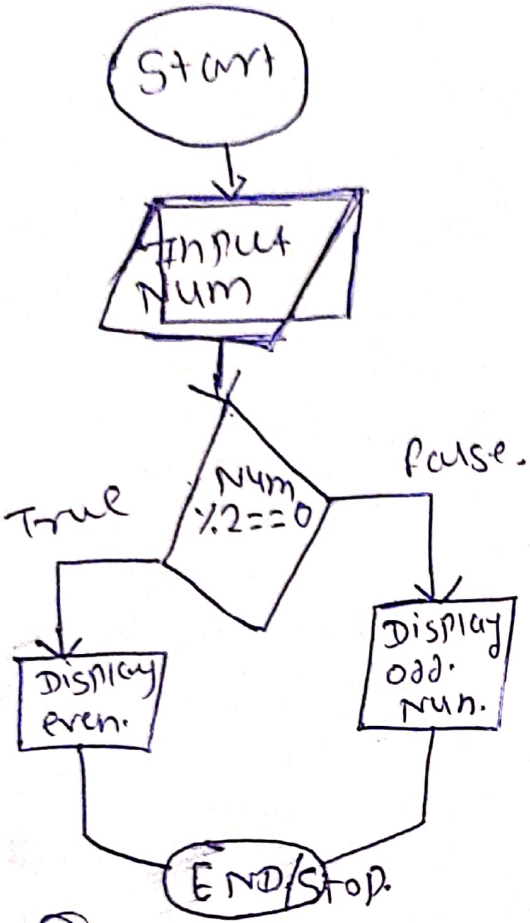
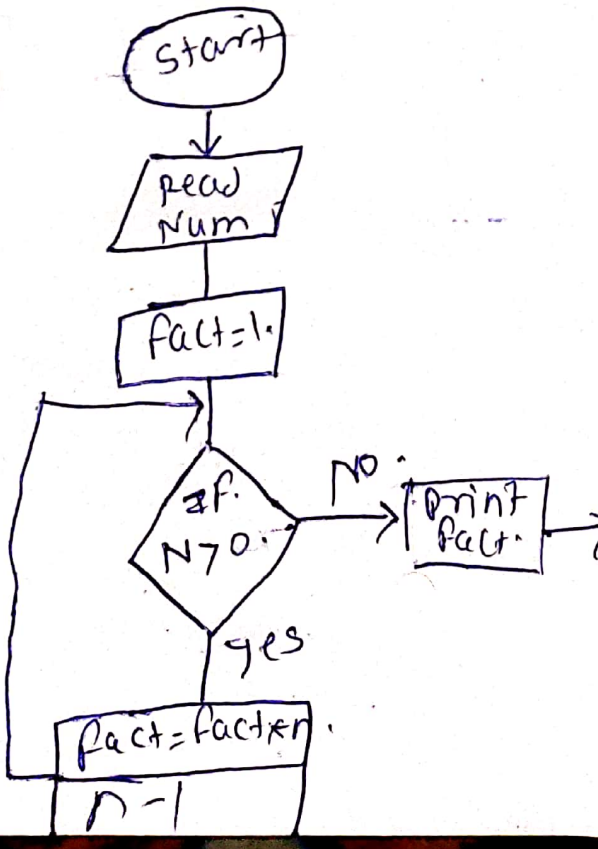
Assignment No. 1

① check if the given Number is Even or odd.



① Start
② To take a input or Read a Number.
③ Condn, If Num. is divisible by 2. then display even No.
④ If Num. is Not divisible by 2 the display odd No.
⑤ Stop.

② write a Java program to find the factorial of a given number.



① Start.
② Read the input of Num
③ Declear the variable fact
④ If Num>0. it will ~~fact~~ multiply the fact to Num.
⑤ then after Num-1 & goto return the condn. N>0.
⑥ If N=0 the print. the fact
⑦ Stop.

(3) Factorial of No. using Recursion.

a] Start
b] Enter No.
c] Give pass No. to Function.

```
int Fact(n).
{ if (n==1)
    return 1;
  else
    return n * Fact(n-1); }
```

d] END.

