The Prime Odometer: A Generative Law for Prime Emergence in Prime Exponent Space

Kenneth Hilton*

August 2025

Abstract

Primes have long been defined negatively—as numbers with no divisors—and generated catalogically by sieves of elimination. We introduce the *Prime Odometer*, a constructive successor law in Prime Exponent Space (PE-space) where primes arise not by rejection but by structural inevitability. At each tick $n \mapsto n+1$, the odometer updates the prime-exponent vector $\sigma(n)$ incrementally via p-adic carries. When redistribution fails, the successor vector is forced to a one-hot state $(0,0,\dots,0,1)$, certifying primality without external verification. We prove correctness by induction, show that the odometer enumerates primes in order, and contrast this generative sieve with Eratosthenes' eliminative sieve. This reframing provides a new theoretical lens: primes are the *canonical carries* of arithmetic, inevitable and positively generative.

1 Introduction

Prime numbers, the indivisible atoms of arithmetic, have been studied for millennia yet remain mysterious in their distribution. Traditionally they are defined *negatively*—as numbers greater than one with no divisors other than one and themselves. This absence-based definition has shaped centuries of number theory: primality is confirmed only by elimination of factors. Classical algorithms mirror this stance. The Sieve of Eratosthenes, for example, marks off composites to leave primes behind; modern primality tests probe for divisors or verify properties of multiplicative groups. In every case, primes are treated as the remainder after composites are discarded.

We propose a new, constructive view: primes can be generated *positively* as inevitable events in *Prime Exponent Space* (PE-space). In this representation, each integer n is encoded by its vector of prime exponents,

$$\sigma(n) = (e_1, e_2, e_3, \dots), \qquad n = \prod_i p_i^{e_i},$$

[*] Email: kenosis@gmail.com

a sparse coordinate system induced by the Fundamental Theorem of Arithmetic. Arithmetic operations become vector operations, and the successor map $n \mapsto n+1$ can be studied as a geometric transition through this lattice.

Our central thesis is that successor steps in PE-space behave like an odometer. Each prime axis p carries its own residue counter $r_p(n) \equiv n \pmod{p}$. When n advances to $n+1$, each counter increments:

$$r_p(n+1) = (r_p(n) + 1) \bmod p.$$

If a counter wraps to zero, then $p \mid (n+1)$ and the exponent of p in the factorization of $n+1$ is at least one. More generally, if $n \equiv -1 \pmod{p^t}$, then a carry of depth t occurs, and the valuation is determined exactly:

$$\nu_p(n+1) = t.$$

In words: the depth of the p-adic carry is the exponent of p in $n+1$. For most steps, redistribution across existing axes suffices. But at special thresholds—exactly the primes—all existing axes fail to absorb the increment. The odometer is forced to roll over into a new axis, and $\sigma(n+1)$ becomes one-hot, $(0, 0, \ldots, 0, 1)$. This canonical carry is the generative certificate of primality.

We call this mechanism the **Prime Odometer**. Unlike sieves of elimination, it is a *generative sieve*: it produces primes sequentially as structural necessities of counting, rather than as residues after testing. The Prime Odometer offers:

- A *constructive definition of primality*: primes are numbers whose successor vectors are forced one-hot.
- A *proof by induction*: maintaining residues for all known primes, the algorithm correctly generates $\sigma(n)$ for all n and identifies primes without external verification.
- A *dual to Eratosthenes*: whereas the classical sieve eliminates composites, the odometer constructs successors, and primes appear inevitably as carries.

This paper develops the Prime Odometer formally. Section 2 introduces the necessary preliminaries in PE-space and p-adic carry depth. Section 3 defines the odometer algorithm. Section 4 states and proves the Odometer Theorem by induction. Section 5 compares this constructive sieve with classical approaches. Sections 6 and 7 explore implications and open questions, and Section 8 concludes. An appendix provides pseudocode for implementation. The result is both a new algorithm and a new conceptual lens: primes as the canonical carries of arithmetic.

Remark. The Prime Odometer was first developed here independently, without reference to prior art on trial-division sieves. Subsequent literature review revealed its algorithmic equivalence to classical incremental trial division [1, 2]. The originality of this work lies in its *structural reframing*: primes are interpreted as canonical carries in PE-space, providing a constructive law and a positive definition of primality.

A companion manuscript develops a heuristic, predictive detector in PE-space that anticipates carry events; the present paper focuses on the constructive law itself.

