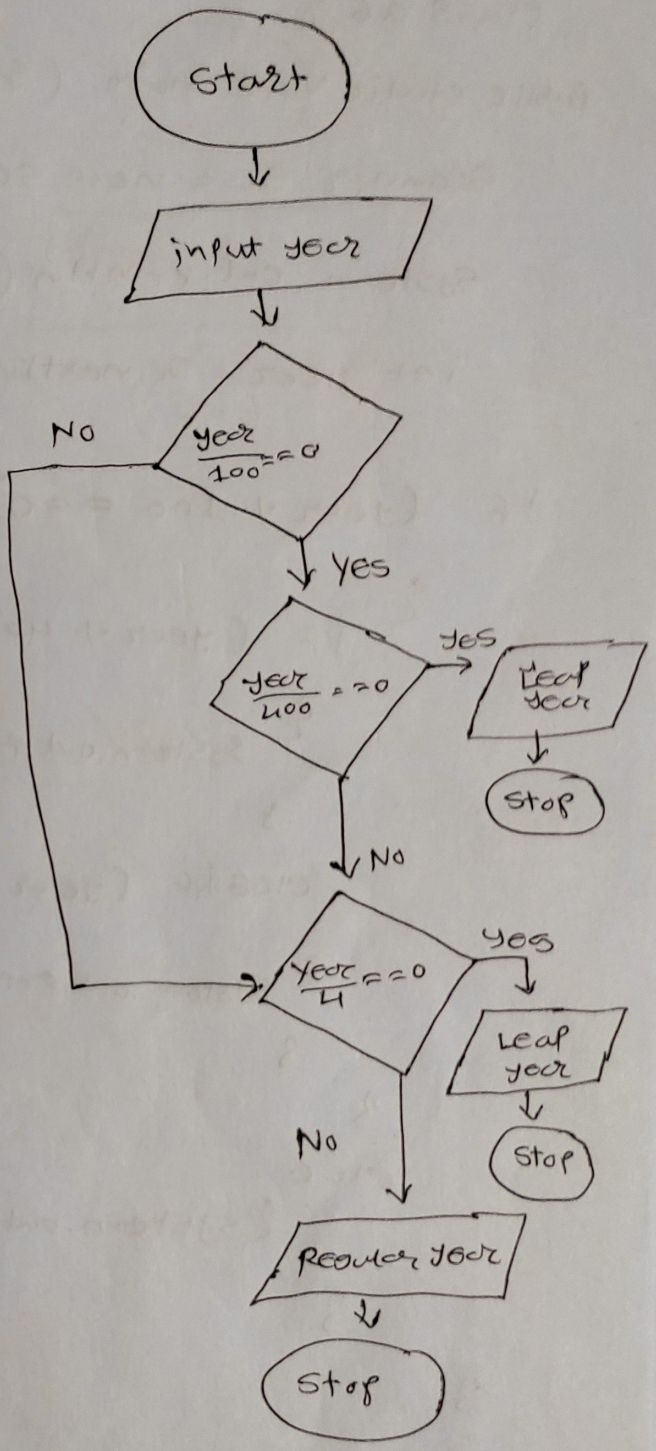
3) if $\frac{year}{400} == 0$ Print Leap year

   else goto step 4

4) if $\frac{year}{4} == 0$, Print Leap year

   else print Regular year

Flowchart -



Program -

```
import java.util.Scanner;

class a6 {
Public static void main (String args[]) {
    Scanner sc = new Scanner (System.in);
    System.out.println ("Enter a year");
    int year = sc.nextInt ();

if (year %100 ==0)
    {
      if (year %400 ==0)
        {
          System.out.println("this is a Leap year");
        }
      else if (year %4 ==0)
        { System.out.println ("this is a Leap year");
        }
    }
    else
        {System.out.println("this is a common year");
    }
}
}
```

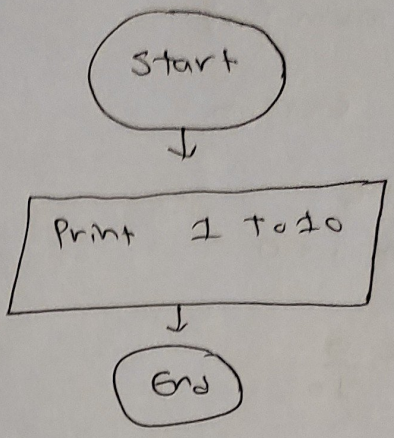
Program - import java.util.Scanner;

Q.7) Write a Java program to print 1 to 10 without using Loop.

algorithm -

1) Print - 1
2
3
4
5
6
7
8
9
10

Flowchart -



Program -

```
Class a7 {

public static void main ( String args[] ) {

    System.out.println (1);
                       (2);
                       (3);
                       (4);
                       (5);
                       (6);
                       (7);
                       (8);
                       (9);
                       (10);
```

3
3

Q.8) WAJP to print the digits of a given no.

Algorithm -

1) input num (n)

int c = num

2) n = c

int i = 0

3) $n = \frac{n}{10}$

4) i++

5) if n < 10, goto next step

else goto step 3

& repeat.

