MATHDISCOVERIES
Mathematical Preprint

Working Draft

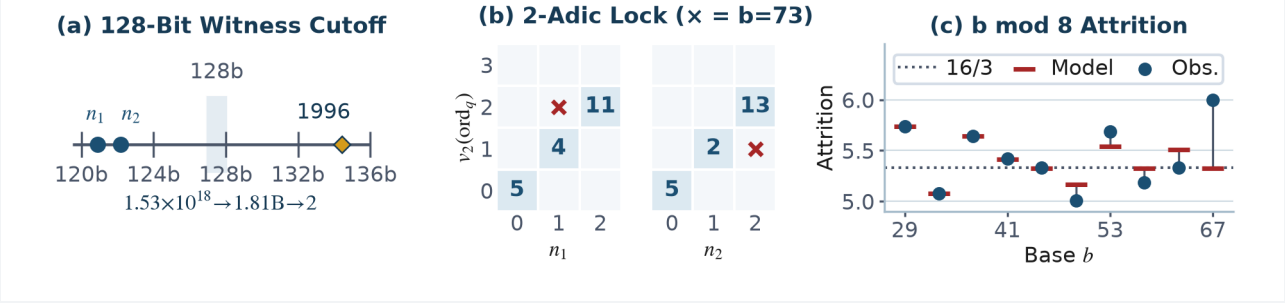
Two 37-Digit 20-Base Strong Pseudoprimes: Kernel-Verified Certificates, Order Criteria, and Exact Sieve Laws

Tyler B. Hughes*,†

†Independent Researcher, United States

Supporting Information: Reproducible exact checker (`recompute.py`), audited Lean 4 library (`lean/SpspPsi20/`), and trust-boundary ledgers

Disclosure: AI-assisted working draft directed by Tyler B. Hughes. Not for submission as is. No human mathematician has yet refereed the proof.



ABSTRACT: The first twelve prime bases suffice for every unsigned 64-bit integer, whereas the first twenty fail inside 128 bits. We exhibit two 37-digit counterexamples $n_1 = p_1(2p_1 - 1)$ (121 bits) and $n_2 = p_2(2p_2 - 1)$ (123 bits) in Zhang's K_2 family, each passing the Miller–Rabin strong test through base 71 and failing at 73 while passing Fermat at 73, giving the unconditional bound $\psi_{20} \le n_1 < 2^{121}$. All four Pratt primality proofs, all twenty-one strong tests, the K_2 order and Fermat criteria, the exact strong-liar share $(1 + 2 \cdot 4^{-e})/6$ and base-2 cap 3/16, the sieve residue count $A(b)$, the exact b mod 8 Jacobsthal character-sum cell split, and the general A/B semiprime quadratic-residue law are proved inside the Lean 4 pure kernel. An external GMP sieve extending Zhang's K_2 search from 10^{36} to 10^{37} across 1,024 shards (64 rechecked) returns exactly n_1 and n_2, while Zhang's unrestricted conjecture $\psi_t = \psi'_t$ $(t \ge 12)$ remains open.

Verification Scope (G11): WARN: Human referee: pending
Main results: 6 of 8 in Lean Kernel (75%) · 2 Cert Only · 1 open (not counted)
Abstract claims: 6 of 7 in Lean Kernel (86%) · 1 Cert Only · 1 open (not counted)

Full Trust Boundary: Section 5.

MSC 2020: 11Y11, 11A51, 11Y16 **Keywords:** strong pseudoprimes, Miller–Rabin primality test, Pratt certificates, Jacobsthal character sums, Lean 4 formalization

1. INTRODUCTION AND MAIN RESULTS

When a composite integer passes a deterministic strong probable-prime test to a fixed set of prime bases, the least such composite marks the exact boundary beyond which the test ceases to be decisive. Setting aside integers below 2 and prime divisors of the chosen bases (checked directly), Sorenson and Webster proved that the threshold ψ_{12} for the first twelve prime bases exceeds 2^{64} [5]. A natural computational question follows immediately: do eight additional prime bases suffice for signed or unsigned 128-bit arithmetic? Two explicit semiprime factorizations settle the question in the negative.

Theorem 1 (Two 37-Digit 20-Base Strong Pseudoprimes). *For each prime p listed below, $q = 2p - 1$ is also prime, the product $n = pq \in (10^{36}, 10^{37}]$ is a strong pseudoprime to every one of the first twenty prime bases $2, 3, \dots, 71$, fails the strong test at the 21st prime 73, and still passes the Fermat test $73^{n-1} \equiv 1 \pmod n$:*

$$p_1 = 1113813905784150541,$$
$$n_1 = 2481162833436289154832260656117034821$$
$$(2^{120} < n_1 < 2^{121}),$$
$$p_2 = 1726253494836549181,$$
$$n_2 = 5959902256870799858089729225626992341$$
$$(2^{122} < n_2 < 2^{123}).$$

Consequently, $\psi_{20} \le n_1 < 2^{121}$ and $\psi_k \le n_1$ for every $k \le 20$.

For both witnesses, $n-1 = 4d$ with d odd, so the strong test to base b accepts precisely when $b^d \equiv 1$, $b^d \equiv -1$, or $b^{2d} \equiv -1 \pmod n$. The standalone Python check below verifies both compositeness factorizations and all twenty-one base tests

verbatim (executed by `recompute.py`).

Instant Witness Check (10 lines, verified verbatim by recompute.py)

```
bases = (2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71)
for p in (1113813905784150541, 1726253494836549181):
    n = p * (2*p - 1)
    d, rem = divmod(n - 1, 4)
    assert rem == 0 and d % 2 == 1 and 1 < p < n < 2**127
    def passes(b):
        x = pow(b, d, n)
        return x in (1, n-1) or x*x % n == n-1
    assert all(passes(b) for b in bases)
    assert not passes(73) and pow(73, n-1, n) == 1
```

All six proved mathematical theorems in this manuscript (Theorems 1, 2, 5 and Propositions 3, 4, 6) are formalized inside the pure Lean 4 kernel (`lean/SpspPsi20/`, Section 5). Only the two large-scale external C sieve enumerations (Propositions 7 and 8) sit outside Lean, and Zhang's unrestricted conjecture (Conjecture 9) is explicitly open. Figure 1 summarizes the sieve counts and bit-length positions of n_1 and n_2 relative to the 128-bit cutoffs and Bleichenbacher's 1996 witness [2].

