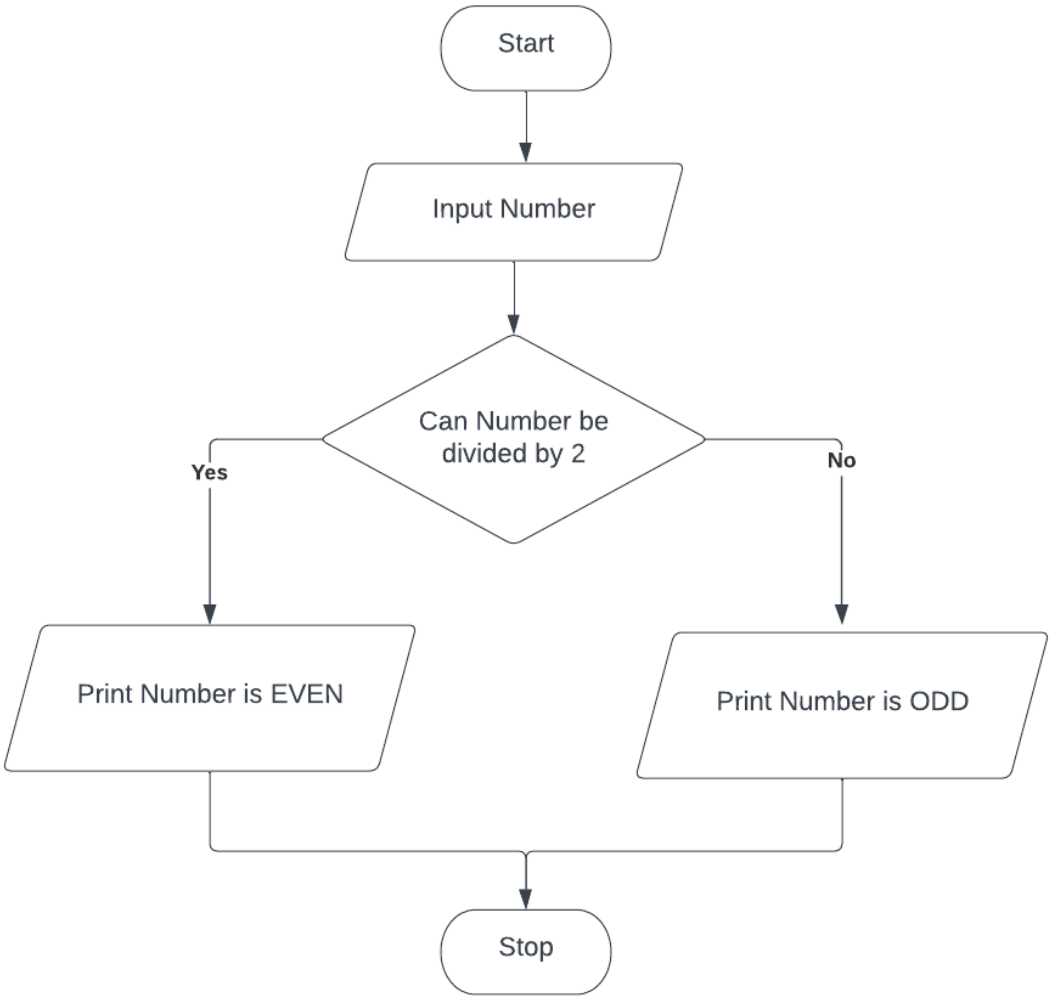
Assignment 1

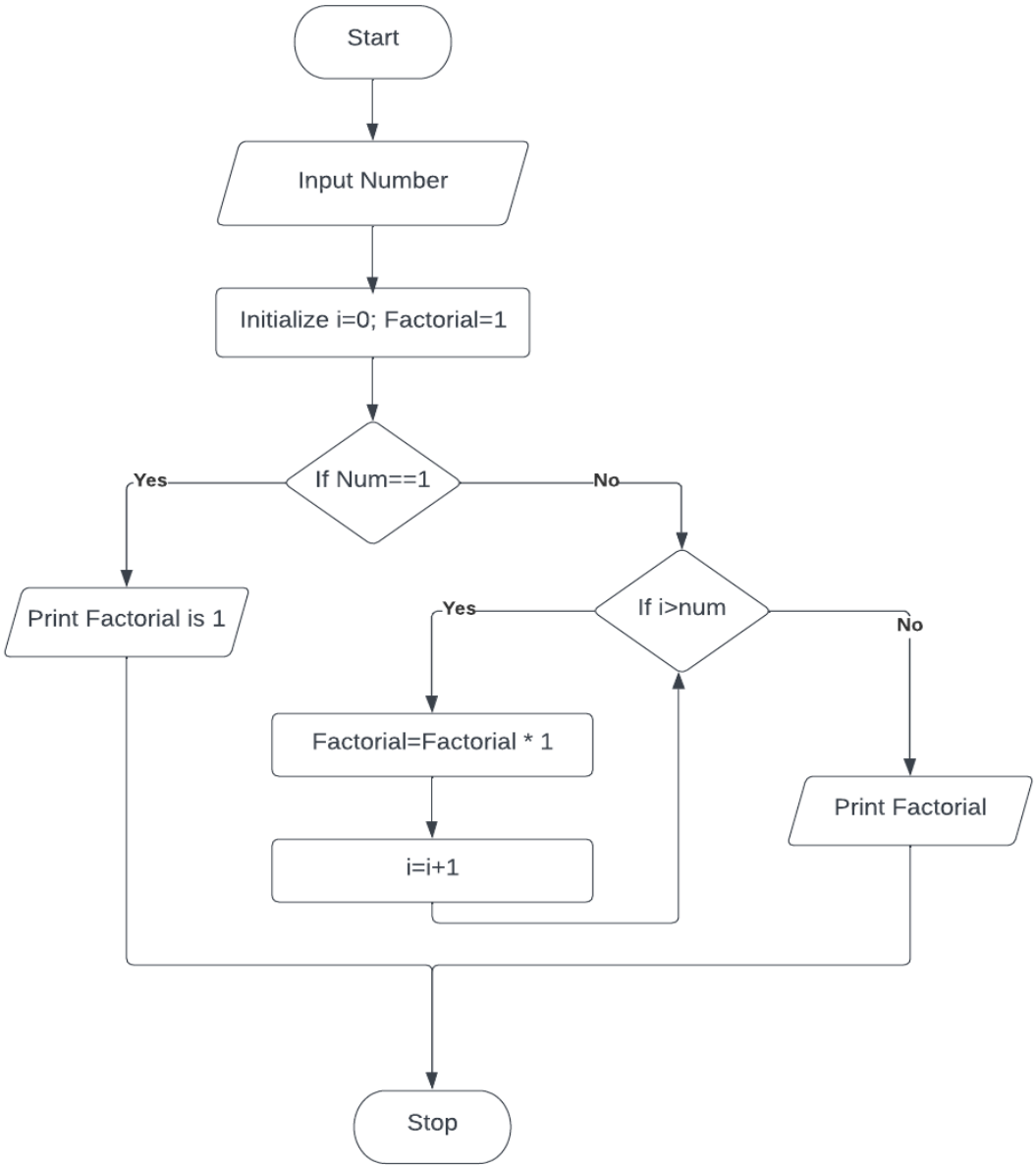
Date: 03.03.2023
Submission date: 13.03.2023
Write Algorithm or Flowchart for the following programs.

Q1. Check if the given number is EVEN or ODD.