6) print n

7) $c = c - 10^{i} \times n$

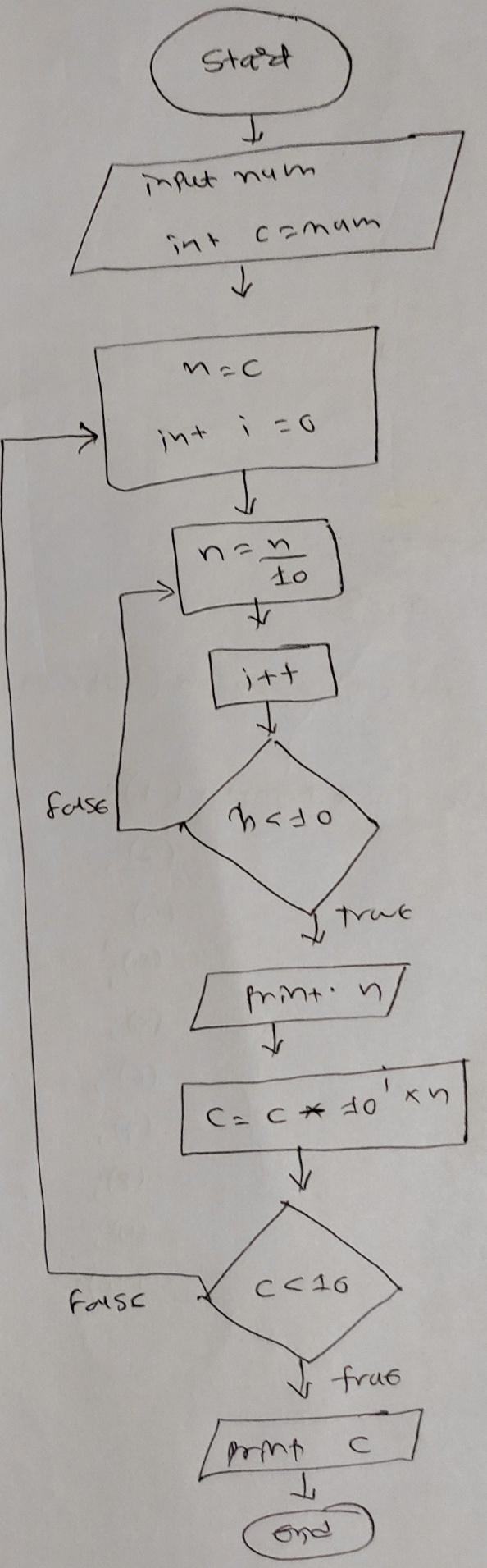
8) if c < 10, goto next step

else goto step 2

& repeat.

9) print 'c' -> Stop

Flow chart -



Program –

```
import java.util.Scanner;

Class a8 {
Public static void main ( String args[]) {
    Scanner sc = new Scanner (System.in);
    System.out.println ("Enter a number");
    int num  = sc.nextInt ();
    int c = num;
    int n = c;
    int i = 0;
    while (c >= 10)
{   do
    {  n = n/10
        i++;
    }
    while (n >= 10);
    System.out.println (n);
      c = c - 10^i * n;
      n = c;
      i = 0;
    }
    System.out.println (c);
}
}
```

(Q.9) WAJP to print all the factors of given no.

Algorithm -

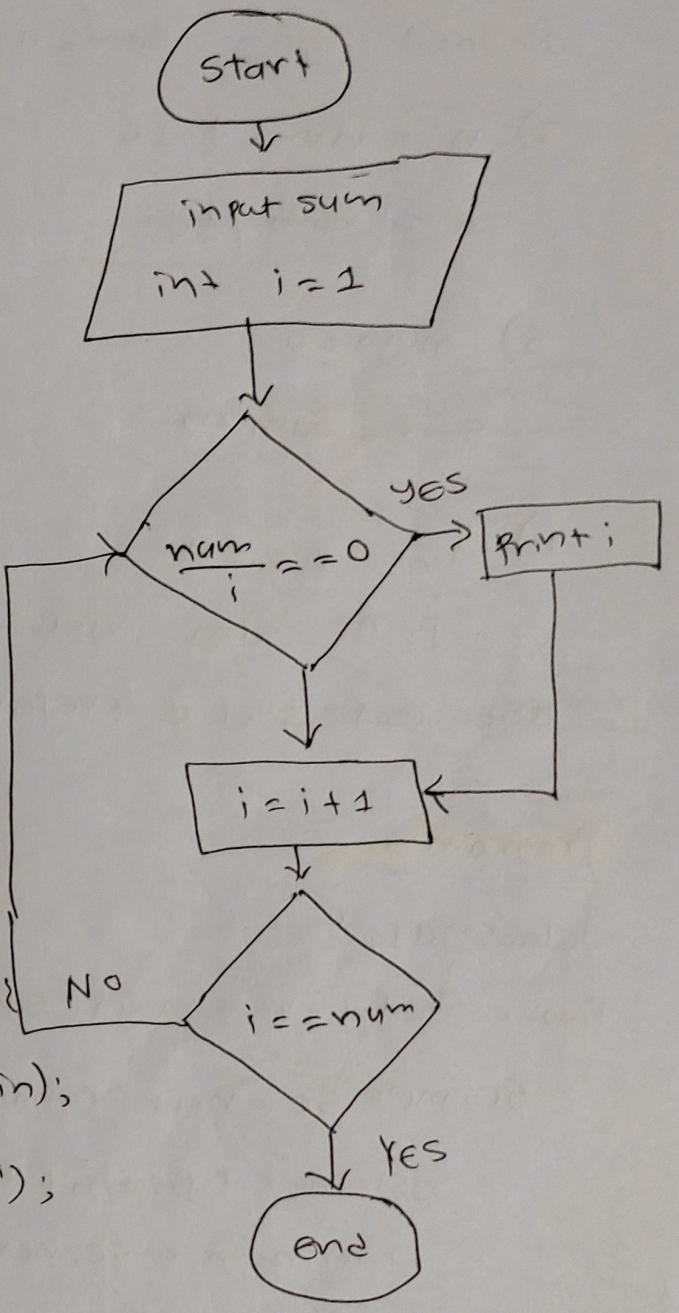
1) input num
   int i = 1

2) num / i == 0 , print i

3) i = i+1

4) if i = num stop
   otherwise goto STEP 2
   & REPEAT.

Flowchart -



Program -

```
class ag {
Public static void main (String args[]) {
   Scanner sc = new Scanner(System.in);
   System.out.println("Enter a no.");
     int num = sc.nextInt();
   for (int i=2; i<=num; i++)
   { while (num%i == 0)
     { System.out.print(" " +i+ " ");
       num = num/i;
   3
3  3 3 .
```

Q.10) WAJP to find sum of digits of given no.