2 Preliminaries

We begin by establishing the framework of Prime Exponent Space (PE-space) and the successor carry law that underlies the Prime Odometer.

2.1 Prime Exponent Space

Let $p_1 = 2 < p_2 = 3 < p_3 = 5 < \cdots$ denote the ordered primes. Every integer $n \geq 2$ has a unique prime factorization

$$n = \prod_{i \geq 1} p_i^{e_i(n)},$$

where all but finitely many exponents $e_i(n)$ are zero. The *prime exponent vector* of n is

$$\sigma(n) = (e_1(n), e_2(n), e_3(n), \dots),$$

a sparse coordinate in the infinite-dimensional lattice $\mathbb{N}_0^{(\infty)}$. By the Fundamental Theorem of Arithmetic, the map $n \mapsto \sigma(n)$ is bijective.

In this representation, multiplication and division become vector addition and subtraction, respectively:

$$\sigma(ab) = \sigma(a) + \sigma(b), \qquad \sigma\left(\tfrac{a}{b}\right) = \sigma(a) - \sigma(b).$$

2.2 Smooth Frontiers

For $k \geq 1$, define the *p_k-smooth semigroup*

$$S_k = \Big\{ \prod_{i=1}^{k} p_i^{e_i} : e_i \in \mathbb{N}_0 \Big\},$$

the set of integers whose prime factors are bounded by p_k. The integers S_k form a frontier: they are all representable using only the first k axes of PE-space.

A number $n \in S_k$ is called a *pre-prime at level k* if its successor $n+1 \notin S_k$. The emergence threshold theorem states:

$$p_{k+1} = \min(\mathbb{N}_{\geq 2} \setminus S_k), \qquad p_{k+1} - 1 \in S_k.$$

Thus the step $p_{k+1} - 1 \mapsto p_{k+1}$ is the minimal exit from S_k: the canonical carry that forces creation of the new axis p_{k+1}.

2.3 p-adic Carry Depth

The odometer rule is governed by p-adic valuation. For a prime p, define

$$d_p(n) = \max\{t \geq 0 : n \equiv -1 \pmod{p^t}\}.$$

This is the *carry depth* of p at n. Then the valuation of p in $n+1$ is exactly

$$\nu_p(n+1) = d_p(n).$$

In words: when incrementing n by 1, the exponent of p in $n+1$ is given by the depth of the carry chain produced in the p-adic expansion of n. This generalizes the familiar notion of trailing digits in base-B: in base 10, the number of trailing 9s in n is the depth of the carry chain when incremented to $n+1$.

2.4 Canonical Carry

From these definitions we can state the canonical carry law: along the successor map $n \mapsto n+1$, if redistribution across the first k prime axes is impossible, then a new axis must appear. Formally:

$$n \in S_k, \quad n+1 \notin S_k \quad \Rightarrow \quad \sigma(n+1) = (0,0,\ldots,0,1).$$

That is, $n+1$ is prime, and its PE-vector is one-hot on the new axis p_{k+1}. This is the structural inevitability on which the Prime Odometer is built.

3 The Prime Odometer Algorithm

Having established the structural framework of PE-space and the canonical carry law, we now define the Prime Odometer as an explicit algorithm. Its state evolves incrementally, ensuring that $\sigma(n)$ and the primality of $n+1$ can be determined without refactoring from scratch.

3.1 State Representation

At integer n, the odometer maintains:

- A list $P = \{p_1, p_2, \ldots, p_k\}$ of all discovered primes, with $p_k^2 \geq n$.
- Residue counters $r_p(n) \equiv n \pmod{p}$ for each $p \in P$.
- (Optional) Higher-power residues $r_{p^t}(n) \equiv n \pmod{p^t}$ for small t, enabling direct computation of valuations $\nu_p(n+1)$.

This state is finite: to certify primality of $m = n+1$, only primes up to $\sqrt{m}$ are needed.

3.2 Successor Update Rule

To advance from n to $m = n+1$:

1. For each $p \in P$, update the residue
$$r_p(m) = (r_p(n)+1) \bmod p.$$
2. If $r_p(m) = 0$ for some $p \in P$, then $p \mid m$. Repeatedly test higher powers p^t (using r_{p^t} if tracked) to determine $\nu_p(m)$.
3. If no $p \in P$ with $p \leq \sqrt{m}$ divides m, then m is prime. Append m to P, initialize its residue $r_m(m) = 0$, and record $\sigma(m) = (0,0,\ldots,0,1)$ as a one-hot vector.