2. ORDER CRITERIA AND STRONG-LIAR DENSITY IN K_2

Write an odd integer $n \geq 3$ as $n-1 = 2^s d$ with d odd, and suppose $\gcd(b,n) = 1$. The Miller–Rabin strong test accepts n to base b when $b^d \equiv 1 \pmod n$ or $b^{2^i d} \equiv -1 \pmod n$ for some $0 \leq i < s$. For any odd prime r with $\gcd(b,r) = 1$, write $t_r(b) = v_2(\mathrm{ord}_r\, b)$ for the 2-adic valuation of the multiplicative order of b mod r, and write $j_r(b) = v_2(r-1) - t_r(b)$ for the maximal $j \leq v_2(r-1)$ such that $b^{(r-1)/2^j} \equiv 1 \pmod r$. Jaeschke established the 2-adic signature criterion for squarefree integers and the Fermat criterion below for K_2 [3].

Theorem 2 (Fermat and Strong-Test Order Criteria for K_2). *Let p and $q = 2p-1$ be odd primes, set $n = pq$, and suppose $\gcd(b,n) = 1$. Then:*

1. $b^{n-1} \equiv 1 \pmod n$ *if and only if* $\left(\frac{b}{q}\right) = 1$;
2. *n is a strong probable prime to base b if and only if $v_2(\mathrm{ord}_p\, b) = v_2(\mathrm{ord}_q\, b)$, equivalently $j_q(b) = j_p(b) + 1$.*

Proof. Write $p-1 = 2^e u$ with u odd. Then $q-1 = 2^{e+1}u$ and $n-1 = (2p+1)(p-1) = 2^e(2p+1)u$. Modulo p, Fermat's congruence holds for every unit b. Modulo q, Euler's criterion gives $b^{p-1} = b^{(q-1)/2} \equiv \left(\frac{b}{q}\right) \pmod q$, and raising to the odd power $2p+1$ preserves this sign, establishing (1) by the Chinese remainder theorem.

For (2), the odd part of $n-1$ is $d = (2p+1)u$, which strips the odd parts of both $\mathrm{ord}_p\, b$ and $\mathrm{ord}_q\, b$. Thus $b^d \bmod p$ has order $2^{t_p(b)}$ and $b^d \bmod q$ has order $2^{t_q(b)}$. If $t_p(b) = t_q(b) = 0$, then $b^d \equiv 1 \pmod{pq}$. If $t_p(b) = t_q(b) = t \geq 1$, then $t \leq e$ (since $2^t \mid p-1 = 2^e u$), so squaring $t-1 < e$ times reaches -1 simultaneously modulo p and modulo q. Conversely, if $t_p(b) \neq t_q(b)$, one residue reaches 1 strictly before the other, so they can never equal -1 at the same squaring step. Finally, $j_q(b) - j_p(b) = (e+1-t_q(b)) - (e-t_p(b)) = 1 + t_p(b) - t_q(b)$. □

Figure 2 plots $(t_p(b), t_q(b))$ across all twenty-one prime bases for both n_1 and n_2. Every base through 71 lands on the diagonal $t_p = t_q$; at base 73, the valuations separate to $(1,2)$ for n_1 and $(2,1)$ for n_2 while still satisfying $t_q(73) < 3 = v_2(q-1)$, explaining why both witnesses pass the Fermat test at 73 yet fail its strong test.

Counting unit classes in $(\mathbb{Z}/p\mathbb{Z})^\times \times (\mathbb{Z}/q\mathbb{Z})^\times$ by their 2-primary order yields the exact strong-liar fraction and base-2 congruence restrictions [3, 6].

Proposition 3 (Strong-Liar Density and Base-2 Cap). *Let p and $q = 2p-1$ be odd primes, $n = pq$, and $e = v_2(p-1)$. Let $L(n)$ be the number of unit classes $b \in (\mathbb{Z}/n\mathbb{Z})^\times$ for which n is a strong probable prime to base b. Then:*

1. $L(n)/\varphi(n) = (1 + 2\cdot 4^{-e})/6$;
2. *if $p \equiv 3 \pmod 4$ (so $e = 1$), then $L(n)/\varphi(n) = 1/4$ and n fails the strong test to base 2;*
3. *if n passes the strong test to base 2, then $p \equiv 1 \pmod 4$, $q \equiv 1 \pmod 8$, $n \equiv 1 \pmod 4$, and $L(n)/\varphi(n) \leq 3/16$, with equality at $e = 2$.*

Proof. Write $p-1 = 2^e u$ with u odd, so $q-1 = 2^{e+1}u$. In any cyclic group of order $2^E u$, exactly u elements have $v_2(\mathrm{ord}) = 0$ and $2^{k-1}u$ elements have $v_2(\mathrm{ord}) = k$ for each $1 \leq k \leq E$. By the Chinese remainder theorem and Theorem 2,

$$L(n) = u^2 + \sum_{k=1}^{e} (2^{k-1}u)^2 = u^2\left(1 + \frac{4^e - 1}{3}\right) = \frac{4^e + 2}{3}u^2.$$

Dividing by $\varphi(n) = (p-1)(q-1) = 2^{2e+1}u^2$ gives $(4^e+2)/(6\cdot 4^e) = (1+2\cdot 4^{-e})/6$. When $p \equiv 3 \pmod 4$, we have $q = 2p-1 \equiv 5 \pmod 8$, so $\left(\frac{2}{q}\right) = -1$ and base 2 fails even the Fermat test by Theorem 2(1). Hence passing base 2 forces $p \equiv 1 \pmod 4$, whence $q \equiv 1 \pmod 8$, $n = pq \equiv 1 \pmod 4$, $e \geq 2$, and $(1 + 2\cdot 4^{-e})/6 \leq 3/16$ (with equality at $e = 2$, achieved by both n_1 and n_2). □

3. PRATT CERTIFICATES AND WHEEL RESIDUE COUNTS

Although n_1 and n_2 are visibly composite, proving membership in K_2 requires certifying primality of p_1, q_1, p_2, q_2. A Pratt certificate [4] proves primality of r by factoring $r-1$ completely and exhibiting a primitive root a satisfying $a^{r-1} \equiv 1 \pmod r$ and $a^{(r-1)/\ell} \not\equiv 1 \pmod r$ for every prime $\ell \mid r-1$, recursing on each prime factor ℓ. Because $q - 1 = 2(p-1)$, each pair (p_i, q_i) shares its entire odd factorization tree (twelve non-leaf nodes across both witnesses, all verified inside `lean/SpspPsi20/Witnesses.lean` via Mathlib's `lucas_primality` and rechecked by `recompute.py`).