Algoritm -

1) Input num , sum = 0

2) b = num % 10

   a = num / 10

3) num = a

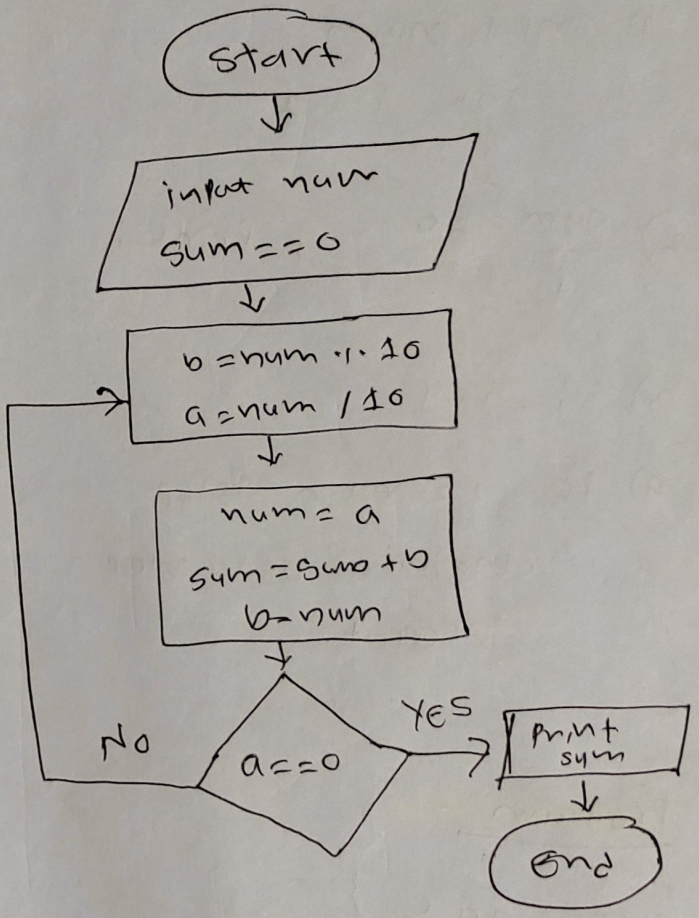
   sum = sum + b

4) if a == 0

   print sum → stop

else goto step 2 & repeat

Flowchart -



Program -

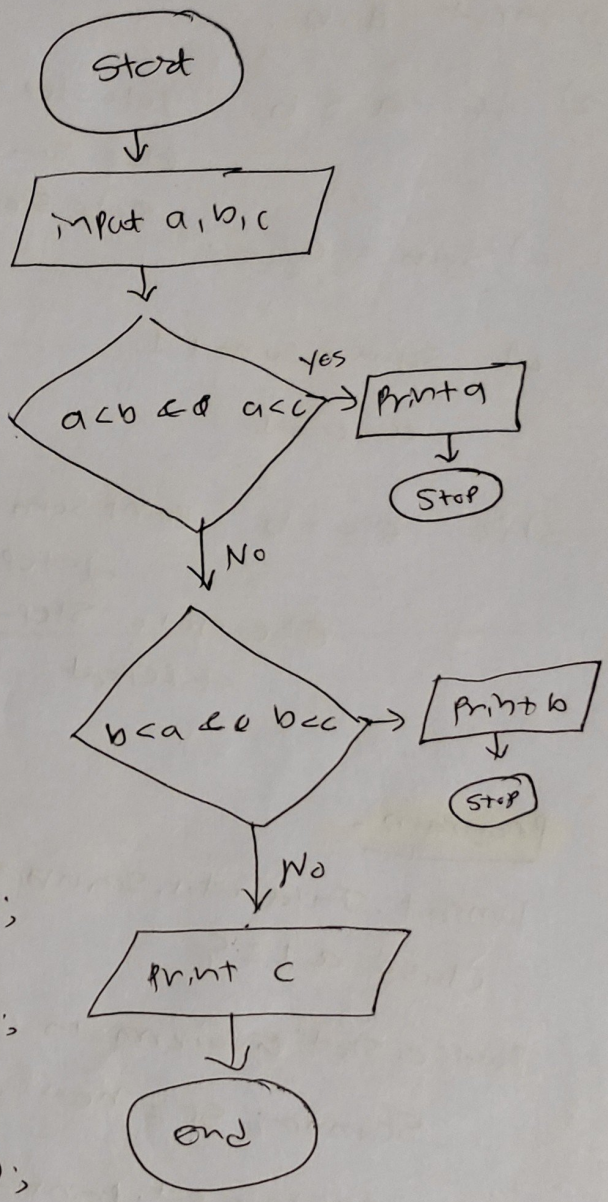
```
class q10 {
Public Static void main(String args[]){
    Scanner sc = new Scanner (System.in);
    System.out.println("Enter a no.");
     int num = sc.nextInt();
      int sum = 0;
      int a,b;
     do
    {
        b = num % 10;
        a = num / 10;
         num = a;
        sum = sum + b;
         b = num;
    }
    while (a>0);
    System.out.println("sum =" +sum);}}
```

(Q.11) WAJP for smallest of 3 No. (a,b,c)

algorithm -

1) input a,b,c
2) if a<b && a<c
   → print 'a'
3) if b<c && b<a
   print 'b'
   else print 'c'

flowchart -



Program -

```
class a11 {

Public static void main(String args[]) {

Scanner sc = new Scanner(System.in);
System.out.println("enter value of a");
int a = sc.nextInt();

System.out.println("enter value of b");
int b = sc.nextInt();

System.out.println("enter value of c");
int c = sc.nextInt();

if (a<b && a<c)
{ System.out.println("smallest number=" + a);
}
else if (b<a && b<c)
{ System.out.println("smallest number=" +b);
}
else { System.out.println("smallest number=" +c);
}
}
}
```