3.3 Interpretation

The odometer "ticks" in synchrony with the successor map:

- Most steps redistribute mass across existing axes: m factors into known primes.
- At primes, redistribution fails: all counters miss, and a new axis is forced.
- The new prime appears as a canonical carry, exactly like 999 → 1000 in a positional odometer.

3.4 Algorithmic Character

The Prime Odometer is thus:

- **Generative:** it constructs $\sigma(n+1)$ from $\sigma(n)$ incrementally.
- **Deterministic:** no probabilistic tests are required.
- **Factor-free:** it never re-factors $m = n+1$; only small modular counters are updated.
- **Sequential:** primes emerge as inevitable events along the counting trajectory.

The contrast with classical sieves is sharp: Eratosthenes marks composites to reveal primes, while the odometer advances successors until a new axis is forced, revealing primes as structural necessities of PE-space.

4 The Odometer Theorem

We now formalize the correctness of the Prime Odometer. The theorem states that the odometer generates the exact prime-exponent vector $\sigma(n)$ for every n, and that primes emerge precisely as canonical carries.

Theorem 4.1 (Prime Odometer Correctness)**.** *Initialize with $\sigma(2) = (1, 0, 0, \dots)$ and residue counters $r_p(2) \equiv 2 \pmod{p}$ for $p = 2$. At each successor step $n \mapsto n+1$, update residues as described in Section 3. Then:*

1. *For all $n \geq 2$, the odometer state correctly represents $\sigma(n)$.*
2. *If $m = n+1$ is composite, then $\sigma(m)$ is determined exactly by the residue wraps of primes $p \leq \sqrt{m}$.*
3. *If $m = n+1$ is prime, then $\sigma(m) = (0, 0, \dots, 0, 1)$: a one-hot vector on a new axis. Thus the odometer identifies primes without external verification.*

Proof sketch. We proceed by induction on n.

Base case. At $n = 2$, the state is initialized with $\sigma(2) = (1, 0, 0, \dots)$ and $P = \{2\}$. The invariant holds.

Inductive step. Assume the odometer state is correct for n.

Let $m = n+1$. Update each residue $r_p(n)$ to $r_p(m) = (r_p(n) + 1) \bmod p$. Two cases arise:

1. **Composite case.** If some $p \in P$ with $p \leq \sqrt{m}$ has $r_p(m) = 0$, then $p \mid m$. Repeated wraps reveal $\nu_p(m)$, and after dividing out all such prime powers, the cofactor is either 1 or has only larger prime divisors. Since P contains all primes $\leq \sqrt{m}$, any remaining cofactor > 1 must be prime $> \sqrt{m}$ and thus m itself. Hence $\sigma(m)$ is computed exactly.

2. **Prime case.** If no $p \in P$ with $p \leq \sqrt{m}$ divides m, then m has no smaller prime factor. Therefore m is prime. Append m to P, initialize its residue, and set $\sigma(m) = (0, 0, \ldots, 0, 1)$: the one-hot vector on the new axis.

In both cases the odometer state is updated consistently and the invariant extends to m. Thus by induction the algorithm correctly generates $\sigma(n)$ for all $n \geq 2$. □

Corollary 4.2 (Canonical Carry Certificate)**.** *For every $k \geq 1$, the successor step*

$$p_{k+1} - 1 \in S_k, \qquad p_{k+1} \notin S_k$$

forces $\sigma(p_{k+1}) = (0, 0, \ldots, 0, 1)$. In other words, primality of p_{k+1} is certified structurally by the canonical carry, without need for external verification.

This theorem establishes the Prime Odometer as both a generative law and a certifying mechanism: primes appear as the inevitable one-hot carries of arithmetic.

5 Comparison with Classical Sieves

The Prime Odometer provides a constructive law for prime emergence. To clarify its significance, we contrast it with classical sieve methods, particularly the Sieve of Eratosthenes.

5.1 The Sieve of Eratosthenes

The sieve, devised in antiquity, remains a conceptual cornerstone. To enumerate primes up to N:

1. Begin with the list $\{2, 3, \ldots, N\}$.

2. Iteratively select the next unused number p; mark all multiples of p as composite.

3. Continue until all numbers are either marked or declared prime.

This method is *eliminative*: primes are defined by exclusion, emerging as survivors after composites are marked. The sieve is efficient, with complexity $O(N \log \log N)$, and underpins modern segmented and wheel-optimized sieves.

