Loop Level 2

1. Take a number from the user and calculate its factorial.

Input	Output
5	5!=120
0	0!=1
1	1!=1

2. Take the value of x and y from the user and calculate x^y without using the pow() function (x can be a decimal number but y must be a positive integer).

Input	Output
2.5 5	97.65625
10 0	1
2 10	1024
-3 3	-27

3. Take two integers from the user and print their highest common factor (HCF).

Input	Output
75 100	25
50 49	1
121 99	11

4. Take two integers from the user and print their lowest common multiple (LCM).

Input	Output
15 10	30
13 5	65
33 99	99

5. Take a number from the user and check if it is a prime number. A prime number is a positive integer divisible by exactly two numbers, 1 and the number itself.

INPUT	OUTPUT
5	Prime
9	Not prime
121	Not prime
1	Not prime
17	Prime

6. Take a number from the user and check if it is a perfect number. A perfect number is a positive integer that is equal to the sum of its positive divisors excluding the number itself. 28 is a perfect number because the sum of its divisors 1+2+4+7+14 is equal to 28.

INPUT	OUTPUT
6	Perfect
10	Not perfect
496	Perfect

7. Take a number from the user and check if it is an Armstrong number. An Armstrong number is an integer such that the sum of the cubes of its digits is equal to the number itself. 153 is an Armstrong number because $1^3+5^3+3^3$ is equal to153.

INPUT	OUTPUT
0	Armstrong number
12	Not an Armstrong number
370	Armstrong number
100	Not an Armstrong number

8. Take a number from the user and check if it is symmetric (palindrome). A number is symmetric if the reverse of the number is equal to the original number.

INPUT	OUTPUT
123	Not symmetric
505	Symmetric
89	Not symmetric
77	Symmetric

9. Take a number from the user and show the digits in separate lines.

INPUT	OUTPUT
825	8 2 5
13	1 3
0	0

10. Take the number of terms, n, from the user and display the following series up to the nth term and its sum.

$$1+2+4+7 \cdots + n^{th}\ \text{term} = ?$$

INPUT	OUTPUT
5	1+2+4+7+11=25
8	1+2+4+7+11+16+22+29=92

11. Takes the number of terms, n, from the user and display the following series up to the nth term and its sum.

$$1-2+3-4 \cdots \pm n^{th}\ \text{term} = ?$$

INPUT	OUTPUT
10	1-2+3-4+5-6+7-8+9-10=-5
3	1-2+3=2

12. Write a program that takes the number of terms, n, from the user and prints the first n terms of the Fibonacci sequence. In the Fibonacci sequence, any term is equal to the sum of the previous two terms (except for the first two terms).

0, 1, 1, 2, 3, 5, 8, 13, 21…

INPUT	OUTPUT
5	0, 1, 1, 2, 3
11	0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55

13. Take a number from the user and check if it is a term in the Fibonacci sequence. If it is one of the terms in the Fibonacci sequence, print the term number.

INPUT	OUTPUT
8	7th term of the Fibonacci sequence
15	Does not exist in the Fibonacci sequence
55	11th term of the Fibonacci sequence

14. Take a number from the user and display its divisors.

INPUT	OUTPUT
12	1,2,3,4,6,12
23	1,23

15. Write a program that asks the user to enter how many inputs to take, then takes the specified number of inputs and counts the number of positive, negative, zero, odd and even numbers.

INPUT	OUTPUT
Number of inputs: 5 Enter 5 numbers: -7 0 4 7 0	Positive: 2 Negative: 1 Zero: 2 Odd: 2 Even: 3
Number of inputs: 10 Enter 10 numbers: 6 23 0 -1 88 5 -10 31 2 0	Positive: 6 Negative: 2 Zero: 2 Odd: 4 Even: 6