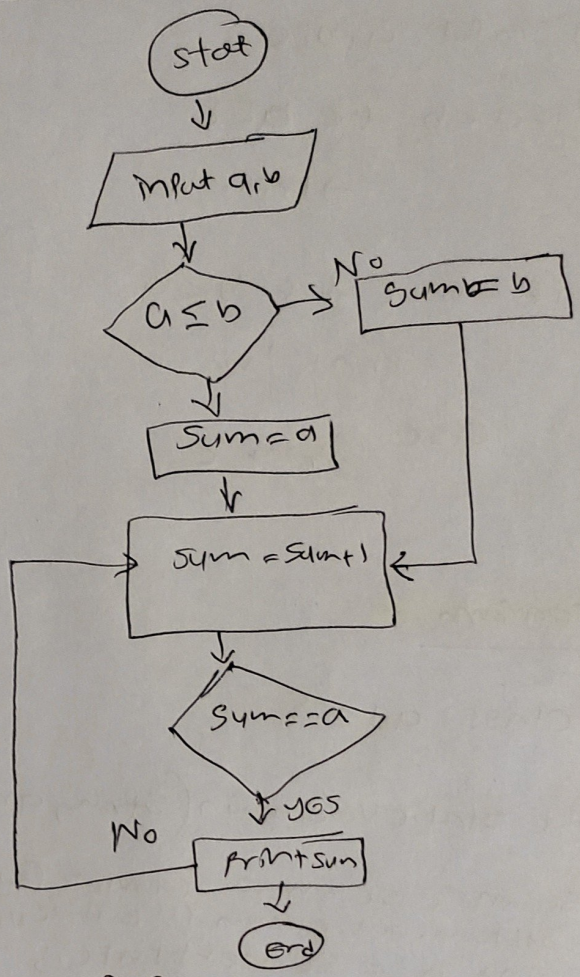
{

Q. 12 WAJP How to add two no. without using arithmatic operators in Java.

Algorithm -

1) input a, b.
2) if $a \leq b$ goto step 3
   else sum = b
   goto step 4
3) sum = a
4) sum = sum + 1
   a = a + 1
5) if a == b print sum
   stop
   else goto step-4
   & Repeat

Flowchart -



Program -

```
Import.Java.util.Scanner;
  Class a12{
Public static void main (String args[]){
    Scanner sc = new Scanner ( System.in);
     System.out.println ("Enter value of 'a'");
      int a = sc.nextInt ();
     System.out.println ("enter value of b");
      int b = sc.nextInt();
  if (b>=0)
  {   for (int i=0 ; i<b ; i++)
        { a++;
        }
        System.out.println("sum = "+a);
        }
```

```
else
 {
      for ( int i=0 ; i>b ; i--)
      { a--;
       }
       System.out.println ("sum =" +a);
      }

}
}
```

greatest common divisor (GCD)

a. No.

Program 14 HCF of two no.

```
Import java.util.scanner;
class a14{
Public static void main (String args [ ]) {
   Scanner sc = new Scanner (System.in);
      System.out.println ("Enter 1st no.");
      Int a = sc.nextInt ();
      System.out.println ("enter 2nd no.");
       int b = sc.nextInt ();
       int i=1;
       int HCF = 1;
if (a==b)
 { System.out.println ("HCF=" +a);
  }
   else
     { while (i<=a && i<=b)
       { if (a%i==0 && b%i==0)
          { HCF =i;
            i++;
           }
        else
            { i++;
            }
       }
      System.out.println ("HCF=" +HCF);
   }
}
```

Q.14) WAJP for GCD of two no.
greatest common divisor (HCF)

Algorithm -

1) input a, b.
int i = 1

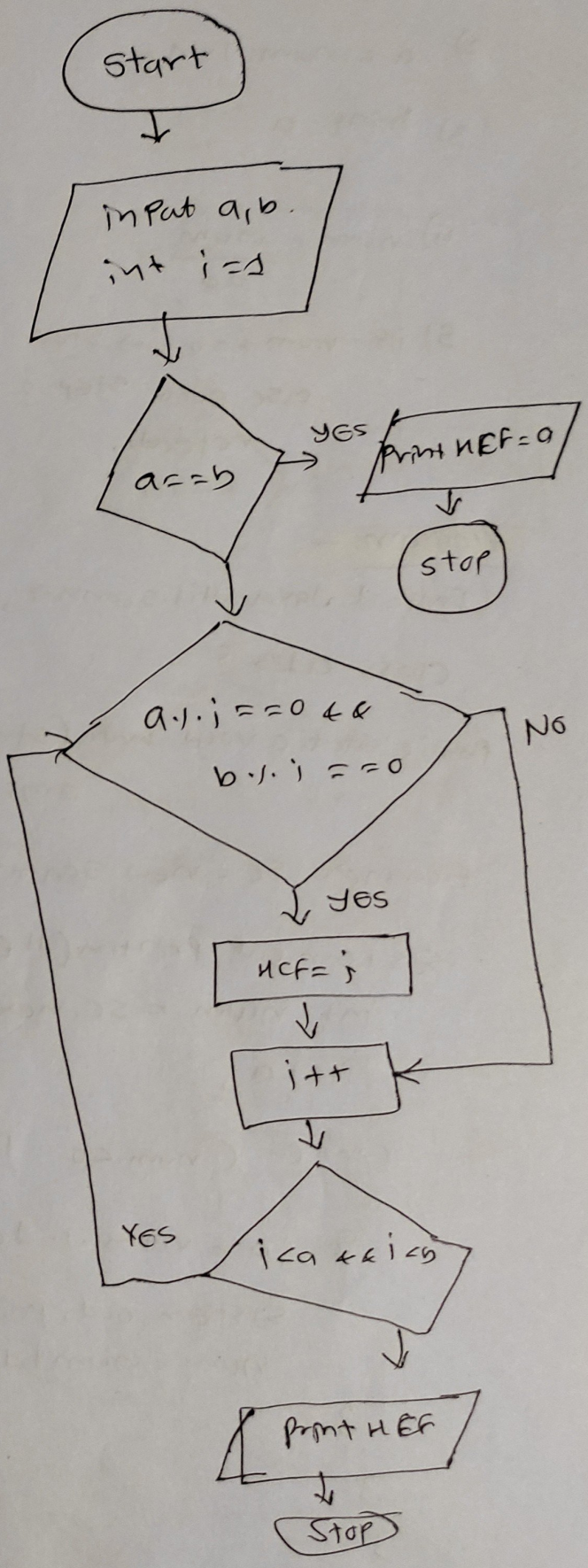
2) if a == b, print
HCF = a
else goto next step