ALGORITHM:
1. Start
2. Input Number (num)
3. if (num%2==0)
4. Print "EVEN"
5. Else Print "ODD"
6. End

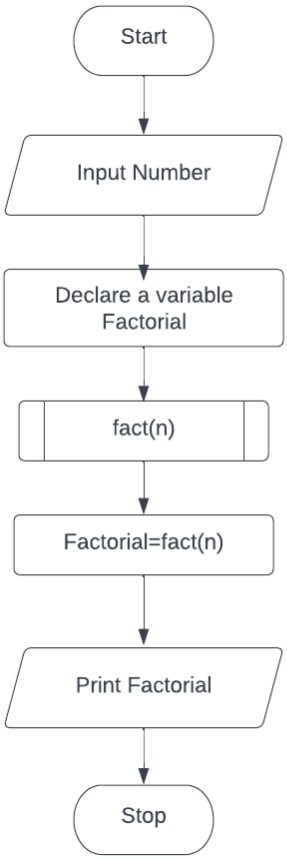
Q2. Write a Java Program to find the Factorial of a given number.

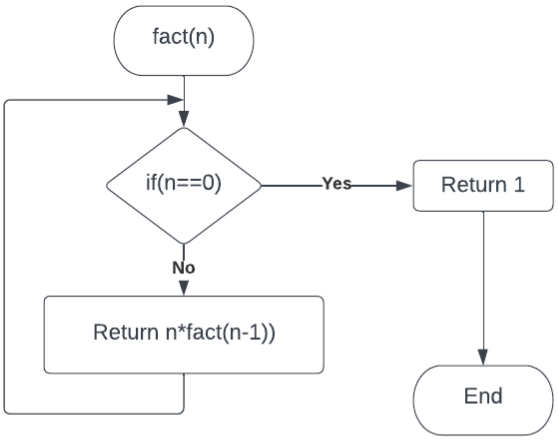


ALGORITHM:

1. Start
2. Enter Number (num)
3. Initialize i=1, Factorial=1
4. if (num==1)
5. Print Factorial is 1 Goto step 11
6. if i>num goto step 10
7. Factorial <- Factorial*i
8. i=i+1
9. Goto Step6
10. Print Factorial
11. End

Q3. Find the Factorial of a number using Recursion.

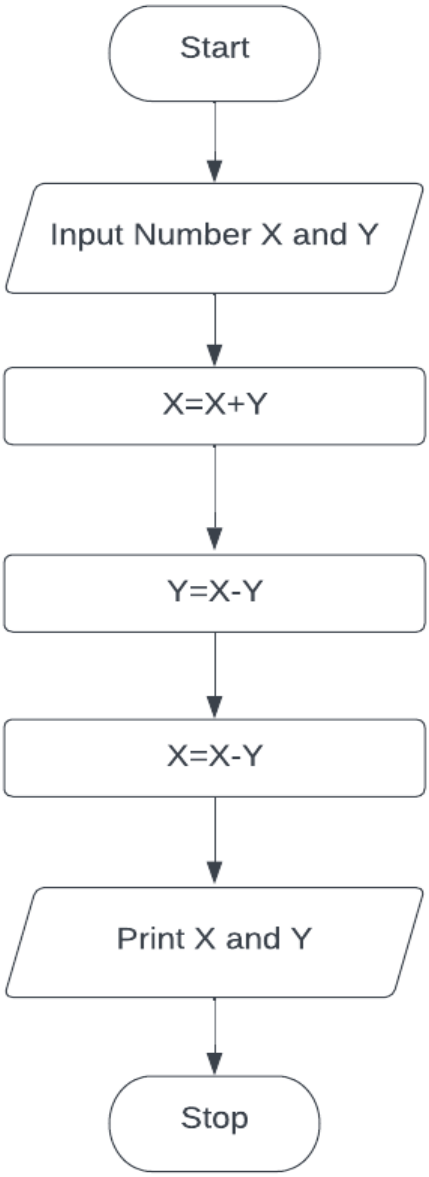




ALGORITHM:

1. Start
2. Input Number (num)
3. Create function "fact"
4. Pass num in fuction fact as an argument
5. if (n==0) then return 1
6. else return n*fact(n-1)
7. Print Factorial from Function fact
8. End

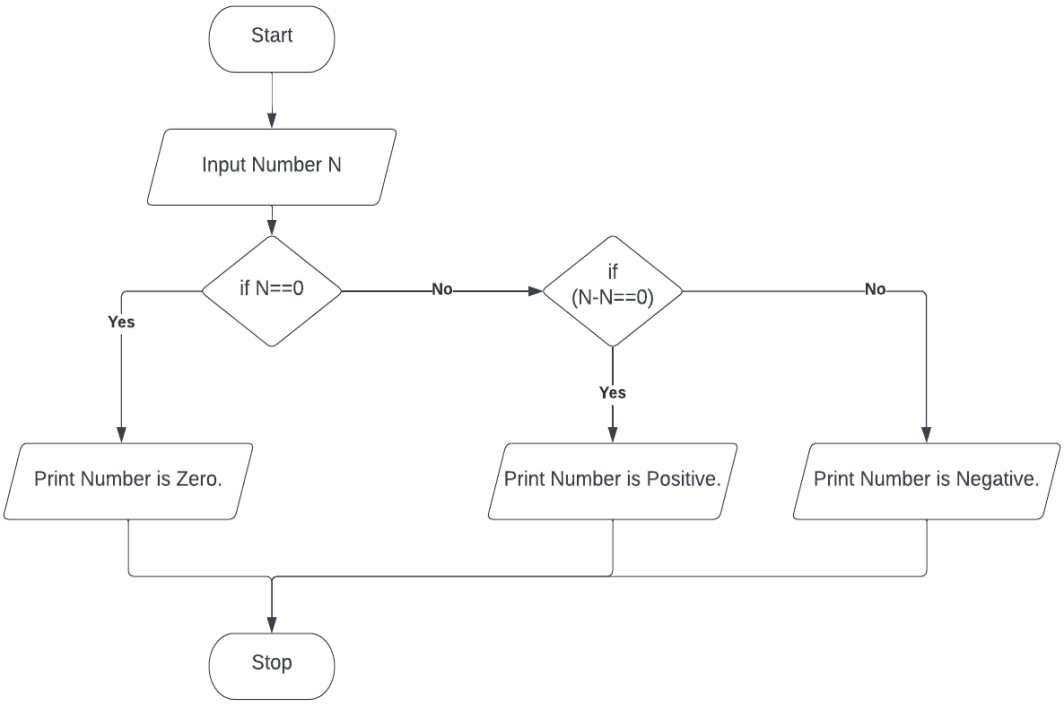
Q4. Swap two numbers without using the third variable approach.



ALGORITHM:

1. Start
2. Input Two Digits X and Y
3. X=X+Y
4. Y=X-Y
5. X=X-Y
6. Print After Swapping X= and Y=
7. End

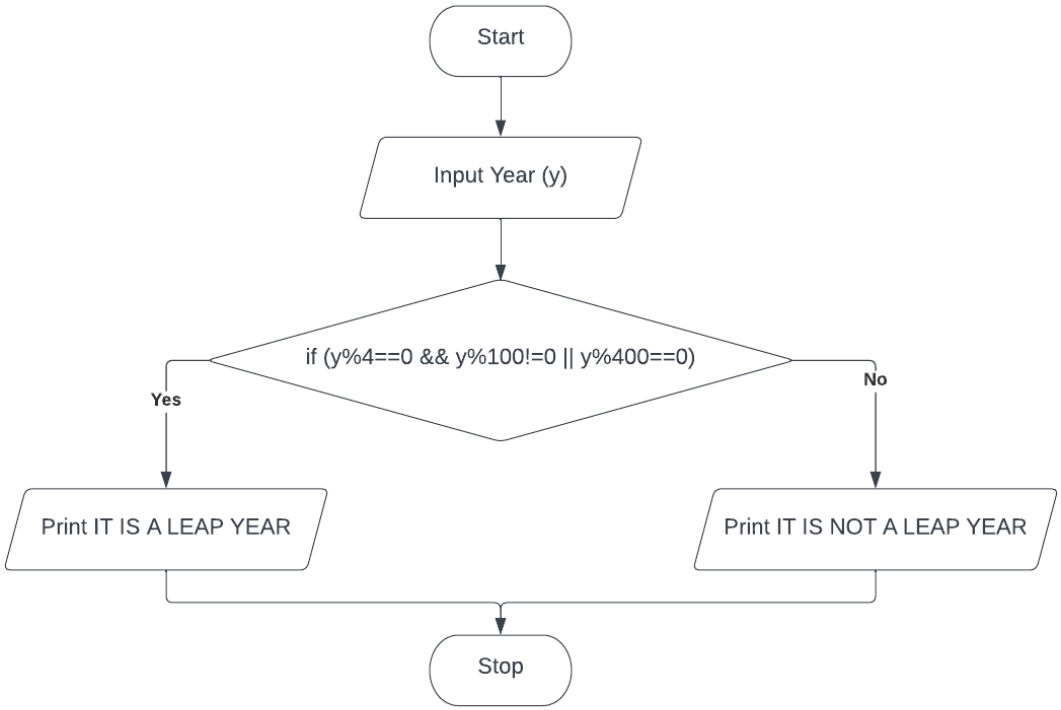
Q5. How to check whether the given number is Positive or Negative in Java?