Table 1. Top-level Pratt certificate nodes for the four prime factors of n_1 and n_2, where $r_1 = 18563565096402509$ and $r_2 = 158955202102813$ are recursively certified in twelve total non-leaf nodes.

Node	Witness a	Factorization of node minus 1
p_1	2	$2^2 \cdot 3 \cdot 5 \cdot r_1$
q_1	101	$2^3 \cdot 3 \cdot 5 \cdot r_1$
p_2	10	$2^2 \cdot 3 \cdot 5 \cdot 181 \cdot r_2$
q_2	89	$2^3 \cdot 3 \cdot 5 \cdot 181 \cdot r_2$

To search $10^{36} < p(2p-1) \leq 10^{37}$ efficiently, note that Proposition 3(3) forces $q \equiv 1 \pmod 4$. For every odd prime base $b \leq 71$, quadratic reciprocity therefore gives $\left(\frac{b}{q}\right) =$

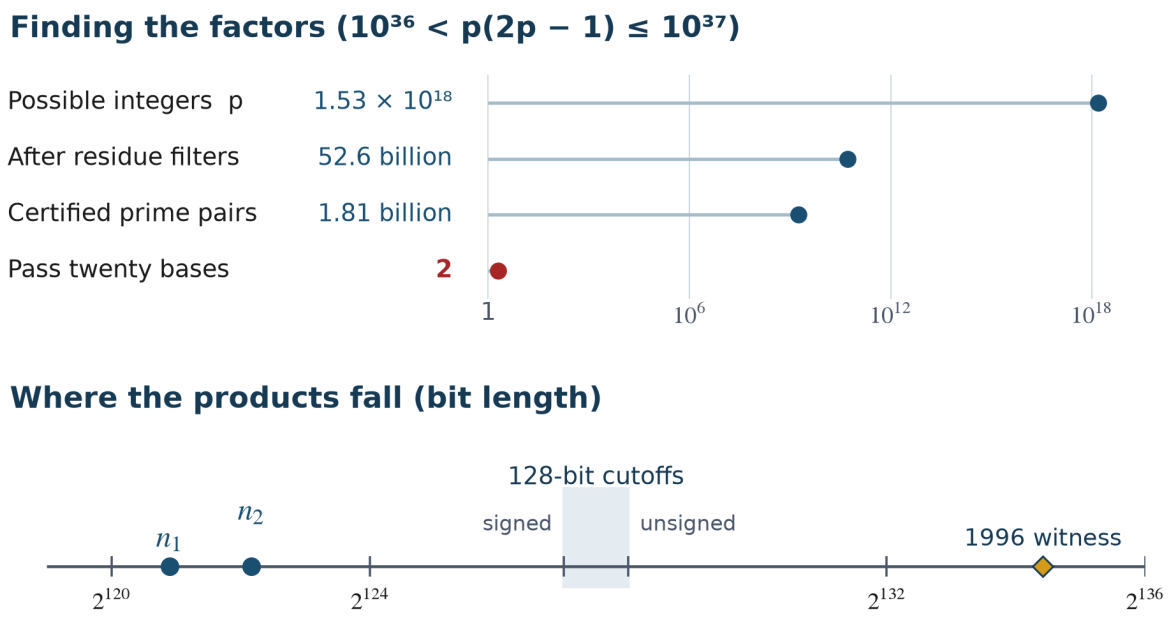


Figure 1. Search attrition and witness sizes in $K_2 = \{p(2p-1) : p, 2p-1 \text{ prime}\}$. Top: horizontal log-scale counts across the interval $10^{36} < p(2p-1) \le 10^{37}$, from 1,528,961,196,313,242,172 candidate integers p through wheel residue filters (52,613,846,073), certified prime pairs (1,809,514,692), and the exact 2 twenty-base survivors. Bottom: bit-length ruler showing n_1 (121 bits) and n_2 (123 bits) below both signed (2^{127}) and unsigned (2^{128}) 128-bit cutoffs, alongside Bleichenbacher's 41-digit (135-bit) 1996 witness [2]. Each witness carries an unconditional Lean 4 Pratt and strong-test proof; only the total count 2 in the band relies on the external enumeration.

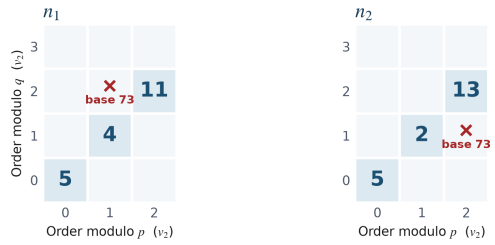


Figure 2. Exact counts of prime bases $b \le 71$ in each $(v_2(\mathrm{ord}_p\, b), v_2(\mathrm{ord}_q\, b))$ cell (all 20 lie on the diagonal), together with the crimson cross locating base 73 at $(1,2)$ for n_1 and $(2,1)$ for n_2. Both crosses lie below $v_2(q-1) = 3$, so 73 remains a quadratic residue modulo q and passes the Fermat test on both composites.

$\left(\frac{2p-1}{b}\right)$. Any residue class $x = p \bmod b$ with $\left(\frac{2x-1}{b}\right) \neq 1$ or $x \equiv 0 \pmod b$ (since $p > 71$) can be discarded before any primality test.

