* **Natural Sum**

Link : <https://practice.geeksforgeeks.org/problems/stuti-and-her-problem5846/1>

**My Initial Approach:**

I initially used two variables, one for sum and another for counter. I employed a for loop to iterate and accumulate the sum. However, this approach didn't pass several test cases.

**Correct Approach:**

The correct approach is much more elegant. By utilizing a single for loop and the formula for the sum of the first N natural numbers (N \* (N + 1) / 2), we can achieve the desired result more efficiently.

**Time complexity : O(N)**

* **Prime Number**

Link : <https://practice.geeksforgeeks.org/problems/prime-number2314/1>

**Initial Approach:**

My first attempt involved looping from 2 to N/2 and checking if N is divisible. While functional, this approach turned out inefficient, causing test cases to stumble due to time constraints.

**Optimized Technique:**

The lightbulb moment arrived when I realized that checking up to N/2 wasn't necessary. By running a loop up to the square root of N, I could efficiently identify divisors. If a divisor is found, the number isn't prime.