How Many Positions Suffice for a Seven-Integer Product?

Qingzhen Hu — Erdős–Surányi Interval Product Problem — November 2026

Let $A \subset \mathbb{Z}_{\geq 2}$ be a set of seven distinct integers with maximum $M = \max A$. In any interval I of M consecutive integers, let $\tau(A, I)$ denote the minimum size of a subset $B \subseteq I$ such that $\prod_{a \in A} a \mid \prod_{b \in B} b$, and write $g(7)$ for the maximum of $\tau(A, I)$ over all such A and I. Intervals containing 0 require only $\{0\}$, and negative intervals reduce to positive intervals upon replacing every element by its opposite.

Main Theorem

Theorem 1 (Exact Seven-Modulus Bound). *For every seven-element set $A \subset \mathbb{Z}_{\geq 2}$ and every interval I of $M = \max A$ consecutive integers, at most nine positions suffice, and this bound is sharp:*

$$g(7) = 9.$$

For every valid target set and every starting point of the interval, nine positions are always sufficient; Section 1 rules out eight. The upper bound is a computer-assisted proof across three regimes: $M \leq 512$ and $513 \leq M \leq 12\,347$ are discharged by exhaustive finite certificates, while $M \geq 12\,348$ is proved by structural reductions combined with finite checks independent of the interval location. The only external small-order values used are $g(1), \ldots, g(5) = 1, 2, 4, 5, 6$, due to Tripodo [2] (Theorem 1.6 and the historical results cited therein); the global six-modulus bound $g(6) \leq 8$ is proved internally here. Furthermore, all analytic reductions, the lower-bound witness, and two finite combinatorial checks (all 792 cofactor multisets and all integer partitions of 7) are machine-checked in Lean 4 (Section 7). This note and the accompanying verification bundle have not yet undergone independent third-party peer review.

1 An Interval Requiring Nine Positions

Take

$$A = \{34, 221, 289, 299, 323, 391, 437\}, \qquad I = [96361, 96797] \cap \mathbb{Z}.$$

The target product is $2 \cdot 13^2 \cdot 17^6 \cdot 19^2 \cdot 23^3$. The interval I has length 437 and contains $T = 96577 = 13 \cdot 17 \cdot 19 \cdot 23$. Every other integer in I lies within distance at most 220 of T; since $220 < 13 \cdot 17 = 221$, no other element of I can be divisible by more than one of the four odd primes $13, 17, 19, 23$. Moreover, the unique multiple of 17^2 in I is $96526 = 334 \cdot 17^2$, in which 17 has exact exponent 2, while the adjacent multiples of $23^2 = 529$ are 96278 and 96807, both outside I.

Strategy for T	Extra 13	Extra 17	Extra 19	Extra 23	Lower bound on $\lvert B \rvert$
Include T	1	4	1	2	$1 + 8 = 9$
Omit T	1	5	1	3	10

The four classes of additional elements in $I \setminus \{T\}$ are pairwise disjoint, so any valid subset $B \subseteq I$ has size at least 9. Conversely, adding the nine offsets

$$0, \quad 13, \quad 19, \quad -23, \quad 23, \quad -51, \quad -17, \quad 17, \quad 34$$

to T yields a valid 9-element cover inside I (since $T - 51 = 96526$ supplies 17^2 and $T + 34$ supplies the factor $2 \cdot 17$). Other explicit constructions achieving this lower bound also appear in Bergström [3] (Proposition 4).

Remark. If one interprets the problem literally as requiring *exact* equality $|B| = g(7)$ for every input, taking $A = \{2, \dots, 8\}$ on an interval of length 8 forces $|B| \le 8$, contradicting any fixed count ≥ 9. Throughout this paper we follow the standard "at most" convention $|B| \le g(n)$ [1].

2 Three Fundamental Covering Tools

Write $P = \prod_{a \in A} a$. Whenever a subset of A has maximum element $M' < M$, we may restrict to an arbitrary sub-interval of I of length M' and apply the corresponding smaller-order upper bound. Furthermore, prime-power demands supported on disjoint sets of primes can be satisfied independently and combined by taking unions; overlaps between chosen sets can only preserve or increase p-adic valuations.

Appending one position factor-by-factor. Suppose we have already used at most $j-1$ positions in I to cover a product H, and wish to incorporate an additional divisor $d \le M$. If an unselected multiple of d exists in I, pick it; otherwise every multiple of d in I already belongs to our chosen set, so the current covered product is already divisible by $\operatorname{lcm}(H, d)$ and only $\gcd(H, d)$ remains missing. Therefore, as long as

$$\gcd(H, d) \le \lfloor M/j \rfloor ,$$

the interval I contains at least j multiples of $\gcd(H, d)$, guaranteeing at least one fresh multiple to append. In particular, any sequence $d_1, \dots, d_k$ with $1 \le d_j \le M$, $\prod_{j=1}^{k} d_j = P$, and $\gcd\big(\prod_{i<j} d_i, d_j\big) \le \lfloor M/j \rfloor$ for all $1 \le j \le k$ certifies $\tau(A, I) \le k$ universally across all intervals I of length M. This cumulative-GCD slot criterion also appears in Bergström [3] (Theorem 6). More generally, if $\gcd(H, d)$ factors into several smaller divisors each having enough interval multiples, those missing factors can be paid off sequentially using fresh positions.