ALGORITHM:

1. Start
2. Input Number (num)
3. if (num==0)
4. Print Number is Zero
5. else if ((num-num)==0)
6. Print the "Number is Positive"
7. Else Print "Number is Negative"
8. End

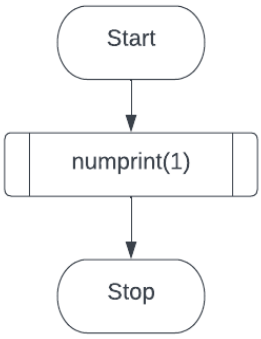
Q 6. Write a Java Program to find whether a given number is Leap year or NOT.

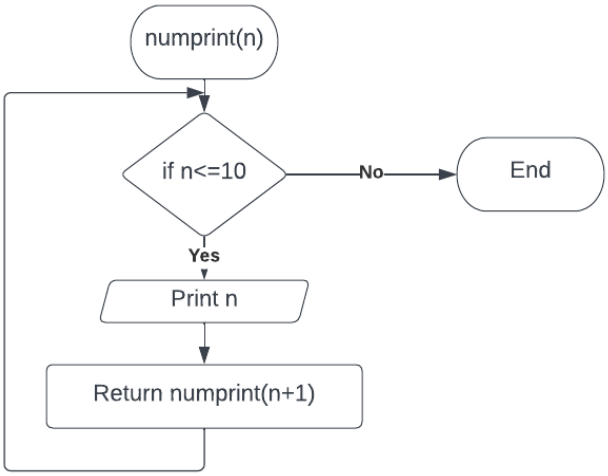


ALGORITHM:

1. Start
2. Input Year (y)
3. if (y%4==0 && y%100!=0 || y%400==0)
4. Print "It is a leap Year"
5. Else Print "It is not a leap year"
6. End

Q7. Write a Java Program to Print 1 To 10 Without Using Loop.

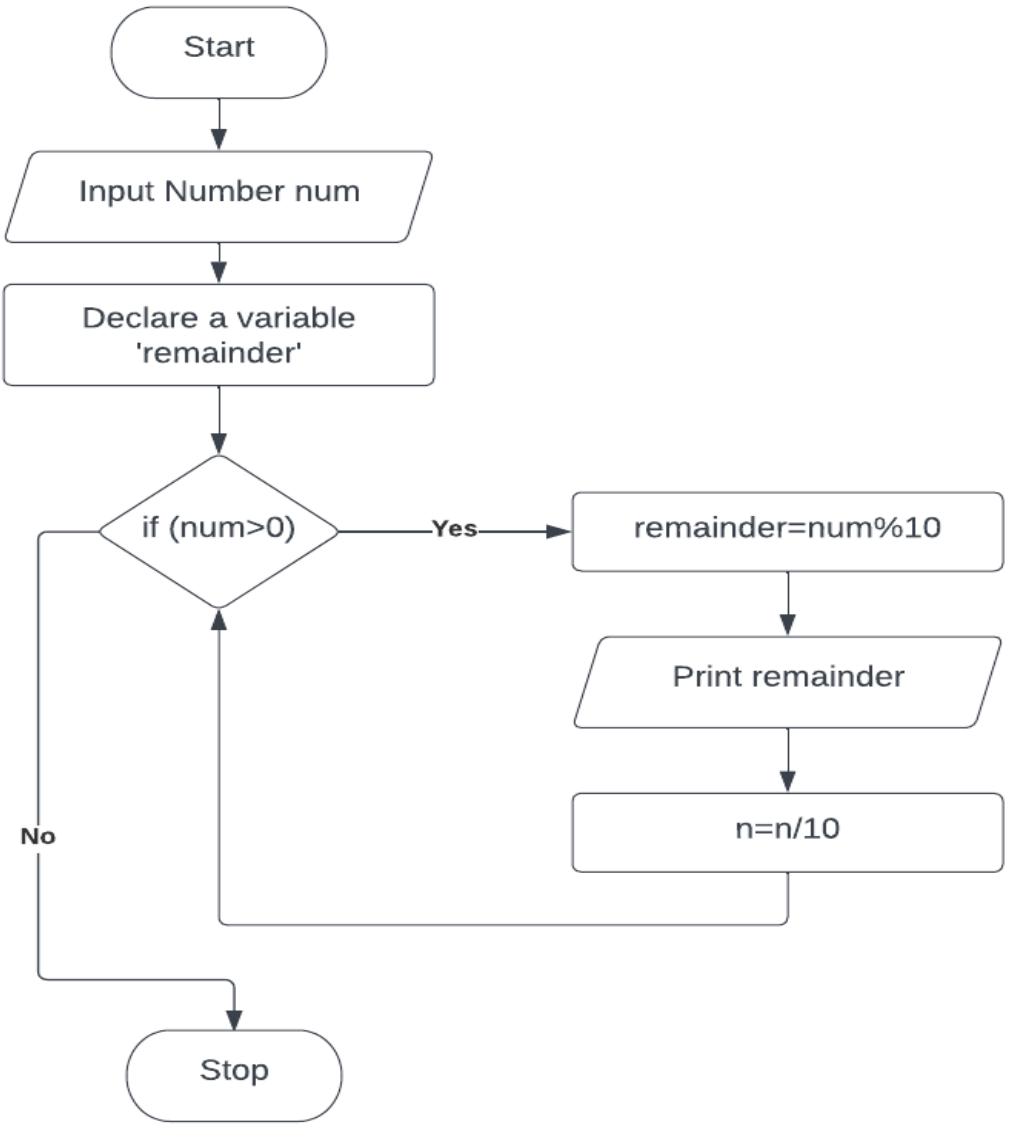




ALGORITHM:

1. Start
2. Create Function numprint (n)
3. Pass value 1 to the function numprint(n)
4. if (n<=10)
5. Print n
6. Return numprint(n+1)
7. End