Proposition 4 (Admissible Residue Count and First-Wheel Products). *For every odd prime b, the number of residue classes $x \in \mathbb{F}_b$ with $x \neq 0$ and $\left(\frac{2x-1}{b}\right) = 1$ is*

$$A(b) = \frac{b - 2 - \left(\frac{-1}{b}\right)}{2} = \begin{cases} (b-1)/2 & \text{if } b \equiv 3 \pmod 4, \\ (b-3)/2 & \text{if } b \equiv 1 \pmod 4. \end{cases}$$

Over 4 and the ten odd primes $3 \le b \le 31$, the first-wheel modulus and local-count products equal

$$4 \prod_{3 \le b \le 31} b = 401{,}120{,}980{,}260, \qquad \prod_{3 \le b \le 31} A(b) = 10{,}135{,}125.$$

Proof. Since b is odd, $x \mapsto 2x - 1$ permutes $\mathbb{F}_b$, so exactly $(b-1)/2$ classes satisfy $\left(\frac{2x-1}{b}\right) = 1$. The excluded class $x = 0$ maps to -1, which is a quadratic residue precisely when $\left(\frac{-1}{b}\right) = 1$ (i.e., $b \equiv 1 \pmod 4$), leaving $(b-1)/2 - (1 + \left(\frac{-1}{b}\right))/2 = (b - 2 - \left(\frac{-1}{b}\right))/2$. Direct multiplication gives $4 \prod_{3 \le b \le 31} b = 401{,}120{,}980{,}260$ and $\prod_{3 \le b \le 31} A(b) = 1 \cdot 1 \cdot 3 \cdot 5 \cdot 5 \cdot 7 \cdot 9 \cdot 11 \cdot 13 \cdot 15 = 10{,}135{,}125$. Because 4 and the ten odd primes $3, 5, \ldots, 31$ are pairwise coprime, the Chinese remainder theorem (written argument) identifies these products with the 10,135,125 simultaneous admissible residue classes modulo 401,120,980,260 satisfying $p \equiv 1 \pmod 4$ and all ten local predicates. □

4. CHARACTER-SUM SIEVE LAWS AND ENUMERATIONS

Although the residue wheel constrains only $\left(\frac{2x-1}{b}\right) = +1$, the survival probability of a prime pair at an odd prime base b also depends on $\left(\frac{x}{b}\right) = \left(\frac{p}{b}\right)$ (by quadratic reciprocity, since $p \equiv 1 \pmod 4$). Write χ for the quadratic character of $\mathbb{F}_b$ ($\chi(0) = 0$), and let N_{-+} and N_{++} count $x \in \mathbb{F}_b$ with $(\chi(x), \chi(2x-1)) = (-1, +1)$ and $(+1, +1)$, respectively.

Theorem 5 (Jacobsthal Correlation Sum and Exact $b \bmod 8$ Split). *For every odd prime b,*

$$\sum_{x \in \mathbb{F}_b} \chi(x)\chi(2x-1) = -\left(\frac{2}{b}\right),$$

and the two wheel-admissible residue cells satisfy exact integer

identities:

$$4N_{-+} = b - 2 - \left(\frac{-1}{b}\right) + 2\left(\frac{2}{b}\right),$$
$$4N_{++} = b - 2 - \left(\frac{-1}{b}\right) - 2\left(\frac{2}{b}\right).$$

Proof. Observe that $8x(2x-1) = (4x-1)^2 - 1$. Setting $u = 4x-1$ permutes $\mathbb{F}_b$ and gives $\chi(8)\sum_x \chi(x)\chi(2x-1) = \sum_{u\in\mathbb{F}_b}\chi(u^2-1)$. The affine conic $v^2 = u^2-1$ has $b-1$ points over $\mathbb{F}_b$, parameterized by $u+v = t \in \mathbb{F}_b^\times$, $u-v = t^{-1}$. On the other hand, counting points over u gives $\sum_u(1+\chi(u^2-1)) = b + \sum_u \chi(u^2-1)$, so $\sum_u \chi(u^2-1) = -1$. Since $\chi(8) = \chi(2)$, this proves $\sum_x \chi(x)\chi(2x-1) = -\chi(2)$. Moreover, $\sum_{x\neq 0,2^{-1}}\chi(x) = -\chi(2^{-1}) = -\chi(2)$. Adding and subtracting $\sum_{x\neq 0,2^{-1}}\chi(x)(1+\chi(2x-1)) = 2(N_{++}-N_{-+}) = -2\chi(2)$ against $N_{-+}+N_{++} = A(b)$ (Proposition 4) yields both identities. □

Heuristic attrition model. For a prime pair in cohort $e = v_2(p-1) \geq 2$, modeling admissible residues as uniform over the $b-2$ classes $x \notin \{0, 2^{-1}\}$ and higher 2-power residue choices as independent yields the per-base pass fraction

$$S_e(b) = \frac{N_{-+}/2 + h_e N_{++}}{b-2}, \qquad h_e = \frac{1+8\cdot 4^{-e}}{6}.$$

In particular, at $e = 2$ (both new witnesses) $h_2 = 1/4$, giving $S_2(3) = 1/4$ and $S_2(5) = 1/12$, while the sign $\left(\frac{2}{b}\right)$ creates systematic variation modulo 8. At base 2, the quadratic supplement yields $S_2(2) = 1/2$ and $S_e(2) = h_e$ for $e \geq 3$. Weighting cohorts by 2^{-e} predicts survival fraction $\sum_{e\geq 2} 2^{-e}\prod_{i\leq t} S_e(b_i)$, whose successive ratios track Zhang's observed fixed-cutoff (10^{36}) attrition ratios closely (Figure 3). Multiplying by the Bateman–Horn prime-pair integral gives an expected 1.614 twenty-base survivors in $(10^{36}, 10^{37}]$ (versus 2 found). Crucially, this model is purely explanatory and is never used by the sieve to discard candidates.

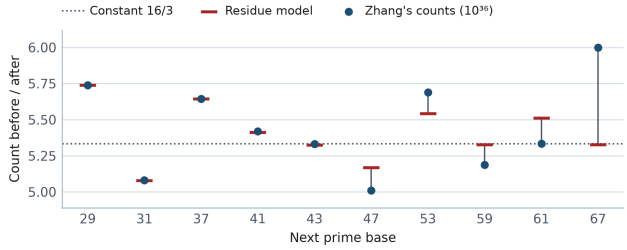


Figure 3. Measured attrition ratios at Zhang's single cutoff 10^{36} (counts before divided by counts after each prime base $b \in [29, 67]$ [6]) compared with the exact-b mod 8 Jacobsthal residue model and the naive constant baseline 16/3. The final ratio is $24/4 = 6.0$.

To probe other linear semiprime families $p = Bm+1$, $q = Am+1$ with coprime $A, B > 0$, we use the following general 2-adic quadratic-residue obstruction.