Per-prime valuation estimate with anchors. Suppose distinct anchor positions have been pre-selected for specified divisors $h_i \le \lfloor M/i \rfloor$. For a prime p, let c_j denote the number of pre-selected positions guaranteed divisible by p^j. Adding the t remaining positions in I of highest p-adic valuation (or all remaining positions if fewer than t exist) yields a combined p-adic exponent at least

$$\sum_{j \ge 1 \,:\, p^j \le M} \max\big(c_j,\ \min\big(c_j + t,\ \lfloor M/p^j \rfloor\big)\big) .$$

Indeed, the total p-adic valuation is the sum over $j \ge 1$ of the number of chosen multiples of p^j. If the pre-selected positions contain more than c_j multiples of p^j, the right-hand side only increases, and the same batch of t highest-valuation remaining positions contributes simultaneously to every layer p^j. Taking the union over all primes $p \mid P$ covers P.

Global proof of $g(6) \le 8$. For $M \le 512$, we use the exhaustive 6-modulus verification of Section 3. Suppose now that $M \ge 513$.

If every prime dividing any element of A is at most $C = \lfloor M/8 \rfloor$, then every $a \in A$ factors as $a = uv$ with $u, v \le C$: take $(u, v) = (a, 1)$ when $a \le C$; if $a > C$ has a prime factor $p \ge 8$, set $u = p \le C$ and $v = a/p \le M/8$, so $v \le C$; otherwise all prime factors of a are at most 7, and multiplying them sequentially until the partial product first reaches at least 8 produces a divisor $u \in [8, 49]$ (as $7 \cdot 7 = 49 < C$), again leaving $v = a/u \le \lfloor M/8 \rfloor = C$. Using $g(5) = 6$ to cover five elements of A and appending two fresh multiples for u, v covers all six elements in at most 8 positions.

Otherwise some prime $p > M/8$ divides $d \ge 1$ elements of A, say $pc_1, \dots, pc_d$ with distinct cofactors $1 \le c_i \le 7$. First cover the $6-d$ elements outside the group in at most $g(6-d)$ positions ($g(0) = 0$). Then select d distinct multiples of p in I, taking one of them to be a multiple of $p \max c_i$. If that multiple was already chosen, we save one new position and spend one fresh position on $\max c_i$; any other overlap among the d multiples already contributes a factor of p that was not needed

outside the group. Thus we add at most d new positions while paying $p^d \max c_i$. When $d = 1$, we are done ($g(5) + 1 = 7 \le 8$). When $d \in \{2, 3\}$, the product of the remaining $d - 1$ cofactors is at most $5 \cdot 6 = 30 \le \lfloor M/8 \rfloor$, requiring at most one more position ($g(4) + 3 = 8$, $g(3) + 4 = 8$). When $d \in \{4, 5, 6\}$, the remaining cofactors lie in $\{1, \dots, 6\}$, which partition into $\{1, 2, 3, 4\}$ (lcm $12 \le \lfloor M/8 \rfloor$) and $\{5, 6\}$ (product $30 \le \lfloor M/8 \rfloor$), requiring at most two more positions ($g(2) + 6 = 8$, $g(1) + 7 = 8$, $0 + 8 = 8$). Hence $g(6) \le 8$ for all M.

3 Exhaustive Check for Small Maxima: $M \le 512$

Our verification suite first certifies $g(6) \le 8$ for all $M \le 512$, and then certifies $g(7) \le 9$ for all $M \le 512$. Whenever M does not exceed the budget k, taking all of I works because $P \mid M!$ and the product of any M consecutive integers is divisible by $M!$.

Fix M and a budget k. It suffices to inspect n-element subsets A that are upward closed under taking multiples $\le M$: otherwise some $a \in A$ has an unselected multiple $a' \le M$, and replacing a by a' makes the target product divisible by the original one while strictly increasing $\sum_{x \in A} x$. Additional structural prunings (elements $\le \lfloor M/k \rfloor$ and primes $p \ge 11$ dividing at most two moduli) are documented in `finite/README.md`.

Each surviving set is checked against the valuation estimates (zero, one, or two anchors) and cumulative-GCD criteria of Section 2. For any remaining set, define

$$L = \prod_{p \mid P} p^{\min(v_p(P),\, \lfloor \log_p M \rfloor)}.$$

Every integer x is divisible by $\gcd(x, L)$, so a valid cover computed from the truncated contributions $\gcd(x, L)$ guarantees a true cover. For every starting residue $x \bmod L$, `finite/check_shifts.cpp` finds at most k distinct offsets whose $\gcd(x + j, L)$ values cover P. Because such a cover remains valid until just before its leftmost selected element leaves the shifting window, the checker skips entire contiguous blocks of starting residues at once. All sets pass; Table 1 summarizes the run.