(4) Swap two Number without using 3rd variable.

→ 1. Start
2. Enter num1, ~~num~~ num2.
3. num1 = num1 + num2.
4. num2 = num1 - num2.
5. num1 = num1 - num2.
6. Display num1 & num2.
7. end.

(5) Check given no. is positive or -ve.

1. start.
2. Enter Num.
3. If (Num >=0)
4. cond^n true → Num is +ve.
5. cond^n false → Num is -ve.
6. end.

6] Given Number is leap year or NOT.

→ ① Start programe.
② take input of year
3. Check leap year = Given year / 4.
4. If remainder is 0, Then print → It is leap year.
5. Else print → It is non leap year.
6. End programe.

7. print No. 1 to 10 without using loop

→ 1. start.
2. Create array of size 10 & (store value) 10
3. Display value of each index.
4. end.

8] print the digits of given Number

→ 1. start.
2. Enter num & while (Num > 0).
3. i = (Num % 10).
4. display i
5. Number = Num / 10.
6. End.

(9) print all the factors of given number

1. Start
2. Enter Num.
3. For int i = 1 to i <= n/2 & increment i by 1.
4. if (n % i == 0).
5. If cond^n true display i
6. End.

(10) sum of digit of given Number

→ (1) Start.
2. Enter the Num ~~3, sum = 0.~~
3. 3. sum = 0.
4. while (Num > 0). then
{ int i = Num % 10.
sum = i + sum
Num = Num / 10. }
5. Display sum.

(11) Find the smalles Num of abc.

1. Start.
2. Enter Num a, b, c.
3. if a < b & a < c.
4. display a.
5. elase compar the (b < c). cond^n is true. &
6. If b is smaller than c. then display b
7. elge cond^n is false display c.
8. end.

(12) Add Number without using arithmatic operation.

1. Start.
2. Enter Num1 & Num2.
3. while (Num2 != 0).
4. carry is equal to Num1 bitwise AND Num2.
5. Num1 = Num bitwise x OR Num1.
6. Num2 = carry << 1.
7. display Num1
8. End.

(13) Reverse Given Number.

1. Start.
2. Enter Num
3. while (Num > 0).
4. i = Num % 10.
5. display i.
6. Num = Num/10.
7. End.

(14) Find GCD of two given Number

1. Start.
2. Enter Num1, Num2.
3. GCD = 1 & i
4. For i = 1 to i <= num1 && i <= num2 & increment i by 1.
5. if (num1 % i == 0) && (num2 % i == 0).
6. display GCD = i
7. End.

15. LCM of two given number

→ 1. Start.
2. Enter Num1, Num2.
3. LCM = (n1 > n2) ? n1 : n2
4. while true.
5. if (LCM % num1 == 0 & LCM % num2 == 0 then display LCM & break.
6. if not then increment LCM by 1.
7. End.

16. Check given number is palindrome or not.

→ 1. Start.
2. Enter Num
3. original Num = Num.
4. while (num != 0).
{ int remainder = num % 10;
reverse ~~Num~~ Num = reverse * 10 + remainder
num = 10;
}
5. if (original Num = reverse Num) then display given no. is palindrome.
6. if con't false given no. is not palindrome.

19.

(19). Print even number series.

1. Start.
2. Set i = 2.
3. while (true).
4. display i.
5. i = 2 + i
6. end.

20. Print odd number

→ 1. start.
2. Set i = 1.
3. while Due.
4. display i.
5. i = 2 + i
6. end.

(18). print all prime factor of number

→ ① start
2. Enter Num.
3. For i = 2 to i <= num & i increase by 1.
4. if (num % i == 0).
   isprime = 1.
5. For j = 2 to j <= i2 & j increase by 1.
6. if (i % j == 0) set is prime = 0 & break.

7. if(isprime == 1) then display i.

8. end.

(16) write a java program to LCM of two given Number using the prime factors method.