Sample problems (Easy):

1. Write a program to determine whether a number is prime or not.(O(sqrt(N))) time. You must have explanation of your code.)

Input	Output
5	Prime
7	Prime
9	Not Prime

2. Write a program to print/generate all the primes up to a given range N (Using sieve method).

Input	Output
10	2 3 5 7
30	2 3 5 7 11 13 17 19 23 29

3. Write a program to find all the prime factors of a number (Prime factorization).

Input	Output
6	2 3
10	2 5
12	2 2 3

4. Write a program to find the number of divisors of a number (NOD).

Input	Output
12	6
15	4

5. Write a program to calculate the Fibonacci sequence up to a given range.

Input	Output
10	0 1 1 2 3 5 8
20	0 1 1 2 3 5 8 13

6. Write a program to find the GCD and LCM of two numbers.

Input	Output
12 8	GCD : 4
	LCM : 24

Sample problems (Medium):

1. **Perfect number**, a positive integer that is equal to the sum of its proper divisors. The smallest **perfect number** is 6, which is the sum of 1, 2, and 3. Other**perfect numbers** are 28, 496, and 8,128.
   Now write a program that will determine whether a given number is Perfect or not

Input	Output
6	Perfect
28	Perfect
100	Not Perfect

2. An **Armstrong number** of three digits is an integer such that the sum of the cubes of its digits is equal to the **number** itself. For example, 371 is an **Armstrong number** since 3**3 + 7**3 + 1**3 = 371.

   Write a program that will determine whether a number is perfect or not.

Input	Output
1634	Armstrong Number
1234	Not Armstrong Number

3. Write an efficient C program to find the sum of contiguous subarray within a one-dimensional array of numbers which has the largest sum.

Input	Output
8	
-2 -3 4 -1 -2 1 5 -3	7

(**This is a greedy algorithm implementation. The link is given below. Understand the algorithm first, then implement it. https://www.geeksforgeeks.org/largest-sum-contiguous-subarray/)

Sample Problems (Hard- Not quite) :

1. An even number can be expressed as the sum of two prime numbers. Now your job is to design such a program that will take an even number as an input and print those two prime numbers which sum up and make that number. If the input is odd, or less than 4, simply print -1.

Input	Output
22	19 3
10	7 3
1	-1

N.B: The input and output format has to be as given under the problem statements. Prompts such as - “Enter a number” or “The result is:” are also not allowed.