5.2 The Prime Odometer

By contrast, the odometer is *constructive*:

- It advances $n \mapsto n+1$ sequentially, updating the residue counters of known primes.
- When redistribution suffices, $n+1$ factors over existing axes and is composite.
- When redistribution fails, the successor vector is forced to $(0, 0, \ldots, 0, 1)$: a new prime axis is created, and $n+1$ is prime.

Whereas Eratosthenes removes composites to reveal primes, the odometer generates successors and primes appear as structural necessities. [1]

5.3 Duality of Odometers

The contrast mirrors positional vs. canonical counting:

- In base-B, carries occur at constant rate: every trailing string of $(B-1)$ digits forces a positional carry (e.g. $999 \mapsto 1000$).
- In PE-space, carries occur exactly at primes: rare, sparse events of rate $\sim 1/\log n$.

Thus Eratosthenes' sieve is the archetypal *eliminative sieve*, while the Prime Odometer is the archetypal *generative sieve.* One defines primes as what remains when composites are marked; the other defines primes as what emerges when carries are forced.

5.4 Resource Profiles

- The sieve of Eratosthenes requires $O(N)$ storage to mark integers up to N.
- The odometer requires only $O(\pi(\sqrt{N}))$ residue counters, growing with the number of primes up to $\sqrt{N}$.

Both methods are theoretically complete, but the odometer frames primes not as anomalies to be detected, but as the natural milestones of arithmetic progression.

5.5 Conceptual Significance

The odometer provides a *positive definition of primality.* Rather than saying "a prime has no divisors," we can now say: *a prime is the inevitable one-hot successor state forced when redistribution in PE-space fails.* This is not only an algorithmic advance but a conceptual shift, placing primes as the constructive carries of arithmetic, dual to the digit carries of positional systems.

[1] Algorithmically, the Odometer realizes an incremental trial-division sieve by known primes up to $\sqrt{n}$, a method well established in classical number theory texts such as Hardy and Wright [1] and Crandall and Pomerance [2]. Its novelty lies in the reframing: representing the process in PE-space as canonical carries yields a constructive law of prime emergence.

6 Implications

The Prime Odometer reframes primality from a negative property to a constructive law. Several implications follow, both mathematical and conceptual.

6.1 Positive Definition of Primes

Classically, primes are defined negatively: an integer $p > 1$ is prime if it has no divisors other than 1 and itself. The odometer provides a positive definition:

> A prime is an integer m such that $\sigma(m) = (0, 0, \ldots, 0, 1)$, the one-hot vector created when redistribution across existing prime axes fails.

In this view, primality is not absence of divisibility but presence of inevitability: the creation of a new axis.

6.2 Inductive Certification

Because the odometer maintains residues for all primes $\leq \sqrt{n}$, it generates $\sigma(n+1)$ correctly by induction. This provides a new style of proof:

- Base case: $\sigma(2) = (1, 0, 0, \ldots)$.
- Inductive step: update residues; either factor $n + 1$ over P, or force a new axis.
- Thus, by induction, $\sigma(n)$ is correct for all n.

Primality is certified not by external test but by structural necessity.

6.3 Generative Sieve

The odometer is not a cataloguer of primes up to N; it is a generator of successors. In this sense it functions as a *generative sieve*:

- It advances incrementally, producing $\sigma(n)$ at each tick.
- Composites arise naturally from redistribution over existing axes.
- Primes arise inevitably as canonical carries.

This contrasts with traditional sieves, which are bulk eliminative methods.

6.4 Duality with Positional Systems

The odometer clarifies a symmetry:

- In positional systems, carries occur constantly (rate $1/(B-1)$), enforcing digit rollover.
- In PE-space, carries occur sparsely (rate $\sim 1/\log n$), enforcing axis creation.

Thus primes are revealed as the *canonical carries* of arithmetic: the sparse but necessary events that extend dimensionality.

6.5 Philosophical Perspective

The odometer shifts the epistemology of primality:

- From *detection* (is n prime?) to *construction* (primes emerge when they must).
- From *anomaly* (exceptions in the sequence) to *inevitability* (structural law).
- From *negative definition* (lack of divisors) to *positive definition* (forced one-hot successor).

This reconceptualization elevates the odometer beyond an algorithm: it is a theoretical lens that positions primes as structural inevitabilities of counting itself.

7 Outlook and Open Questions

The Prime Odometer establishes a generative law of primality. Yet many questions remain open, both mathematical and practical. We highlight several directions for further study.