3) if a % i == 0 && b % i == 0
↳ HCF = i
else goto next step

4) i++

5) if i < a && i < b,
goto step 3 ← Repeat
else print HCF
↓
stop

Flowchart -



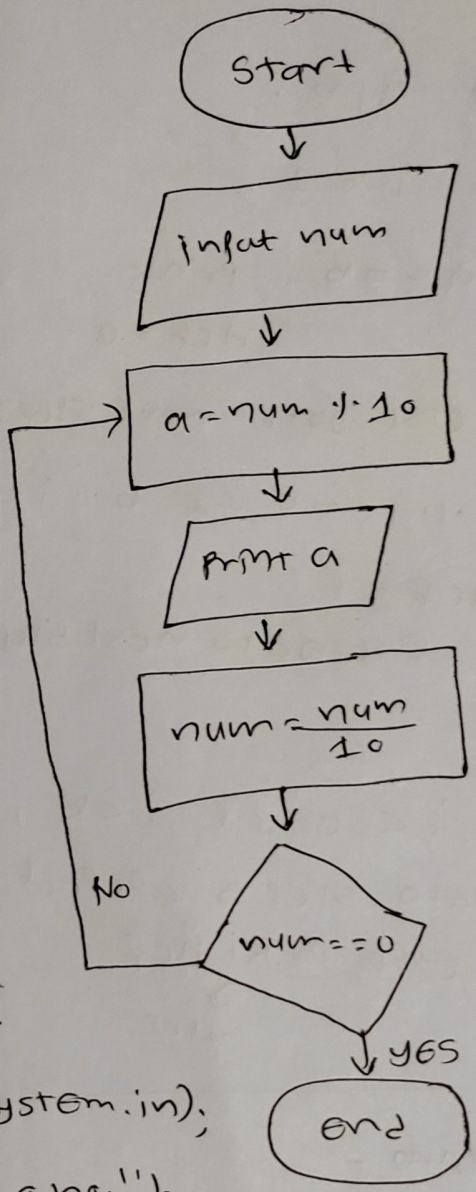
Program -

(Q.13) WAJP for REVERSE a given no.

Algorithm -

1) input num.
2) a = num % 10
3) Print a
4) num = num / 10
5) if num == 0 → Stop
   else goto step 2 &
   repeat.

Flow chart -



Program -

```
Import Java.util.Scanner;
 Class a13 {
public static void main(String
                    args[]) {
Scanner sc = new Scanner(System.in);
 System.out.println("Enter a no.");
    int num = sc.nextInt();
     int a;
   while (num<0  || num>0)
    {  a = num % 10;
       System.out.print(a);
       num = num / 10;
  }
 }
}
```

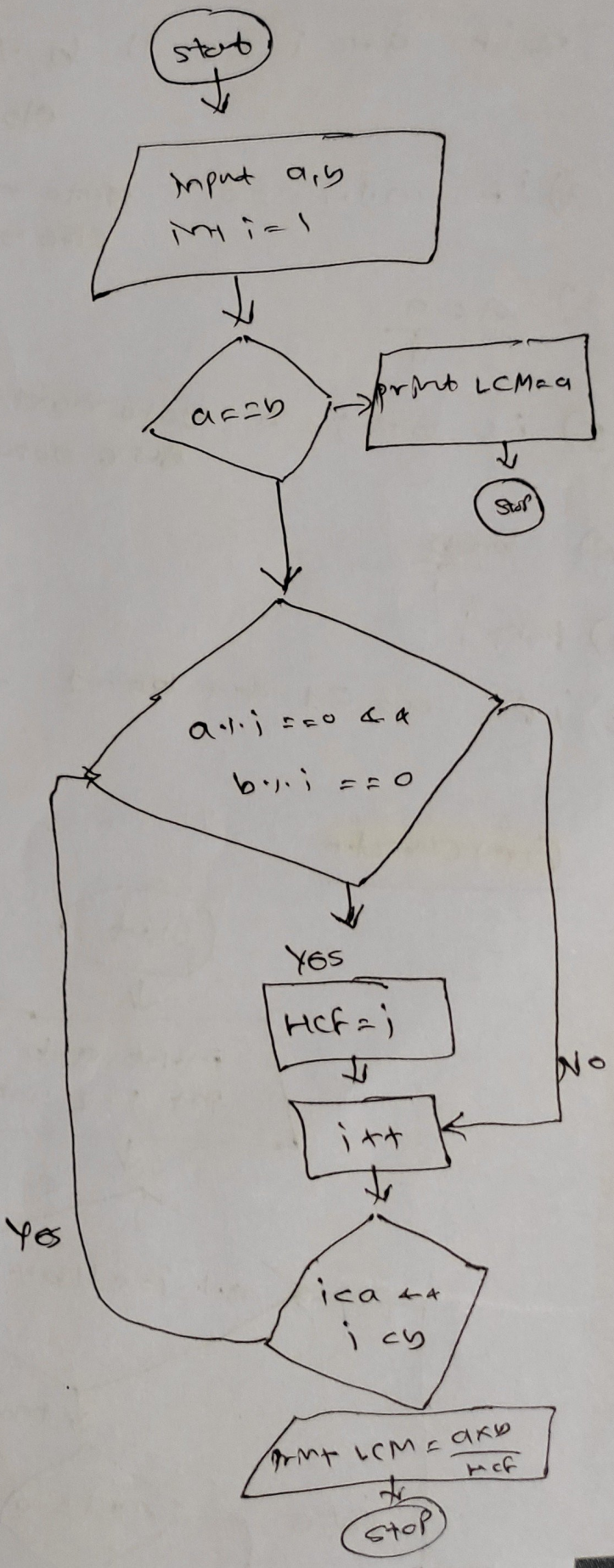
1 *-

(@45) write a Java program to find LCM of two given no.

Algorithm -

1) input a, b, int i=1.
2) if a==b
   print LCM = a
   else goto next step
3) if a%i == 0 && b%i == 0
   HCF = i
   else goto next step
4) i++
5) if i<a && i<b
   goto step 3 & repeat
6) print LCM = $\frac{a \times b}{HCF}$

Flow chart -



Q.16) Write a Java Program to Find LCM of two given no. using the prime factors method.