Proposition 6 (General Ratio A/B Necessary Condition). *Let p, q be odd primes and a, b positive integers with $\gcd(a,b) = 1$ and $b(q-1) = a(p-1)$. Set $n = pq$, and let c satisfy $\gcd(c,n) = 1$. If n is a strong probable prime to base c, then:*

1. *if $v_2(a) = v_2(b)$, then $\left(\frac{c}{p}\right) = \left(\frac{c}{q}\right)$;*
2. *if $v_2(b) < v_2(a)$, then $\left(\frac{c}{q}\right) = 1$;*
3. *if $v_2(a) < v_2(b)$, then $\left(\frac{c}{p}\right) = 1$.*

Proof. By Jaeschke's signature criterion, passing the strong test forces $t_p(c) = t_q(c)$. On the other hand, for any odd prime r and unit c, Euler's criterion gives $\left(\frac{c}{r}\right) = -1 \iff t_r(c) = v_2(r-1)$, and $\left(\frac{c}{r}\right) = 1 \iff t_r(c) < v_2(r-1)$. Since $b(q-1) = a(p-1)$, we have $v_2(q-1) - v_2(p-1) = v_2(a) - v_2(b)$. If $v_2(a) = v_2(b)$, then $v_2(p-1) = v_2(q-1)$ so $t_p(c) = t_q(c)$ hits the top valuation at p if and only if it hits it at q. If $v_2(b) < v_2(a)$, then $t_q(c) = t_p(c) \leq v_2(p-1) < v_2(q-1)$, forcing $\left(\frac{c}{q}\right) = 1$; symmetrically $v_2(a) < v_2(b)$ forces $\left(\frac{c}{p}\right) = 1$. □

External computational enumerations. Our main search extended Zhang's K_2 enumeration [6] across the next decimal decade using a two-stage wheel (first wheel modulo 401,120,980,260 from Proposition 4, second wheel for $37, 41, 43$, and direct Legendre lookups through 71), testing surviving factors $p, 2p-1 < 2^{62}$ deterministically via the proven twelve-base threshold [5] and applying GMP strong tests to the 1,809,514,692 surviving prime pairs.

Proposition 7 (External K_2 Enumeration on $(10^{36}, 10^{37}]$ — Cert Only). *Across all 1,024 shards covering $10^{36} < p(2p-1) \leq 10^{37}$, the primary GMP C sieve scanned 1,528,961,196,313,242,172 integers p, filtered 52,613,846,073 wheel-admissible candidates, certified 1,809,514,692 prime pairs, and returned exactly n_1 and n_2 passing the first 20 prime bases (and 0 passing all 21 prime bases through 73). An independent second implementation using a different traversal and deterministic primality test reproduced every shared count and hit across 64 shards (62 pre-selected summary/boundary shards plus both hit shards); a planted wrong-factor control recovered 0/24 of Zhang's benchmark rows versus 24/24 for the verified code. Subject to Zhang's enumeration through 10^{36} and this external search, $\psi'_{20} = n_1$ and $\psi'_{21} > 10^{37}$.*

Proposition 8 (Companion Seven-Ratio Search — Cert Only). *A separate single-implementation sieve built from Proposition 6 tested $A/B \in \{3, 4, 5, 6, 3/2, 4/3, 5/2\}$ with $p > 173$ and $pq \leq \psi'_{17}$, finding no strong pseudoprime below ψ'_t for $14 \leq t \leq 17$.*

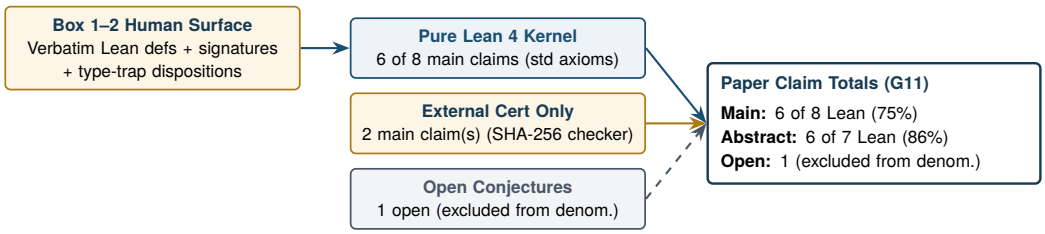
Conjecture 9 (Zhang's Minimality Conjecture [6]). *For every integer $t \geq 12$, the unrestricted threshold ψ_t equals the K_2-restricted threshold ψ'_t.*

5. FORMAL VERIFICATION & READER'S TRUST BOUNDARY

What You Must Trust to Accept This Paper:

1. **Lean 4 Kernel & Toolchain:** leanprover/lean4:v4.35.0; committed axiom receipt anchored to git commit fa70c3374d1a (independent gate-box kernel replay pending); standard axioms only (propext, Classical.choice, Quot.sound).
2. **Core Definitions & Fidelity (Boxes 1–2):** Verify that the verbatim Lean definitions in Box 1 match Section 2 and that each Lean signature in Box 2 matches its stated theorem.
3. **External Certificates & Written Proofs:** 2 SHA-256-pinned external artifact(s) plus any claims marked Written or Open in Table 3 below.

Reader's Trust-Flow Architecture (Distinct Evidence Paths to Claim Totals)



Anti-Shortcut Audit Strip — Source-Level Approx (Gate Replay Pending)

WARN: Committed receipt (gate replay pending) | PASS: sorry/admit: 0 (per committed receipt)
PASS: native_decide: 0 (per committed receipt) | PASS: Custom axioms & meta: 0 (per committed receipt)
PASS: Coverage: 100/100 public theorems (source-level approx) | WARN: Fidelity: 6/6 exact (draft cross-family)
WARN: Witnesses: pending gate replay | WARN: Human referee: pending

Box 1 — Verbatim Lean 4 Custom Definitions (Human Audit Surface)

SpspPsi20.IsStrongProbablePrime (SpspPsi20/Defs.lean:26-29) — Section 2 (Miller-Rabin strong probable prime test)

```
def IsStrongProbablePrime (n a : ℕ) : Prop :=
  Odd n ∧ 3 ≤ n ∧
    ∃ s d : ℕ, Odd d ∧ n - 1 = 2 ^ s * d ∧
      (a ^ d ≡ 1 [MOD n] ∨ ∃ r < s, a ^ (2 ^ r * d) ≡ n - 1 [MOD n])
```