7.1 Prime Gaps and Next-Prime Distance

The odometer identifies primes at the exact moment redistribution fails, but it does not predict *how far ahead* the next prime lies. The length of prime gaps remains encoded in the interplay of residue counters, but an explicit formula has not yet been extracted. A natural open problem is:

> Can the odometer state at n be used to bound or predict the distance to the next prime?

If so, this would provide a constructive mechanism for understanding prime gaps directly in PE-space.

7.2 Truncated Odometers and Engineering Approximations

In theory, the odometer requires residues for all primes $\leq \sqrt{n}$. In practice, this is memory-intensive at scale. Classical engineering techniques such as wheels and smallest-prime-factor tables approximate the full odometer state with bounded resources. A research direction is to analyze:

> When and how can truncated odometers preserve correctness guarantees while reducing cost?

This bridges the pure theory with practical implementation.

7.3 Analytic Connections

The odometer frames prime emergence as carries of rate $\sim 1/\log n$. This is in harmony with the Prime Number Theorem, which states that

$$\pi(x) \sim \frac{x}{\log x}.$$

A fruitful direction is to relate odometer dynamics explicitly to analytic objects such as Chebyshev's function $\psi(x)$ and to study whether the odometer yields new insights into error terms in prime counting.

7.4 Complexity and Performance

The asymptotic complexity of the odometer, compared with sieves, remains to be explored in detail. While the state size grows as $\pi(\sqrt{n})$, incremental updates may amortize efficiently. Precise complexity analyses and optimizations are natural future work.

7.5 Broader Emergence Principle

The odometer provides a clean arithmetic instantiation of a broader principle:

> Irreducible structures emerge when redistribution fails.

This principle has already been connected to physics, complexity science, and information theory in prior work. The odometer now grounds this axiom in a generative arithmetic law, suggesting fertile cross-disciplinary analogies remain to be drawn.

7.6 Summary of Open Questions

- Can prime gaps be bounded or predicted from odometer state?
- How can truncated odometers be engineered efficiently without losing rigor?
- What new connections to analytic number theory emerge from the carry law?
- What is the precise computational complexity of the odom

8 Conclusion

We have introduced the *Prime Odometer*, a constructive successor law in Prime Exponent Space that generates primes as the inevitable one-hot carries of arithmetic. Rather than defining primes negatively—as numbers lacking divisors—the odometer defines them positively: they are the states that *must* arise when redistribution across existing axes fails.

The Odometer Theorem establishes correctness by induction: starting from $\sigma(2)$, each tick $n \mapsto n+1$ updates residue counters, factors composites over known primes, and forces a new axis precisely when $n+1$ is prime. Primality is thus certified structurally, without external tests.

This reframes primality in three key ways:

- **Constructive definition:** a prime is the one-hot successor vector forced by failure of redistribution.
- **Generative sieve:** the odometer produces primes sequentially, rather than leaving them as remainders after composites are eliminated.
- **Canonical carry:** primes are the sparse but necessary carries of PE-space, dual to the frequent digit carries of positional systems.

Conceptually, the odometer complements and completes the sieve of Eratosthenes: together they form a duality of prime generation, one eliminative and one constructive. Practically, the odometer suggests new pathways for incremental prime generation and analysis of prime gaps. Theoretically, it elevates primes from anomalies to inevitabilities: the canonical carries that arithmetic cannot avoid.

In this sense, the Prime Odometer is more than an algorithm. It is a law of emergence, revealing that primes are not random accidents scattered along the number line, but structural necessities of counting itself.

External Reviews and Strategic Positioning

Independent reviewers emphasized that the central novelty of the Prime Odometer lies not in its raw mechanics—which are algorithmically equivalent to classical incremental sieves—but in its conceptual and pedagogical reframing. The Odometer reframes prime generation as a *positive*, *constructive*, and *inevitable* process in Prime Exponent Space (PE-space). Primes appear as *canonical carries*, rather than as survivors of elimination.

Key points from the reviews:

- The algorithm is mechanically equivalent to trial-division with residue counters, but its **interpretive reframing** is original, powerful, and pedagogically valuable.
- The duality between **eliminative sieves** (Eratosthenes) and the **generative sieve** (Odometer) is central, and reviewers found this conceptual contrast compelling.
- The Odometer provides a **positive definition of primality**: a prime is the successor state forced when redistribution in PE-space fails.
- While not an algorithmic breakthrough, the Odometer offers a new **mental model** of why primes emerge, likely to influence teaching, exposition, and interdisciplinary analogy.