Algorithm

1) input a, b , int i = 2

2) if a %. i == 0 || b %. i == 0 , goto next step
   else i++ & repeat step 2

3) if a %. i == 0 goto next step
   else goto step 5

4) a = a / i

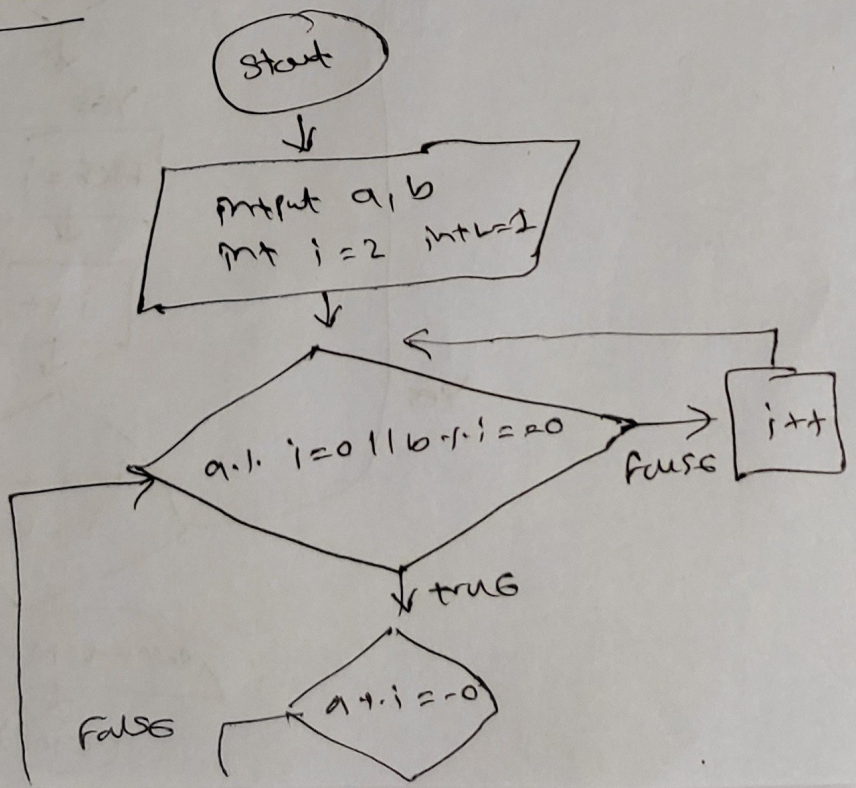
5) if b %. i = 0 goto next step
   else goto step 7

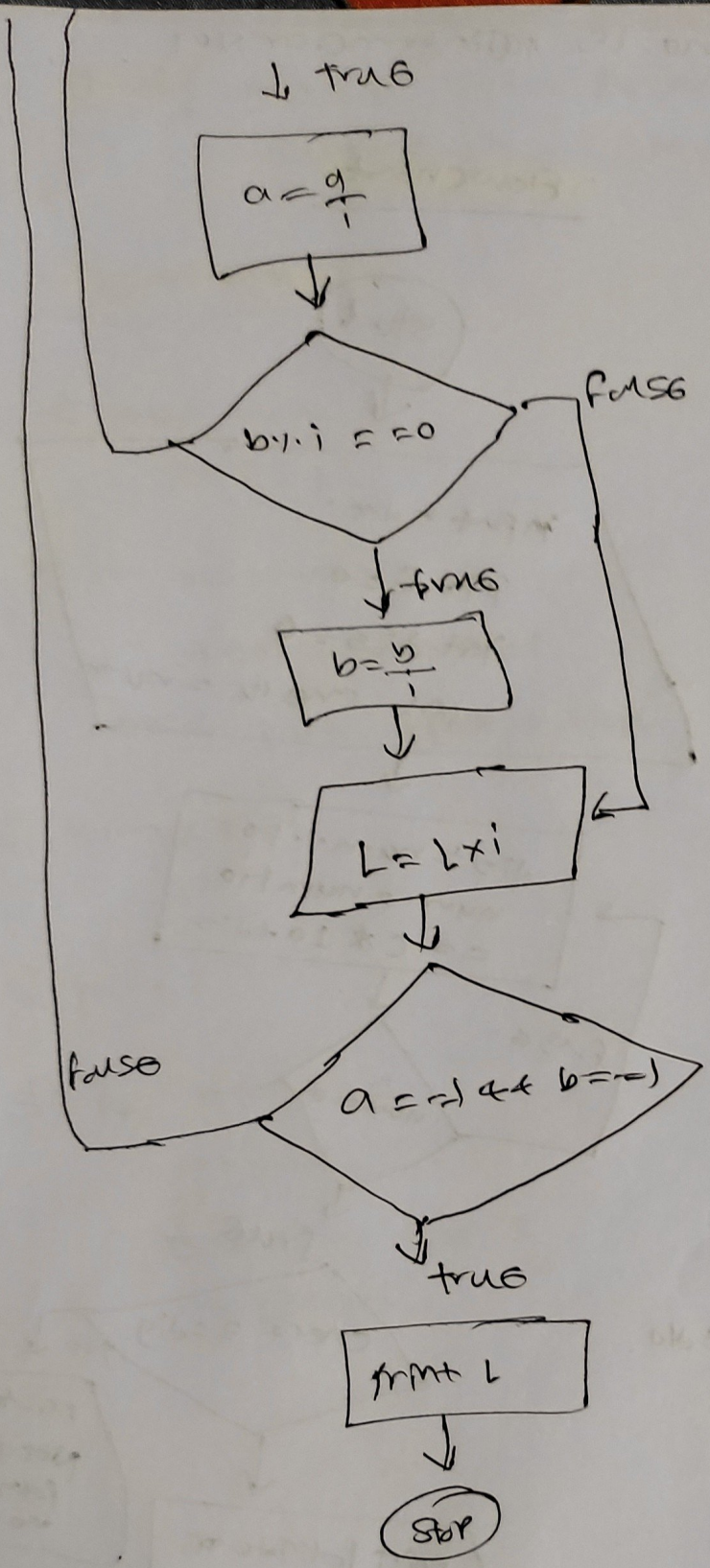
6) b = b / i

7) print i

8) if a == 1 && b == 1 → stop
   else goto step 2 & repeat

Flow Chart:-




true
a = a / i
b % i == 0
false
true
b = b / i
L = L * i
false
a == -1 && b == -1
true
print L
Stop

Q.17) Check whether the no. is Palindrome or Not.

