A Recursive Formula for the n-th Prime Number

Samagra Ganore

20th September 2025

Abstract

This paper introduces a novel recursive formula for computing the n-th prime number, denoted as $p(n)$. The approach leverages a combination of summation, floor functions, and inclusion-exclusion principles to define both $p(n)$ and an auxiliary function $f(m)$. Base cases are provided for small values, and the formula is presented for general n.

1 Introduction

The sequence of prime numbers has been a cornerstone of number theory since antiquity. Efficient methods for generating primes, such as the Sieve of Eratosthenes, exist for finding all primes up to a given limit. However, closed-form or recursive expressions for the exact n-th prime remain elusive in simple terms, often relying on approximations like the Prime Number Theorem.

In this work, we propose a recursive formula for $p(n)$, the n-th prime, defined mutually with an auxiliary function $f(m)$ that counts the number of primes less than or equal to m, plus . The formula incorporates logarithmic bounds inspired by prime number estimates and uses inclusion-exclusion over products of smaller primes to sieve composites.

The base cases are:

- $f(1) = 1$
- $p(1) = 2$
- $p(2) = 3$
- $p(3) = 5$
- $p(4) = 7$
- $p(5) = 11$
- $p(6) = 13$

These allow the recursion to bootstrap for larger values. The formula is intended for integer $n \geq 6$.

2 The Recursive Formula

We define the n-th prime $p(x)$ as follows:

$$p(x) = 1 + \sum_{n=1}^{\lfloor x(\ln(x)+\ln(\ln(x)))+1 \rfloor} \left\lfloor \left(\frac{x}{f(n)} \right)^{1/x} \right\rfloor$$

The auxiliary function $f(n)$ is defined recursively as:

$$f(n) = n + f(\lfloor \sqrt{n} \rfloor) + \sum_{k=1}^{f(\lfloor \sqrt{n} \rfloor)} (-1)^k \sum_{1 \le i_1 < i_2 < \cdots < i_k \le f(\lfloor \sqrt{n} \rfloor)} \left\lfloor \frac{n}{p(i_1) \cdot p(i_2) \cdots p(i_k)} \right\rfloor$$

For small x where $\ln(\ln(x))$ is undefined (e.g., $x \le 1$), use the base cases directly.

3 Derivation

To construct the recursive formula for the n-th prime, we begin by defining the auxiliary function $f(n)$, which represents the number of primes less than or equal to n plus one. The definition of $f(n)$ employs the principle of inclusion-exclusion to count numbers up to n that are not divisible by primes up to $\lfloor \sqrt{n} \rfloor$, effectively sieving out composite numbers.

Consider the set of integers from 1 to n. We aim to count numbers that are either prime or 1. Start with all integers up to n, i.e., n numbers. We then subtract numbers divisible by each prime $p(i)$ where $i \le f(\lfloor \sqrt{n} \rfloor)$, as composites with prime factors $\le \sqrt{n}$ cover all composites up to n. This gives terms like $\lfloor n/p(i_1) \rfloor$. To correct for double-counting, we add back numbers divisible by pairs of primes, subtract triples, and so forth, following the inclusion-exclusion principle:

$$\sum_{k=1}^{f(\lfloor \sqrt{n} \rfloor)} (-1)^k \sum_{1 \le i_1 < i_2 < \cdots < i_k \le f(\lfloor \sqrt{n} \rfloor)} \left\lfloor \frac{n}{p(i_1) \cdot p(i_2) \cdots p(i_k)} \right\rfloor$$

The recursive term $f(\lfloor \sqrt{n} \rfloor)$ accounts for the number of relevant primes at the next level down, ensuring the function builds upon smaller values. Then we add $f(\lfloor \sqrt{n} \rfloor)$ because the above inclusion and exclusion also subtracts the base primes. Thus, $f(n)$ is defined as:

$$f(n) = n + f(\lfloor \sqrt{n} \rfloor) + \sum_{k=1}^{f(\lfloor \sqrt{n} \rfloor)} (-1)^k \sum_{1 \le i_1 < i_2 < \cdots < i_k \le f(\lfloor \sqrt{n} \rfloor)} \left\lfloor \frac{n}{p(i_1) \cdot p(i_2) \cdots p(i_k)} \right\rfloor$$

The base case $f(1) = 1$ ensures the recursion terminates.