Reviewers also stressed the strategic sequencing of publications. Publishing *The Prime Odometer* first grounds the PE-space framework in a provably correct, non-controversial setting. Only after this foundation is established should the more heuristic work on rupture dynamics and the Φ-detector be introduced. This order mitigates skepticism and frames the rupture model as a natural extension rather than a speculative leap.

Relation to prior art. The underlying mechanism of the Odometer is algorithmically equivalent to classical incremental sieves and trial-division methods, which are well documented in standard references such as Hardy and Wright [1], Cohen [3], and Crandall and Pomerance [2]. Its contribution lies not in asymptotic improvement, but in its *reframing*: the Odometer renders primes as canonical carries in PE-space, providing a constructive law and a positive definition of primality. This situates the work alongside, but conceptually distinct from, eliminative sieves and their modern refinements.

Appendix A: Pseudocode (Prime Odometer)

A.1 Minimal Odometer (Decision-Only, Factor-Free)

State

- P : list of discovered primes, initially [2].
- r : dictionary of residues $r[p] \equiv n \pmod{p}$ for each $p \in P$.
- n : current integer (start at $n = 2$).

Tick Procedure (advance to $m = n + 1$)

```
function TICK(n, P, r):
    m = n + 1

    # 1) Increment residues
    for p in P:
        r[p] = (r[p] + 1) % p

    # 2) Test divisibility up to sqrt(m) via residues
    composite = False
    for p in P:
        if p*p > m: break
        if r[p] == 0:
            composite = True
            break

    if composite:
        # m is composite; factors may be recovered separately if desired.
        return (m, "composite", P, r)

    # 3) Otherwise m is prime: append to prime list and initialize its residue
    P.append(m)
    r[m] = 0  # since m  0 (mod m)
    return (m, "prime", P, r)
```

Driver (enumerate up to a bound N)

```
P = [2]
r = {2: 2 % 2}  # i.e., 0
n = 2
yield 2

while n < N:
    (m, tag, P, r) = TICK(n, P, r)
    if tag == "prime":
        yield m
    n = m
```

Notes. This minimal odometer:

- never re-factors $m = n + 1$;
- decides primality using only residues for $p \leq \sqrt{m}$;
- enumerates primes in increasing order.

A.2 Extended Odometer (Optional Valuations via p^t Stacks)

Additional State

- R : dictionary-of-dictionaries $R[p][t] \equiv n \pmod{p^t}$ for $t = 1, \ldots, T_p$ (small).

Tick Procedure with Valuations

```
function TICK_WITH_VALUATIONS(n, P, r, R, Tpow):
    m = n + 1

    # 1) Increment residues mod p and mod p^t (for small t)
    for p in P:
        r[p] = (r[p] + 1) % p
        for t in 2..Tpow[p]:
            mod_pt = p**t
            R[p][t] = (R[p][t] + 1) % mod_pt

    # 2) Compute _p(m) from carry-depth wraps (optional)
    nu = {}            # valuations _p(m)
    composite = False
    for p in P:
        if p*p > m: break
        if r[p] == 0:
            composite = True
            # detect depth by scanning highest t where residue wrapped to 0
            tmax = 1
            for t in 2..Tpow[p]:
                if R[p][t] == 0:
                    tmax = t
                else:
                    break
            nu[p] = tmax

    if composite:
        # cofactor can be inferred as m / prod p^{_p} if desired
        return (m, "composite", nu, P, r, R)

    # 3) Otherwise m is prime: append to primes and initialize residues
    P.append(m)
    r[m] = 0
    R[m] = {}
    for t in 2..Tpow_default:
        R[m][t] = 0
    return (m, "prime", {m:1}, P, r, R)
```

Remarks.

- The minimal variant suffices for *primality decision and enumeration.*
- The extended variant recovers $\nu_p(m)$ by reading carry-depth wraps in $R[p][t]$, still without refactoring m.
- If a full $\sigma(m)$ vector is required, the cofactor $R\_m = m/\prod p^{\nu_p(m)}$ can be computed arithmetically; when the decision branch declares m prime, this cofactor equals m and the one-hot vector is certified.

References

[1] G.H. Hardy and E.M. Wright. *An Introduction to the Theory of Numbers*. Oxford University Press, 6th ed., 2008.

[2] R. Crandall and C. Pomerance. *Prime Numbers: A Computational Perspective*. Springer, 2nd ed., 2005.

[3] H. Cohen. *Number Theory, Volume I: Tools and Diophantine Equations*. Springer, 2nd ed., 2007.