Algorithm

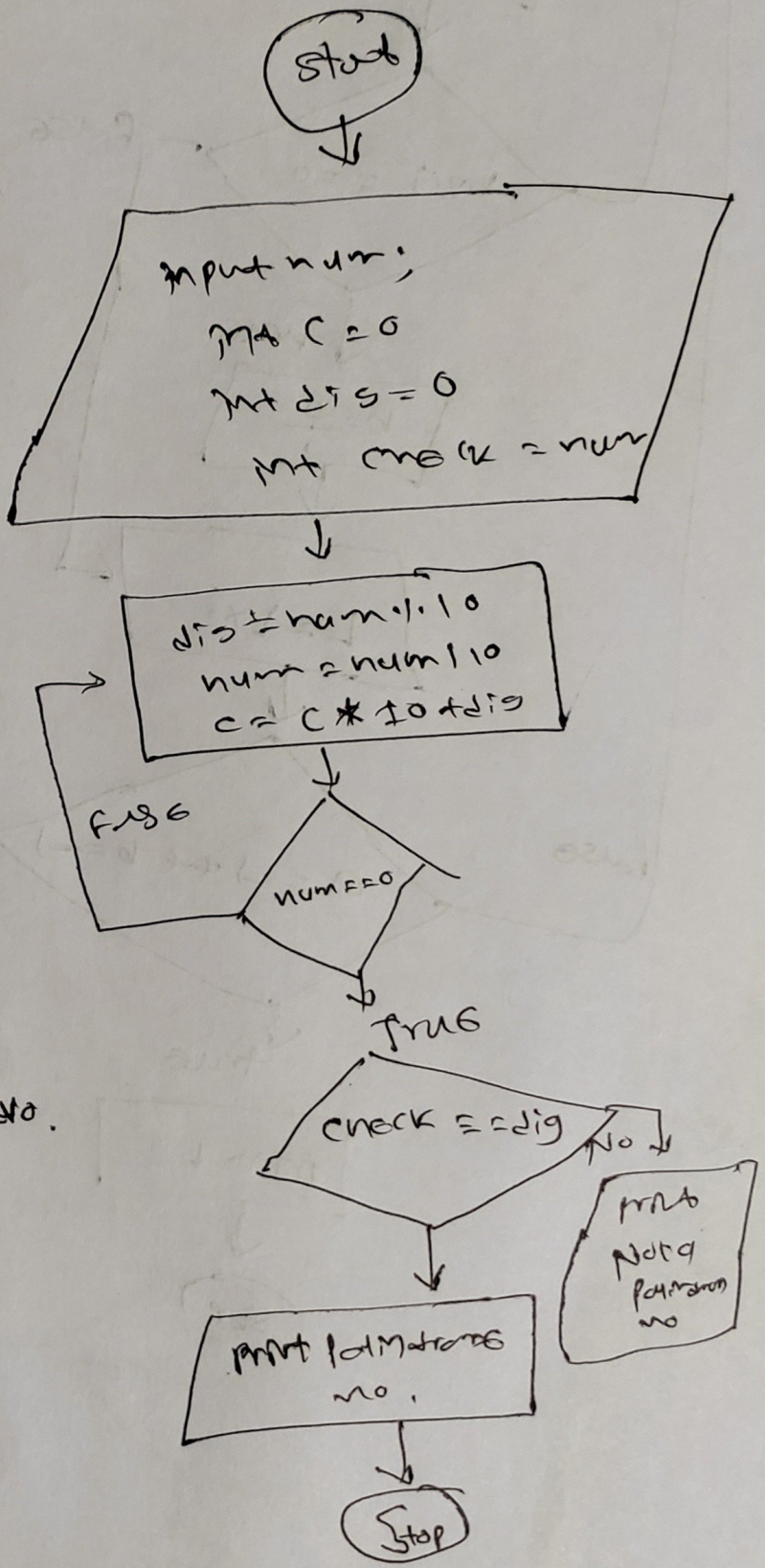
1) input num,
   int c = 0
   int dig = 0
   int check = num

2) dig = num % 10
   num = num / 10
   c = c * 10 + dig

3) if num = 0,
   goto next step
   else goto step 2 & repeat

4) if check = = dig
   print Palindrome no.
   else print not palindrome no.

Flowchart



Q.18) write a Java program to print all the prime factors of a given no.