SpspPsi20.IsFirstKPrimesStrongPseudoprime (SpspPsi20/Defs.lean:37-39) — Section 1 (strong pseudoprime to first k primes via Nat.nth Nat.Prime)

```
def IsFirstKPrimesStrongPseudoprime (k n : ℕ) : Prop :=
  Odd n ∧ 3 ≤ n ∧ ¬ Nat.Prime n ∧
    ∀ i < k, IsStrongProbablePrime n (Nat.nth Nat.Prime i)
```

SpspPsi20.psi (SpspPsi20/Defs.lean:43-44) — Section 1 (threshold psi_k via Nat.sInf)

```
noncomputable def psi (k : ℕ) : ℕ :=
  sInf { n : ℕ | IsFirstKPrimesStrongPseudoprime k n }
```

SpspPsi20.IsK2Number (SpspPsi20/Defs.lean:47-48) — Section 1 (Zhang K2 semiprime family n = p(2p-1))

```
def IsK2Number (n p : ℕ) : Prop :=
  Nat.Prime p ∧ Nat.Prime (2 * p - 1) ∧ n = p * (2 * p - 1)
```

SpspPsi20.twoPowLevel (SpspPsi20/Criteria.lean:136-138) — Section 2 (2-power level j_r(b))

```
def twoPowLevel (r b : ℕ) : ℕ :=
  ((Finset.range (padicValNat 2 (r - 1) + 1)).filter
    (fun j => b ^ ((r - 1) / 2 ^ j) ≡ 1 [MOD r])).sup id
```

SpspPsi20.strongLiarCount (SpspPsi20/Criteria.lean:573-574) — Proposition 3 (strong-liar unit count L(n))

```
noncomputable def strongLiarCount (n : ℕ) : ℕ :=
  (strongLiarFinset n).card
```

Selected Concrete Sanity Examples (checked in Lean; examples only, not global identity proofs): SpspPsi20.nth_prime_20 (Nat.nth Nat.Prime 20 = 73); SpspPsi20.n_1_eq (n_1 = 248116283343628915483226065611703482 1); SpspPsi20.n_2_eq (n_2 = 59599022568707998580897292256269923 41).

Supporting Semantic Closure Index (non-Box-1 helpers): SpspPsi20.p_1 (SpspPsi20/Witnesses.lean), SpspPsi20.q_1 (SpspPsi20/Witnesses.lean), SpspPsi20.n_1 (SpspPsi20/Witnesses.lean), SpspPsi20.p_2 (SpspPsi20/Witnesses.lean), SpspPsi20.q_2 (SpspPsi20/Witnesses.lean), SpspPsi20.n_2 (SpspPsi20/Witnesses.lean).

Box 2 — Verbatim Main Theorem Signatures, Context Binders & Applicability

SpspPsi20.main_two_witnesses (SpspPsi20/Witnesses.lean:353-363) src:dc9bc95f8ee11e

```
theorem main_two_witnesses :
    IsK2Number n₁ p₁ ∧ (10 ^ 36 < n₁ ∧ n₁ ≤ 10 ^ 37) ∧ (2 ^ 120 < n₁ ∧ n₁ < 2 ^ 121) ∧
    IsFirstKPrimesStrongPseudoprime 20 n₁ ∧ ¬ IsStrongProbablePrime n₁ 73 ∧
    73 ^ (n₁ - 1) ≡ 1 [MOD n₁] ∧
    IsK2Number n₂ p₂ ∧ (10 ^ 36 < n₂ ∧ n₂ ≤ 10 ^ 37) ∧ (2 ^ 122 < n₂ ∧ n₂ < 2 ^ 123) ∧
    IsFirstKPrimesStrongPseudoprime 20 n₂ ∧ ¬ IsStrongProbablePrime n₂ 73 ∧
    73 ^ (n₂ - 1) ≡ 1 [MOD n₂] ∧
    psi 20 ≤ n₁ ∧ psi 20 < 2 ^ 121 ∧ (∀ k ≤ 20, psi k ≤ n₁)
```

Assumptions: None beyond Box 1 definitions (all 4 Pratt primality proofs and 21 base tests inside Lean kernel). Applicability: Unconditional (closed statement; no outer premise witness needed).

SpspPsi20.k2_fermat_and_sprp_criteria (SpspPsi20/Criteria.lean:1547-1556) src:ff8a982fc87153

```
theorem k2_fermat_and_sprp_criteria {n p b : ℕ} (hk2 : IsK2Number n p) (hp_odd : Odd
  ↪ p)
    [Fact (Nat.Prime (2 * p - 1))] (hcop : Nat.Coprime b n) :
    (b ^ (n - 1) ≡ 1 [MOD n] ↔ legendreSym (2 * p - 1) (b : ℤ) = 1) ∧
    (IsStrongProbablePrime n b ↔
      padicValNat 2 (orderOf (b : ZMod p)) =
        padicValNat 2 (orderOf (b : ZMod (2 * p - 1)))) ∧
    (IsStrongProbablePrime n b ↔ twoPowLevel (2 * p - 1) b = twoPowLevel p b + 1)
```

Assumptions: Odd primes p, q = 2p - 1, n = pq, gcd(b, n) = 1. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SpspPsi20.k2_strongLiar_share_and_base2_cap (SpspPsi20/Criteria.lean:1561-1576) src:223b8620610859

```
theorem k2_strongLiar_share_and_base2_cap {n p : ℕ} (hk2 : IsK2Number n p) (hp_odd :
  ↪ Odd p) :
    ((strongLiarCount n : ℚ) / (Nat.totient n : ℚ) =
      (1 + 2 * (4 : ℚ) ^ (-(padicValNat 2 (p - 1) : ℤ))) / 6) ∧
    (p % 4 = 3 → (strongLiarCount n : ℚ) / (Nat.totient n : ℚ) = 1 / 4 ∧
      ¬ IsStrongProbablePrime n 2) ∧
    (IsStrongProbablePrime n 2 →
      p % 4 = 1 ∧ (2 * p - 1) % 8 = 1 ∧ n % 4 = 1 ∧
        (strongLiarCount n : ℚ) / (Nat.totient n : ℚ) ≤ 3 / 16) ∧
    (padicValNat 2 (p - 1) = 2 →
      (strongLiarCount n : ℚ) / (Nat.totient n : ℚ) = 3 / 16)
```