For $p(x)$, the x-th prime, we design a formula that counts primes by evaluating a function that distinguishes numbers based on their position relative to the x-th prime. The term $\left\lfloor \left(\frac{x}{f(n)} \right)^{1/x} \right\rfloor$ is crafted to yield 1 when $f(n) < x$, and 0 when

$f(n) \geq x$. This is because $\frac{x}{f(n)} > 1$ when $f(n) < x$, and raising to the power $1/x$ keeps the result below 2, so the floor is 1. Conversely, when $f(n) \geq x$, $\frac{x}{f(n)}1$, effectively contributing 0 after flooring.

To ensure we sum over enough terms to capture the x-th prime, we use an upper bound inspired by the Prime Number Theorem $x(\ln x + \ln\ln x)$, provides a safe overestimate for n where $f(n)$ transitions across x. Adding 1 ensures coverage for small x. Thus, the sum runs up to $\lfloor x(\ln x + \ln\ln x) + 1 \rfloor$. The final formula for $p(x)$ counts these 1s plus an offset:

$$p(x) = 1 + \sum_{n=1}^{\lfloor x(\ln(x)+\ln(\ln(x)))+1 \rfloor} \left\lfloor \left(\frac{x}{f(n)} \right)^{1/x} \right\rfloor$$

The base cases $p(1) = 2$, $p(2) = 3$, etc., handle small x where logarithmic terms may be undefined or insufficient.

4 Optimisation

To improve the computational efficiency of the recursive formula for $f(n)$, we consider an optimization for values of n less than the product of the first j primes, i.e., $n < p(1) \cdot p(2) \cdot \ldots \cdot p(j)$. The original definition of $f(n)$ involves a summation over k from 1 to $f(\lfloor \sqrt{n} \rfloor)$, which can be computationally expensive due to the potentially large number of terms, especially as $f(\lfloor \sqrt{n} \rfloor)$ grows with $\sqrt{n}$.

For $n < p(1) \cdot p(2) \cdot \ldots \cdot p(j)$, we can limit the inclusion-exclusion sum to only the first j primes, significantly reducing the number of terms. This is because, for such n, only primes up to the j-th prime contribute meaningfully to the divisors in the inclusion-exclusion process, as larger primes produce quotients $\lfloor n/(p(i_1) \cdot p(i_2) \cdots p(i_k)) \rfloor = 0$ when the product exceeds n.

Thus, we propose a modified formula for $f(n)$ when $n < p(1) \cdot p(2) \cdot \ldots \cdot p(j)$:

$$f(n) = n + f(\lfloor \sqrt{n} \rfloor) + \sum_{k=1}^{j} (-1)^k \sum_{1 \leq i_1 < i_2 < \cdots < i_k \leq f(\lfloor \sqrt{n} \rfloor)} \left\lfloor \frac{n}{p(i_1) \cdot p(i_2) \cdots p(i_k)} \right\rfloor$$

This optimization caps the outer summation at $k = j$, rather than $k = f(\lfloor \sqrt{n} \rfloor)$, reducing the number of combinations from $2^{f(\lfloor \sqrt{n} \rfloor)}$ to at most 2^j for the inclusion-exclusion terms. For example, if $j = 3$, and $p(1) = 2$, $p(2) = 3$, $p(3) = 5$, then for $n < 2 \cdot 3 \cdot 5 = 30$, we only consider combinations of the primes $\{2, 3, 5\}$, limiting the sum to $k = 1, 2, 3$. This significantly reduces computational complexity for smaller n, making the formula more practical for implementation while preserving correctness, as higher-order terms contribute zero for such n.

For larger n, the original formula can still be used, or j can be chosen dynamically based on the magnitude of n relative to the product of the first j primes. This optimization maintains the recursive structure and correctness of $f(n)$ while improving performance for a practical range of inputs.