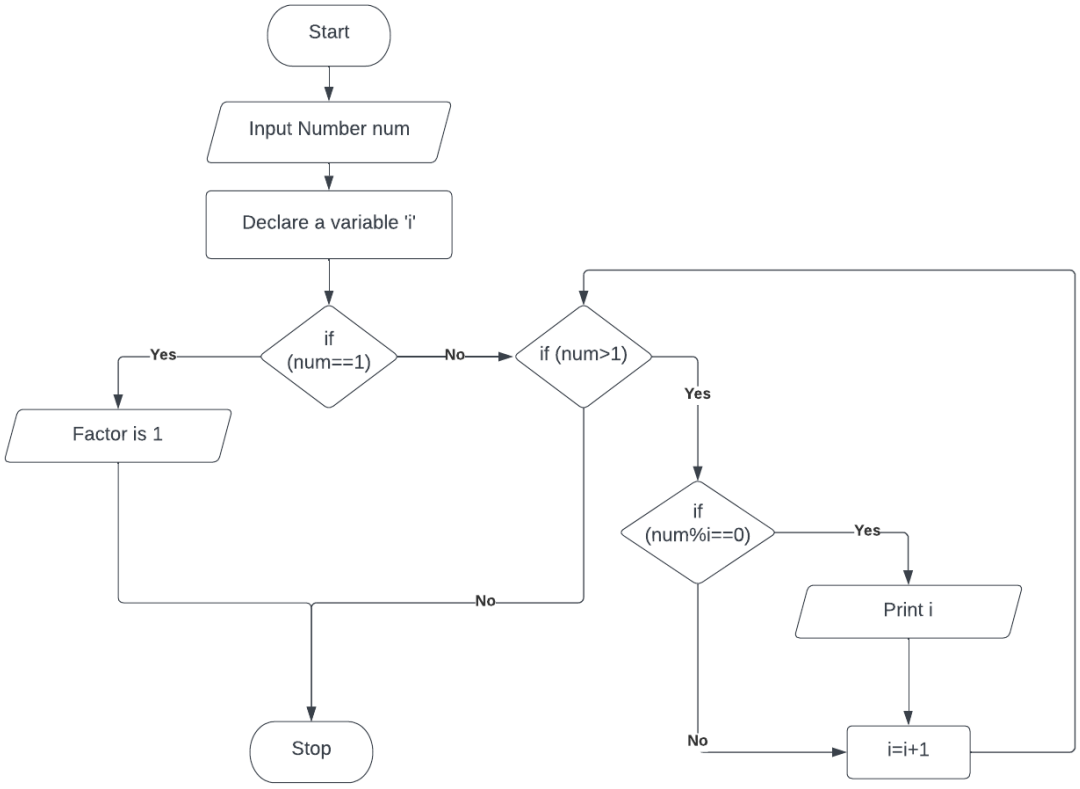
Q8. Write a Java Program to print the digits of a Given Number.



ALGORITHM:

1. Start
2. Input Number (num)
3. if (num > 0) Goto step 4 else Goto step 8
4. initialise remainder = num%10
5. Print remainder
6. num=num/10
7. Goto step 3
8. End

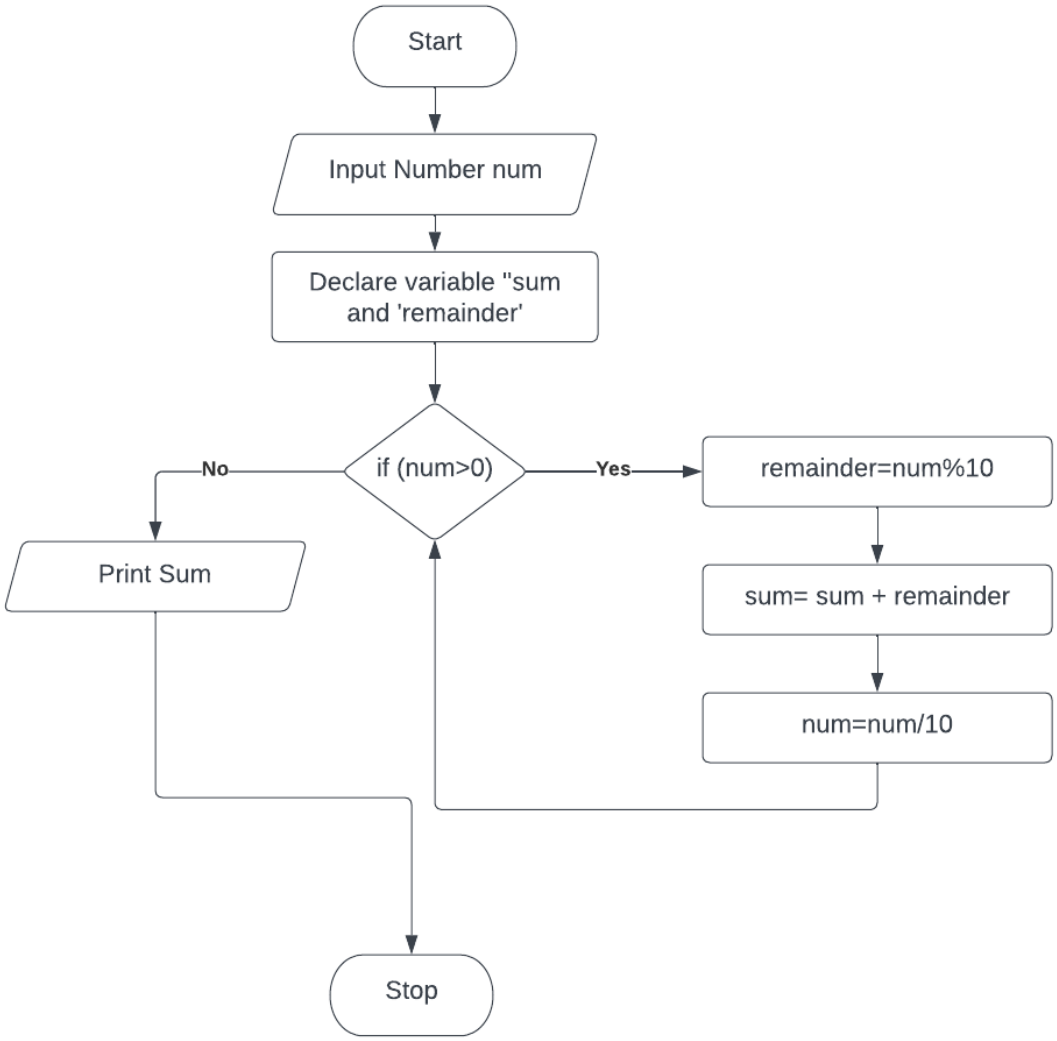
Q9. Write a Java Program to print all the Factors of the Given number.



ALGORITHM:

1. Start
2. Input Positive Number (num)
3. Initialize i=1
4. if (num==1)
5. Print Factor is 1
6. if num>i Goto Step 7 else goto step 11
7. if (num%i==0)
8. Print i
9. i=i+1
10. Goto Step 6
11. End