Assumptions: Odd primes p, q = 2p - 1, n = pq. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SpspPsi20.sieve_admissible_and_wheel (SpspPsi20/Criteria.lean:1588-1595) src:2d120a93c84f87

```
theorem sieve_admissible_and_wheel {b : ℕ} [Fact b.Prime] (hb_odd : Odd b) :
    (Finset.univ.filter
      (fun x : ZMod b => x ≠ 0 ∧ quadraticChar (ZMod b) (2 * x - 1) = 1)).card =
      (if b % 4 = 3 then (b - 1) / 2 else (b - 3) / 2) ∧
    4 * ([3, 5, 7, 11, 13, 17, 19, 23, 29, 31] : List ℕ).prod = 401120980260 ∧
    (([3, 5, 7, 11, 13, 17, 19, 23, 29, 31] : List ℕ).map
      (fun q => if q % 4 = 3 then (q - 1) / 2 else (q - 3) / 2)).prod = 10135125
```

Assumptions: Odd prime b. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SpspPsi20.k2_jacobsthal_and_cells (SpspPsi20/Criteria.lean:1599-1611) src:58c423dac5b13e

```
theorem k2_jacobsthal_and_cells {b : ℕ} [Fact b.Prime] (hb_odd : Odd b) :
    ((∑ x : ZMod b, quadraticChar (ZMod b) x * quadraticChar (ZMod b) (2 * x - 1)) =
      -legendreSym b 2) ∧
    (4 * ((Finset.univ.filter
        (fun x : ZMod b =>
          quadraticChar (ZMod b) x = -1 ∧ quadraticChar (ZMod b) (2 * x - 1) =
            ↪ 1)).card : ℤ) =
      (b : ℤ) - 2 - legendreSym b (-1) + 2 * legendreSym b 2) ∧
    (4 * ((Finset.univ.filter
        (fun x : ZMod b =>
          quadraticChar (ZMod b) x = 1 ∧ quadraticChar (ZMod b) (2 * x - 1) =
            ↪ 1)).card : ℤ) =
      (b : ℤ) - 2 - legendreSym b (-1) - 2 * legendreSym b 2)
```

Assumptions: Odd prime b. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

SpspPsi20.k_gen_legendre_of_ratio (SpspPsi20/Criteria.lean:1049-1064) src:104d73cbc1788b

```
theorem k_gen_legendre_of_ratio {p q a b n c : ℕ}
    [Fact p.Prime] [Fact q.Prime]
    (hp_odd : Odd p) (hq_odd : Odd q)
    (hb : 0 < b) (hcop_ab : Nat.Coprime a b)
    (hrel : b * (q - 1) = a * (p - 1))
    (hn_eq : n = p * q) (hcop_cn : Nat.Coprime c n)
    (hsprp : IsStrongProbablePrime n c) :
    (padicValNat 2 a = padicValNat 2 b →
      legendreSym p (c : ℤ) = legendreSym q (c : ℤ)) ∧
    (padicValNat 2 b < padicValNat 2 a →
      legendreSym q (c : ℤ) = 1) ∧
    (padicValNat 2 a < padicValNat 2 b →
      legendreSym p (c : ℤ) = 1)
```

Assumptions: Odd primes p, q, b > 0, gcd(a, b) = 1, b(q - 1) = a(p - 1), n = pq, gcd(c, n) = 1, SPRP(n, c). Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

Table 2. Lean 4 Type-Trap Audit Checklist — Source-Level Approximation (Gate Replay Pending).[a]

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
IsK2Number	nat_sub	- 1	intended_convention	Prime p >= 2 gives 2*p >= 4 > 1 (no underflow).
IsStrongProbablePrime	nat_sub	- 1	intended_convention	Explicit 3 <= n prevents Nat underflow.
k2_fermat_and_sprp_criteria	nat_sub	- 1	intended_convention	IsK2Number gives p >= 3, q >= 5, n >= 15.
k2_jacobsthal_and_cells	nat_sub	- 1	intended_convention	Subtraction in ZMod b or Int (signed ring).
k2_jacobsthal_and_cells	nat_sub	- 2	intended_convention	Signed Int subtraction (b : Int) - 2.
k2_strongLiar_share_and_base2_cap	nat_div	/ (Nat.totient n : ℚ)	intended_convention	Q division by phi(pq) >= 8 > 0.
k2_strongLiar_share_and_base2_cap	real_log_rpow	^ (-(padicValNat 2 (p - 1) : ℤ))	intended_convention	Integer power 4^(-e) in Q (base 4 != 0).
k2_strongLiar_share_and_base2_cap	nat_div	/ 6	intended_convention	Exact Q division by constant 6.
k2_strongLiar_share_and_base2_cap	nat_mod	% 4	intended_convention	Standard Nat remainder mod 4.
k2_strongLiar_share_and_base2_cap	nat_div	/ 4	intended_convention	Exact Q constant 1/4.
k2_strongLiar_share_and_base2_cap	nat_sub	- 1	intended_convention	Odd prime p >= 3 gives p - 1 >= 2, q >= 5.
k2_strongLiar_share_and_base2_cap	nat_mod	% 8	intended_convention	Standard Nat remainder mod 8.
k2_strongLiar_share_and_base2_cap	nat_div	/ 16	intended_convention	Exact Q constant 3/16.
k_gen_legendre_of_ratio	nat_sub	- 1	intended_convention	Odd primes p, q >= 3 prevent Nat underflow.
q_1	nat_sub	- 1	intended_convention	Concrete prime p1 gives 2*p1 - 1 > 0.
q_2	nat_sub	- 1	intended_convention	Concrete prime p2 gives 2*p2 - 1 > 0.
sieve_admissible_and_wheel	nat_sub	- 1	intended_convention	Odd prime b >= 3 gives b - 1 >= 2 (ZMod 2x-1).
sieve_admissible_and_wheel	nat_mod	% 4	intended_convention	Standard Nat remainder mod 4.
sieve_admissible_and_wheel	nat_div	/ 2	intended_convention	Odd b makes b - 1, b - 3 even (exact / 2).
sieve_admissible_and_wheel	nat_sub	- 3	intended_convention	Else branch has odd prime b % 4 = 1, b >= 5.
twoPowLevel	finset_range	Finset.range	intended_convention	Finset.range(v2(r-1)+1) includes upper endpoint.
twoPowLevel	nat_sub	- 1	intended_convention	Evaluated on odd primes r >= 3 (r - 1 >= 2).
twoPowLevel	nat_div	/ 2	intended_convention	For j <= v2(r - 1), 2^j divides r - 1 exactly.