Algorithm

1) input num,
   int i = 2

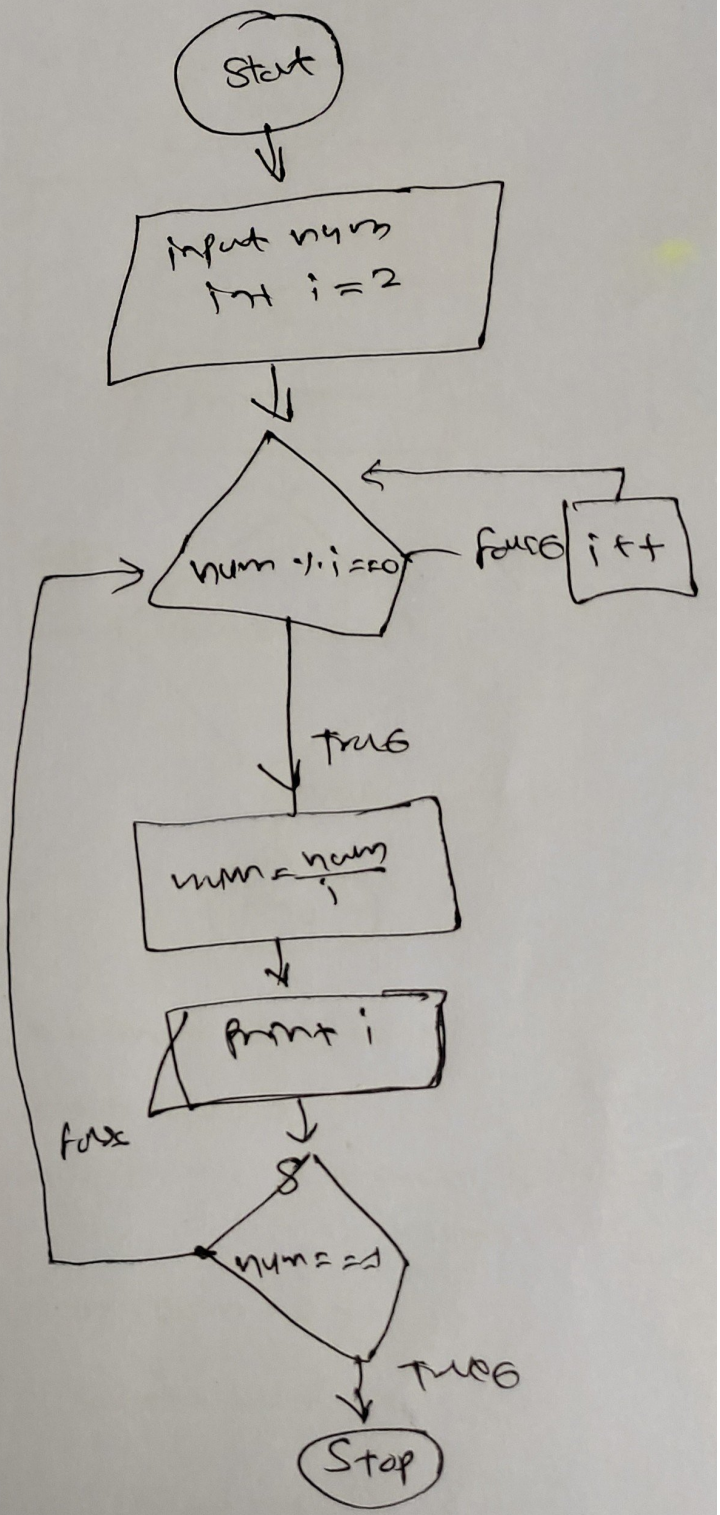
2) if num % 2 == 0
   → goto next step
   else i++ & Repeat step2

3) num = num / i

4) print i

5) if num == 1
   → stop
   else goto step 2
   & Repeat.

Flowchart



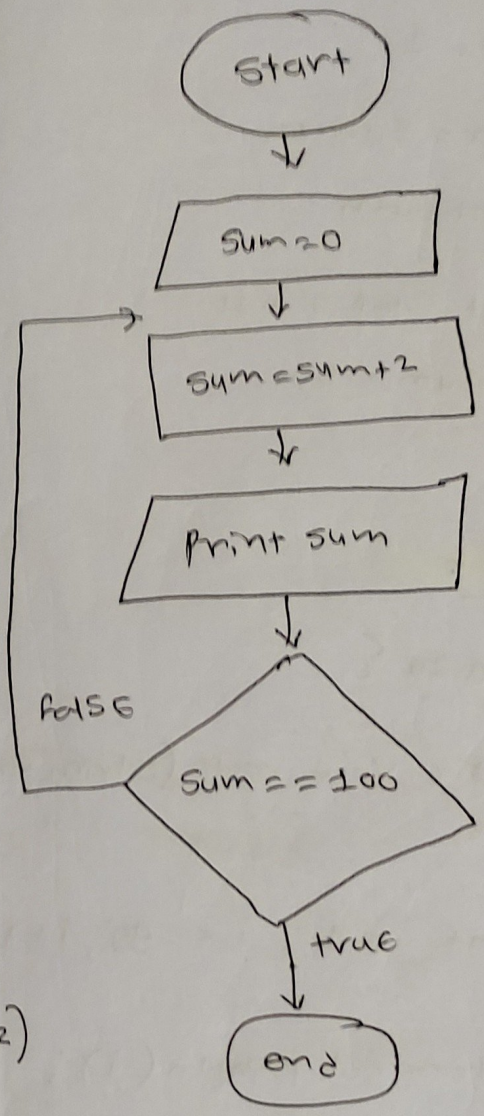
(Q.19) WAJP for even no. Series 2 4 6 8 10 ...

Algorithm =

1) Sum = 0] input
2) Sum = Sum + 2
   Print Sum] logic

Repeat step 2 upto
Sum = 100

flowchart =



Program =

```
class a19 {

public static void main (String
                    args [] ) {

for (int i=2; i<=100; i=i+2)

  {
    System.out.println(i);

  }
}
}
```

* without using Loop

```
class a19 {
Public static void main {String
                    args() ) {

  int sum = 0;

   sum = sum + 1;

if (sum <=100)
  { return
   System.out.println(i++);
}
```

(Q.20) WAJP for odd no. series

Algorithm -

1) Sum = 1

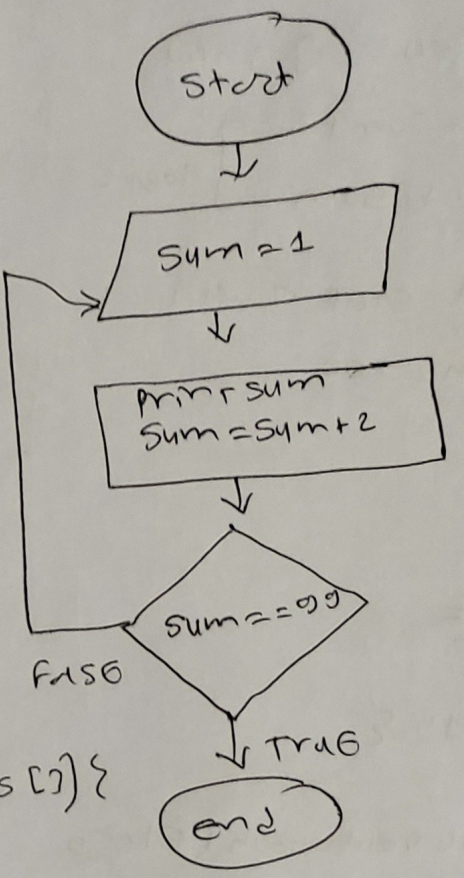
2) Sum = Sum + 2

   Print Sum

3) Repeat Step 2 till

   Sum == 99

Flowchart -



Program =

```
class a20 {

Public static void main (String args [ ]) {

   for ( int i = 1 ; i <= 99 ; i++)

    { System.out.println ( i );

    3

3
3
```