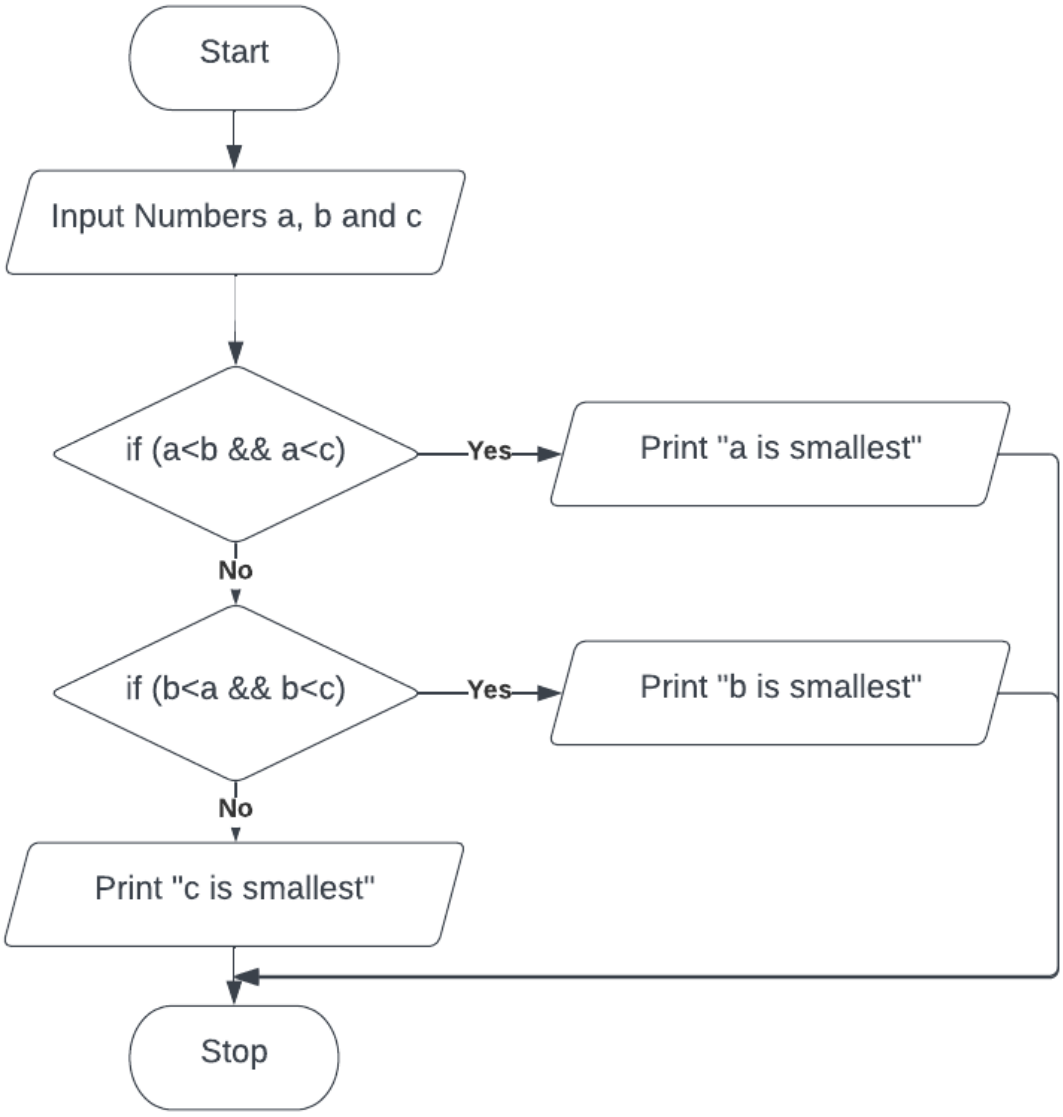
Q10. Write a Java Program to find the sum of the digits of a given number.



ALGORITHM:

1. Start
2. Input Number (num)
3. Intialize sum=0, remainder=0
4. if (num>0) goto step 5 else goto step 9
5. remainder = num%10
6. sum= sum + remainder
7. num=num/10
8. Goto Step 4
9. Print sum
10. End

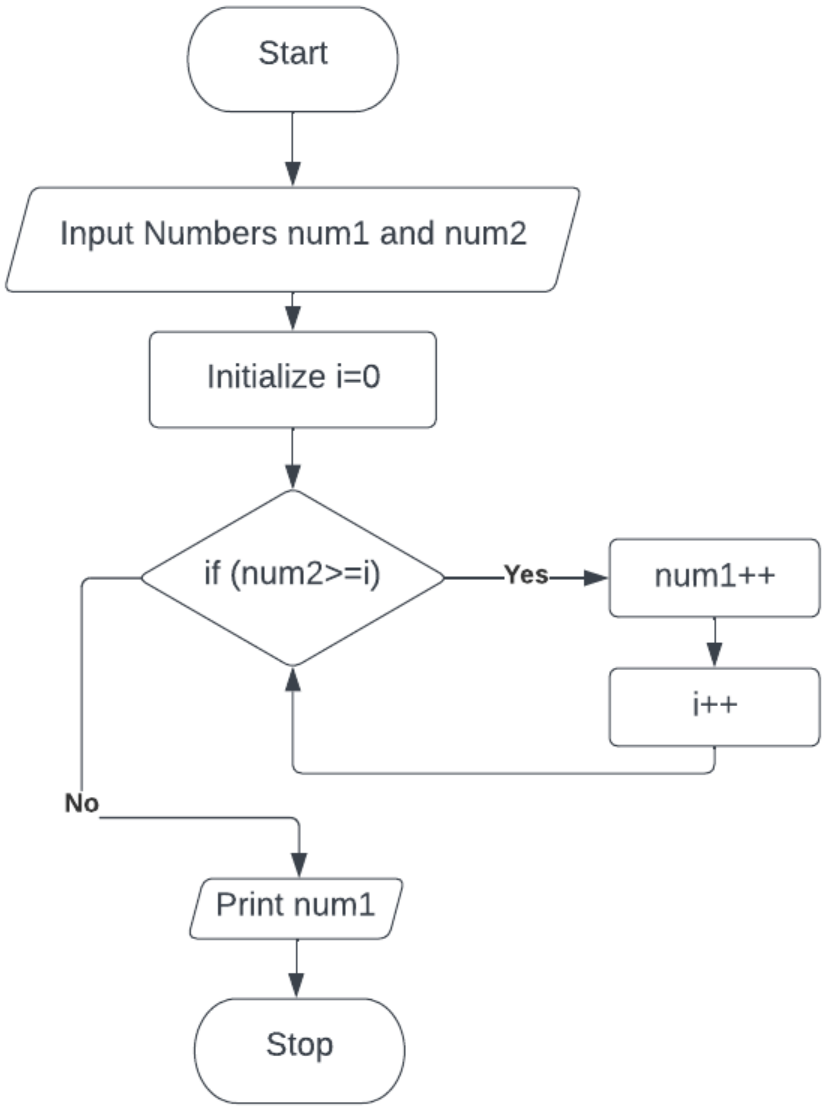
Q11. Write a Java Program to find the smallest of 3 numbers (a,b,c)



ALORITHM:

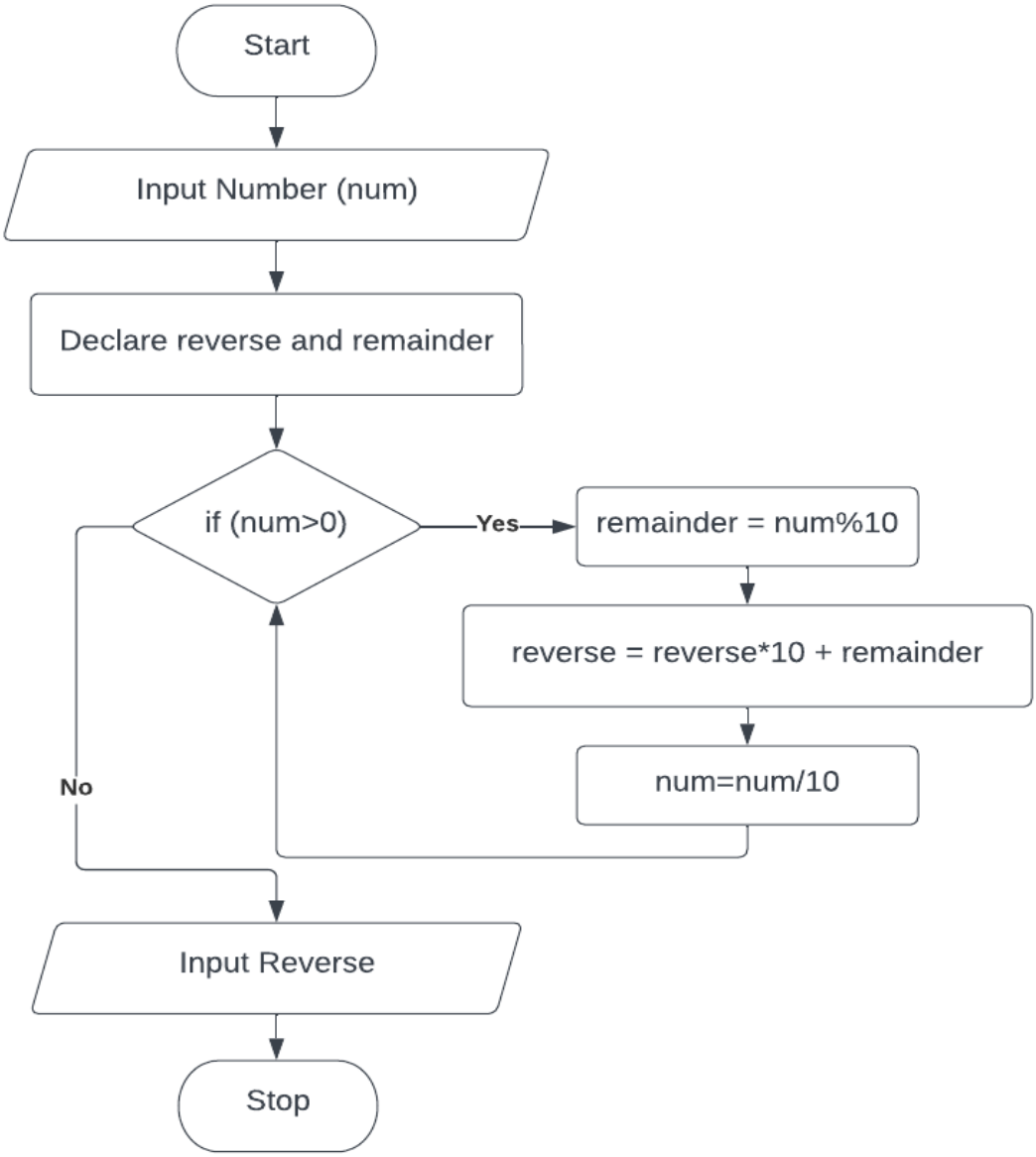
1. Start
2. Input the numbers a,b,c
3. if (a<b && a<c)
4. Print a is smallest
5. else if (b<a && b<c)
6. Print b is smallest
7. else Print c is smallest
8. End