[a] Covers the specified Lean 4 hazard set (Nat subtraction/division/mod, totalized real/rational division, Real.log/rpow, Finset.range, Classical.choose, local instances); no finite checklist certifies all semantics.

Table 3. Per-Claim Reader's Trust Boundary Ledger (G11 Denominators Derived Mechanically).[a]

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Thm. 1: Two 37-digit K2 witnesses n1 (121b) & n2 (123b); psi_20 <= n1 < 2^121	Lean Kernel	`SpspPsi20.main_two_witnesses`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 pure kernel + Box 1 defs only (all 4 Pratt primality proofs & 21 base tests verified inside kernel).
Thm. 2: K2 Fermat (b/q)=1 and SPRP v2(ord_p b)=v 2(ord_q b) criteria	Lean Kernel	`SpspPsi20.k2_fermat_and_sprp_criteria`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 pure kernel + Box 1 defs (odd primes p, q = 2p - 1, gcd(b, n) = 1).
Prop. 3: Strong-liar share (1 + 2*4^(-e))/6 and base-2 cap <= 3/16	Lean Kernel	`SpspPsi20.k2_strongLiar_share_and_base2_cap`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 pure kernel + Box 1 defs (odd primes p, q = 2p - 1).
Prop. 4: Sieve count A(b) and wheel products 10,135,125 and 401,120,980,260	Lean Kernel	`SpspPsi20.sieve_admissible_and_wheel`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 pure kernel + Mathlib ZMod/quadraticChar (odd prime b).
Thm. 5: Jacobsthal sum -(2/b) and exact b mod 8 cells 4*N_{-+}, 4*N_{++}	Lean Kernel	`SpspPsi20.k2_jacobsthal_and_cells`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 pure kernel + Mathlib legendreSym/quadraticChar (odd prime b).
Prop. 6: General ratio A/B 2-adic quadratic-residue obstruction	Lean Kernel	`SpspPsi20.k_gen_legendre_of_ratio`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 pure kernel + Box 1 defs (odd primes p, q, b > 0, gcd(a, b) = 1, b(q-1) = a(p-1)).
Prop. 7: External 1,024-shard K2 enumeration on (10^36, 10^37] finds 2 hits	Cert Only	recompute.py (76c647740008) [regenerable; replay pending]	External cert only; fidelity: n/a	External GMP C sieve on 1.53e18 ints (64 shards rechecked; barrier: beyond Lean kernel).
Prop. 8: Companion 7-ratio search A/B in {3.. 6,3/2,4/3,5/2} to psi'_17	Cert Only	recompute.py (13c82937ac8e) [regenerable; replay pending]	External cert only; fidelity: n/a	External single-impl C sweep (barrier: ~1.5e10 candidates beyond Lean kernel).
Conj. 9: Zhang minimality psi_t = psi'_t for all t >= 12	Unsettled	Open conjecture	Lean: none; fidelity: n/a	Explicitly open conjecture (excluded from G11 denominators).

[a]Allowed pure-kernel axioms: `propext`, `Classical.choice`, `Quot.sound`. Actual axioms printed above are read directly from verified outputs. Provenance nonce: 6c37829e4767 | Lean mode: committed_receipt | Closure SHA-256: 9ace5e7a2345 | Git anchor: fa70c3374d1a | Axioms SHA-256: c2e238c24511 | Fidelity SHA-256: 581800af6adb.

6. PRIOR WORK AND OPEN QUESTIONS

Jaeschke [3] characterized strong pseudoprimes via 2-adic orders; Arnault [1] built multi-base composites; Bleichenbacher [2] (Table 3.2, $t = 78$) gave a 41-digit 21-base witness ($\approx 11{,}727 \times n_1$). Zhang [6] exhausted K_2 to 10^{36} and stated Conjecture 9; Sorenson and Webster [5] proved $\psi_{12} > 2^{64}$ (certifying factor tests $< 2^{62}$); Pratt [4] gave the certificates in `lean/SpspPsi20/`. Chinese-language journals were not exhaustively searched (no priority claim). Open: $\psi_{20} = n_1$ (Conjecture 9, $t = 20$) and $\psi_{21} < 2^{128}$.

7. DATA AVAILABILITY, REPRODUCIBILITY, AND USE OF AI

Sources live in `papers/spsp-psi20/draft4/` and `lean/SpspPsi20/`. Running `python3 papers/spsp-psi20/draft4/recompute.py` checks the Section 1 box, twelve Pratt nodes, $(10^{36}, 10^{37}]$ count, wheel counts, and Jacobsthal cells (`-s DIR` aggregates archived 1,024-shard summaries; full GMP/C replay pending). AI agents built and directed by Tyler B. Hughes assisted with calculations, Lean 4 proofs, figures, and drafting. No human mathematician has yet refereed the proof.

Drafted: November 2026 **Updated:** November 23, 2026

REFERENCES

(1) Arnault, F. Rabin–Miller primality test: Composite numbers which pass it. *Math. Comp.* **1995**, *64* (209), 355–361. DOI: 10.1090/S0025-5718-1995-1260124-2.

(2) Bleichenbacher, D. Efficiency and security of cryptosystems based on number theory. Ph.D. thesis, Diss. ETH No. 11404, ETH Zürich, **1996**.

(3) Jaeschke, G. On strong pseudoprimes to several bases. *Math. Comp.* **1993**, *61* (204), 915–926. DOI: 10.1090/S0025-5718-1993-1192971-8.

(4) Pratt, V. R. Every prime has a succinct certificate. *SIAM J. Comput.* **1975**, *4* (3), 214–220. DOI: 10.1137/0204018.

(5) Sorenson, J. P.; Webster, J. Strong pseudoprimes to twelve prime bases. *Math. Comp.* **2017**, *86* (304), 985–1003. DOI: 10.1090/mcom/3134.

(6) Zhang, Z. Two kinds of strong pseudoprimes up to 10^{36}. *Math. Comp.* **2007**, *76* (260), 2095–2107. DOI: 10.1090/S0025-5718-07-01977-1.