Q12. How to add two numbers without using the arithmetic operators in Java?



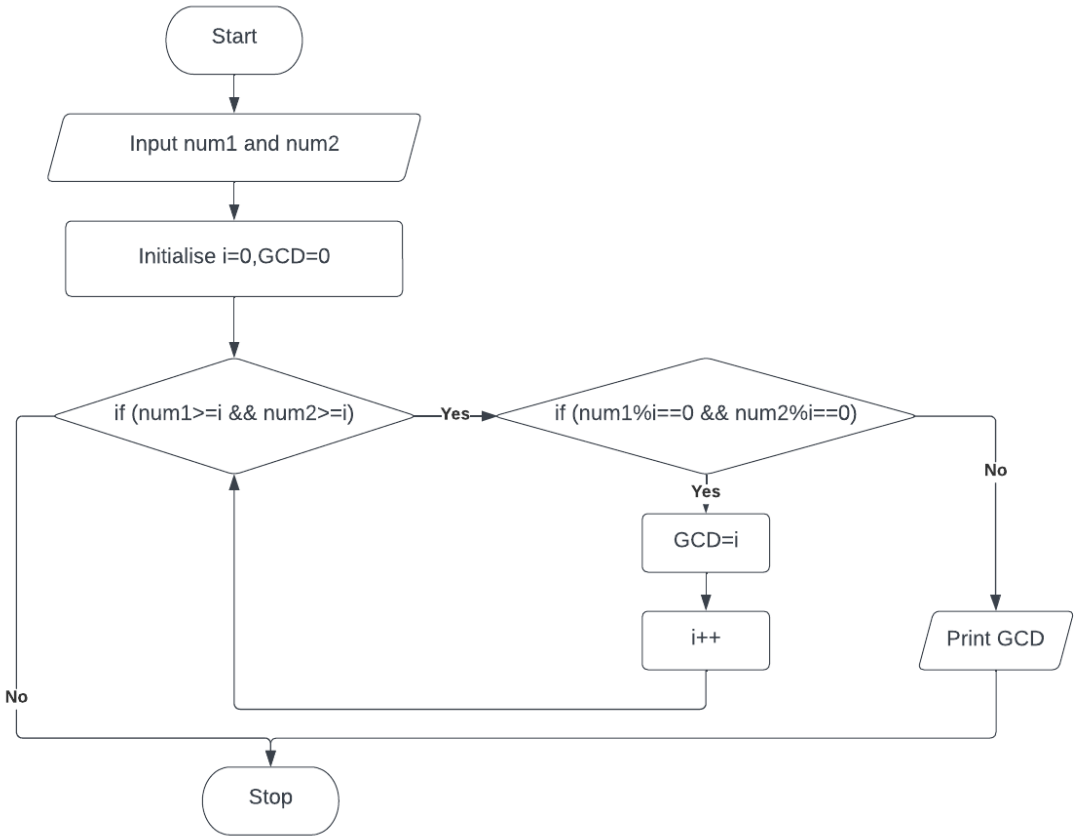
ALGORITHM:
1. Start
2. Enter num1 and num2
3. Initialize i=0
4. if (num2>=i) Goto Step 5 else Goto step 8
5. num1++
6. i++
7. Goto Step 4
8. Print num1
9. End

Q13. Write a java program to Reverse a given number.



ALGORITHM:
1. Start
2. Input Number (num)
3. Initialize reverse=0
4. if (num>0) Goto step 5 else Goto step 9
5. remainder = num%10
6. reverse = reverse*10 + remainder
7. num=num/10
8. Goto Step 4
9. Print Reverse
10.End

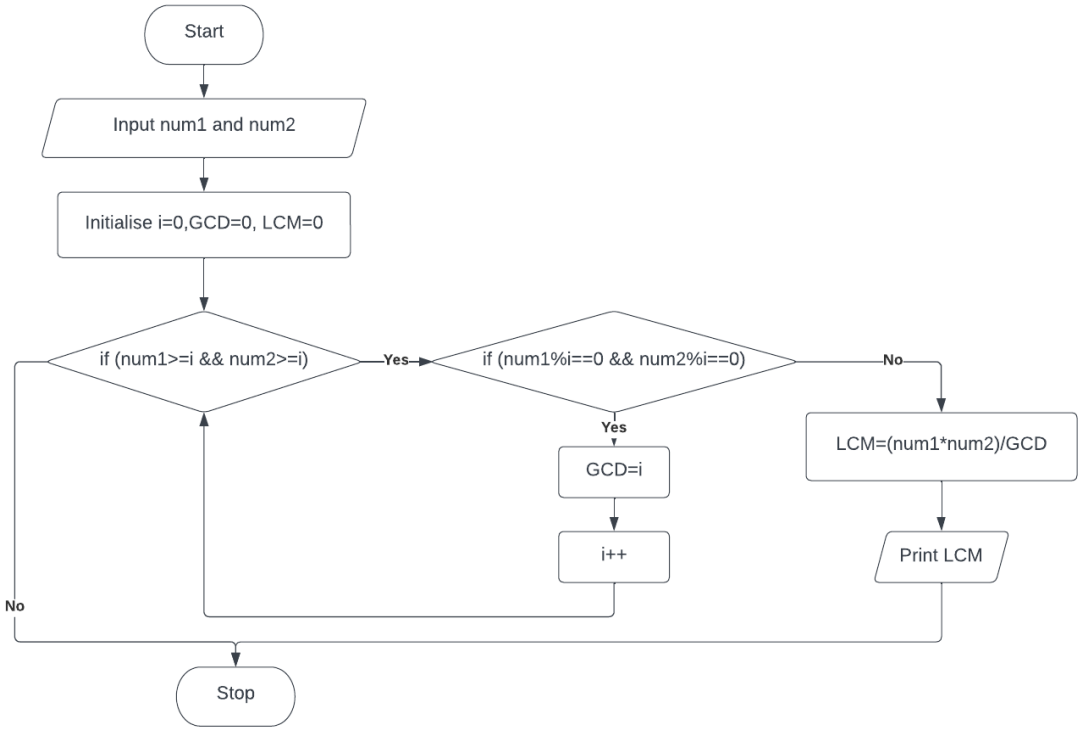
Q14. Write a Java Program to find the GCD of two given numbers.



ALGORITHM:

1. Start
2. Input num1 and num2
3. Initialise i=0, GCD=0
4. if (num1>=i && num2>=i) Goto Step 5 else goto step 9
5. if (num1%i==0 && num2%i==0)
6. GCD=i
7. i++
8. Goto Step 4
9. Print GCD
10. End

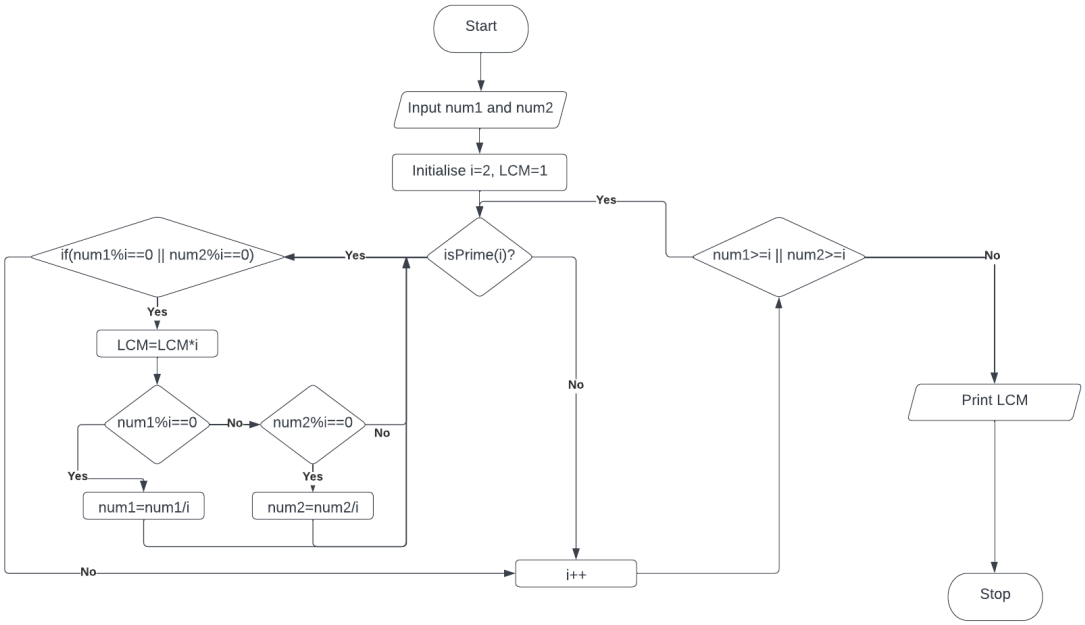
Q15. Write a java program to LCM of TWO given numbers.



ALGORITHM:

1. Start
2. Input num1 , num2
3. Initialise i=0, GCD, LCM
4. if (i>num1 && i>num2) Goto step 9 Else Goto step 5
5. if(num1%i==0 && num2%i==0)
6. GCD=i
7. i++
8. Goto Step 4
9. LCM = (num1*num2)/GCD
10. Print LCM
11. END

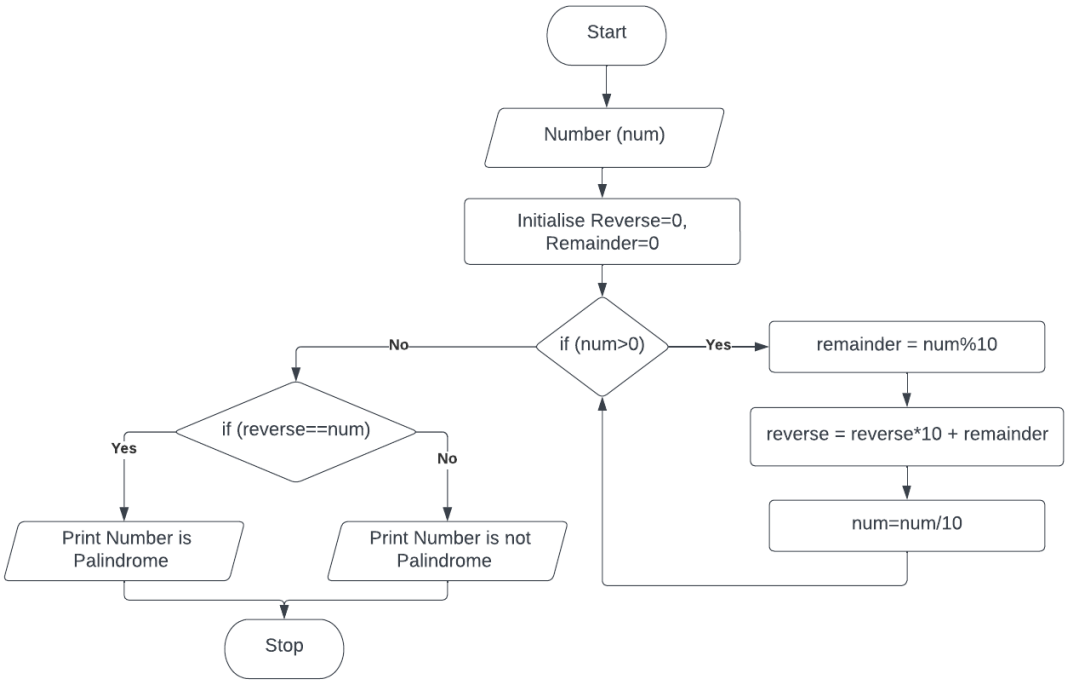
Q16. Write a java program to LCM of TWO given numbers using the Prime Factors method.



ALGORITHM:

1. Start
2. Input num1, num2
3. Initialize i=2, LCM=1
4. If isPrime(i) then Goto Step 5 Else Goto Step 11
5. if(num1%i==0 || num2%i==0) else Goto Step 11
6. LCM=LCM*i
7. if(num1%i==0) then num1=num1/i
8. Goto Step 5
9. else if(num2%i==0) then num2=num2/i
10. Goto Step 5
11. i++
12. if (num1>=i || num2>=i)
13. Goto Step 4
14. Else Print LCM
15. End

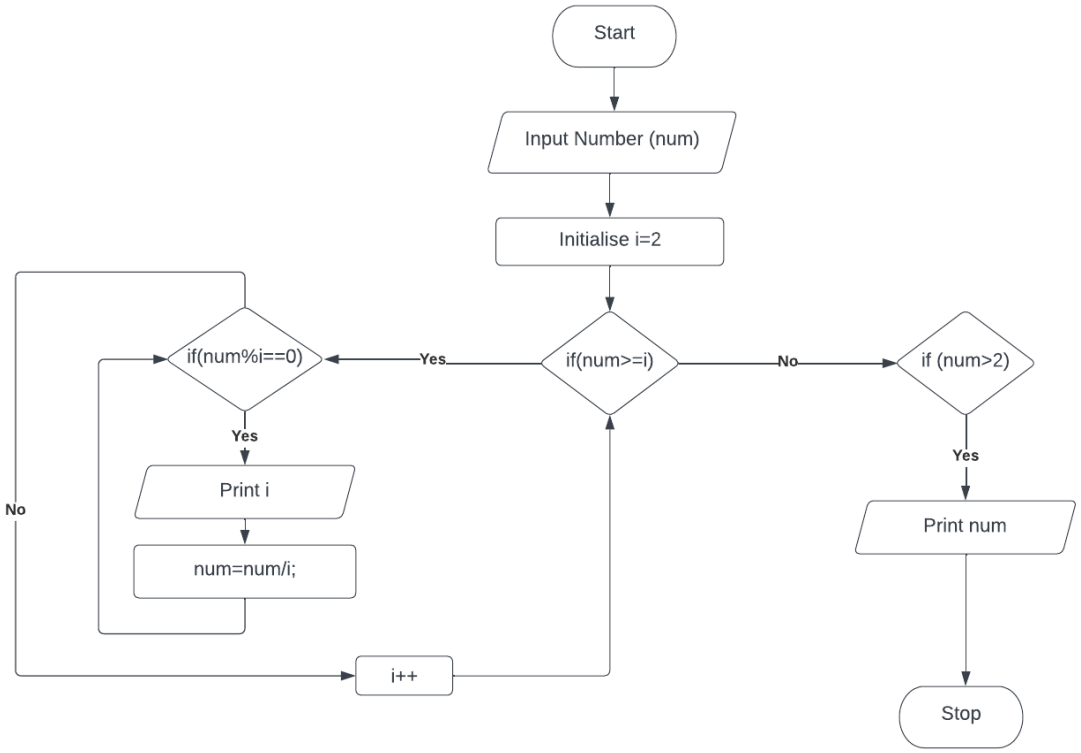
Q17. Check whether the Given Number is a Palindrome or NOT.



ALGORITHM:

1. Start
2. Input Number (num)
3. Initialize reverse=0, remainder=0
4. if (num>0) Goto step 5 else Goto step 9
5. remainder = num%10
6. reverse = reverse*10 + remainder
7. num=num/10
8. Goto Step 4
9. if (reverse==num)
10. Print Number is Palindrome
11. Else Print Number is not Palindrome
12. End

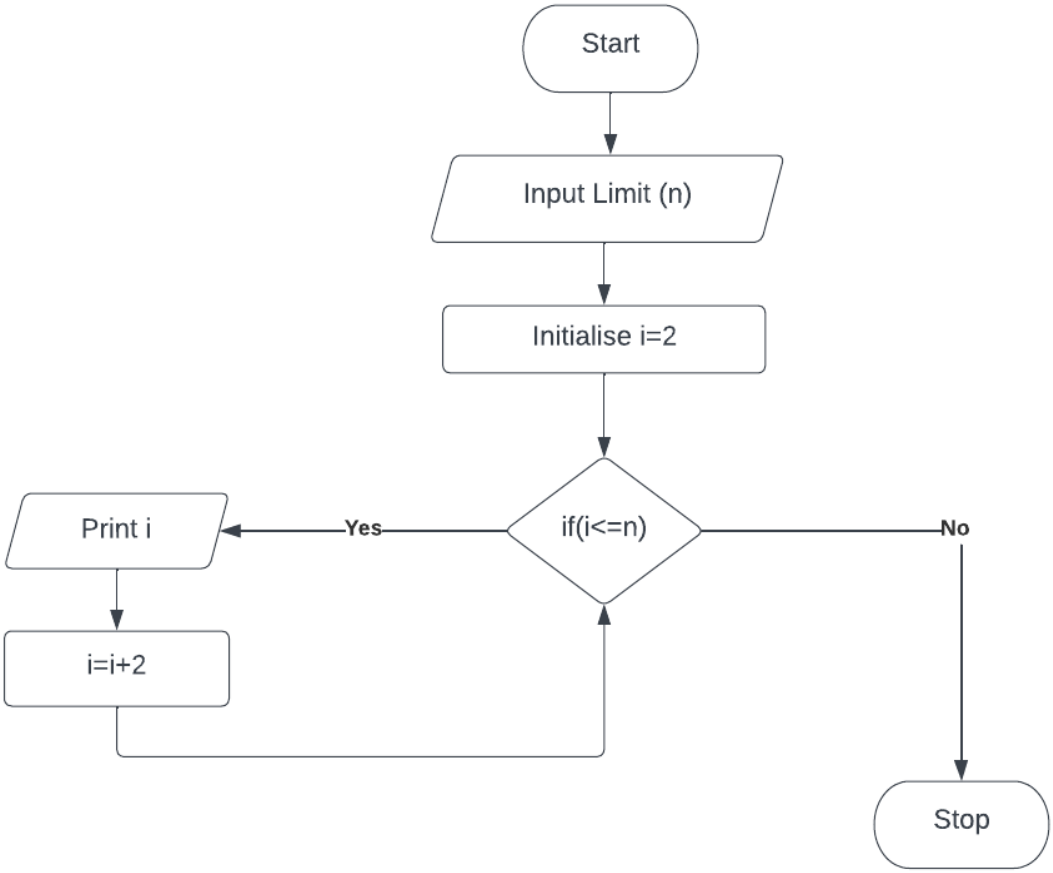
Q18. Write a Java Program to print all the Prime Factors of the Given Number.



ALGORITHM:

1. Start
2. Input Number (num)
3. Initialise i=2
4. if (i>num) Goto Step 10 else Goto Step 5
5. if(num%i==0)
6. Print i
7. num=num/i;
8. Goto Step 5
9. i++
10. Goto step 5
11. if (num>2)
12. Print num
13. End

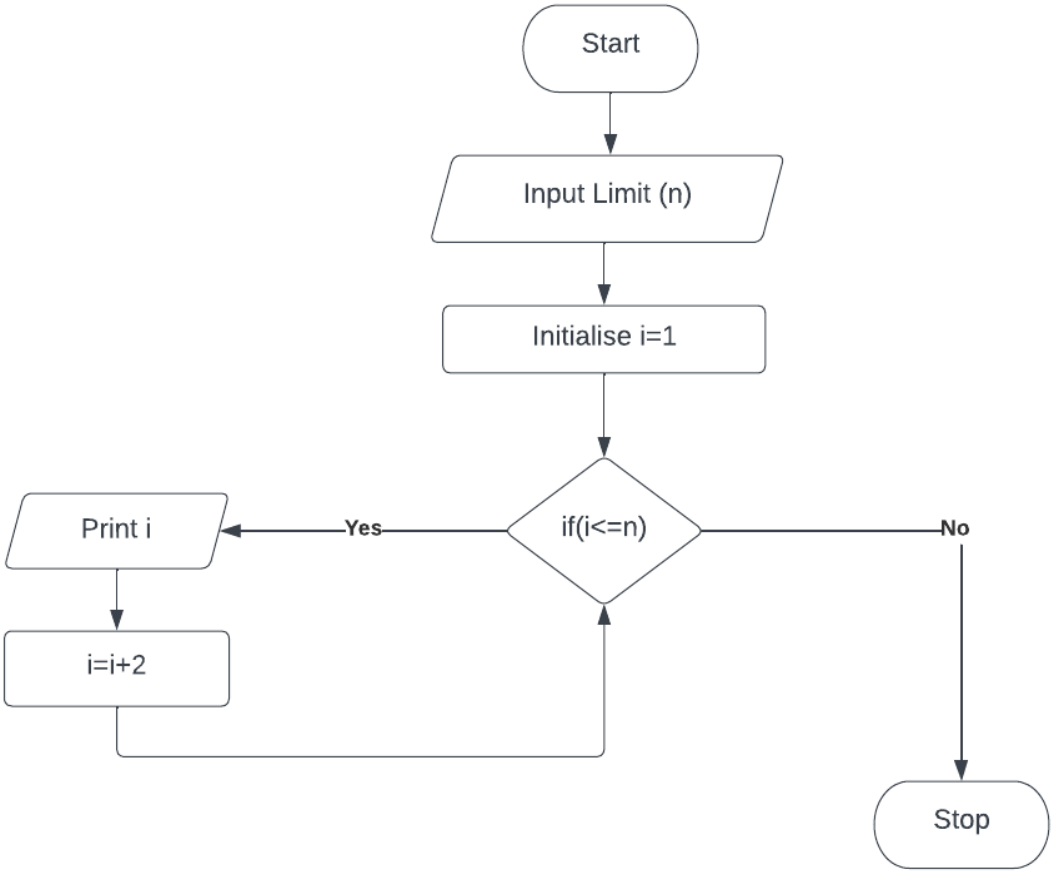
Q19. To print the following series EVEN number Series 2 4 6 8 10 12 14 16 .....



ALGORITHM:

1. Start
2. Initialize i=2, & Take Limit (n)
3. if (i<=n)
4. Print i
5. i=i+2
6. Goto Step 3
7. End

Q20. To print the following series ODD number Series 1 3 5 7 9 11 13...



ALGORITHM:

1. Start
2. Initialize i=1, & Take Limit (n)
3. if (i<=n)
4. Print i
5. i=i+2
6. Goto Step 3
7. End