Finite verification metric ($M \le 512$)	Six moduli (≤ 8 slots)	Seven moduli (≤ 9 slots)
Sets entering per-set checks	5 965	7 886 270
Sets reaching CRT shift check	1	22
Largest CRT period L	514 800	10 810 800

Table 1: Summary of exhaustive finite checks for $M \le 512$ (`finite/run.py`).

Eliminating primes $p > M/9$ in the intermediate and long ranges. Suppose $M \ge 513$ and some prime $p > M/9$ divides d elements of A, written pc_i with $h = \max c_i \le 8$ and $R = \prod c_i$. First cover the $7 - d$ elements outside the group using at most $t = g(7 - d)$ positions ($g(0) = 0$). The h consecutive multiples of p in I have quotients $x + 1, \dots, x + h$ for some integer x. Enumerate every overlap subset $O \subseteq \{1, \dots, h\}$ of size $|O| \le t$ (indices whose p-multiples were already selected outside the group), and search for a disjoint subset $S \subseteq \{1, \dots, h\} \setminus O$ satisfying $|O| + |S| \ge d$, which pays p^d in full while crediting cofactor divisors only on the fresh positions S.

Let $Q = \operatorname{lcm}\{u \le h : u \mid R\}$. For each $x \bmod Q$, divide R by $\gcd\big(R, \prod_{i \in S} \gcd(x + i, Q)\big)$. Whenever the quotient splits into at most $9 - t - |S|$ factors each at most 15, appending fresh multiples finishes the cover because $\lfloor M/9 \rfloor \ge 15$ for all $M \ge 135$. Running `finite/check_high_groups.cpp` verifies all 254 cofactor sets, all 42 167 $(C, x \bmod Q)$ pairs, and all 2 237 536 overlap cases with 0 failures. Consequently, in every remaining case we may assume all prime divisors of elements of A are at most $M/9$.

4 Compressing Potential Counterexamples into Prime Pairs

Let $M \geq 513$ and suppose A requires at least 10 positions. By Section 3, every prime divisor of P is at most $M/9$.

First, every prime $p \geq 11$ must divide at least three distinct elements of A. Indeed, if p divides only one element $a = p^e c$, cover the remaining six elements in at most $g(6) \leq 8$ positions and pick a multiple of a; if already chosen, p^e is covered and appending one fresh multiple of $c \leq M/11 \leq \lfloor M/9 \rfloor$ suffices. If p divides exactly two elements $p^e c$ and $p^f d$ $(e \geq f)$, cover the other five elements in $g(5) = 6$ positions, pick a multiple of $p^e c$, and pick another distinct multiple of p^f (which exists since two elements of A are divisible by p^f). If the first multiple is fresh, one fresh multiple of d suffices; if already selected, fresh multiples of c and d suffice. Both cases use at most 9 positions.

Second, no two distinct elements $a, b \in A$ can have product factoring into three integers bounded respectively by $\lfloor M/7 \rfloor$, $\lfloor M/8 \rfloor$, and $\lfloor M/9 \rfloor$: otherwise covering the remaining five elements in 6 positions and appending three fresh multiples settles A in 9 positions. In particular, if $p \geq 11$ satisfies $p^2 \leq M/7$, choosing two multiples $a, b \in A$ of p and factoring $ab = p^2 \cdot (a/p) \cdot (b/p)$ puts $p^2 \leq M/7$ in the first bin and $a/p, b/p \leq M/11 \leq \lfloor M/9 \rfloor$ in the remaining two bins. Therefore every prime factor $p \geq 11$ in a counterexample satisfies

$$\sqrt{M/7} < p \leq M/9.$$

We refer to primes in this range as *large prime factors*, counted with multiplicity. Every element of A contains at most two large prime factors, since $(M/7)^{3/2} > M$ for $M > 343$.

For $M \geq 12\,348$, every element contains exactly two large prime factors. Suppose some element $a \in A$ contains at most one large prime factor, and pick any second element $b \in A$. Together a and b have at most three large prime factors. Insert the prime factors of ab in descending order greedily into the three bins of capacities $\lfloor M/7 \rfloor$, $\lfloor M/8 \rfloor$, $\lfloor M/9 \rfloor$. If r is the first prime factor that fails to fit in any bin, then all three bins are already nonempty (as $r \leq M/9$) and contain partial products strictly exceeding $M/(7r)$, $M/(8r)$, and $M/(9r)$. Since at least four prime factors of size at least r have now been encountered, r cannot be a large prime, so $r \leq 7$ and

$$ab > \frac{M^3}{504r^2} \geq \frac{M^3}{24\,696}.$$

When $M \geq 24\,696$, this contradicts $ab \leq M^2$. The intermediate range $12\,348 \leq M \leq 24\,695$ is settled by the finite pair certificate below.

Finite certificate when some element has at most one large prime factor ($12\,348 \leq M \leq 24\,695$). For each integer $2 \leq a \leq 24\,695$, any interval length $M \in [12\,348, 24\,695]$ in which a can remain eligible must satisfy

$$\max\big(\{12\,348, a\} \cup \{9p : p \geq 11,\ p \mid a\}\big) \leq M \leq \min\big(\{24\,695\} \cup \{7p^2 - 1 : p \geq 11,\ p \mid a\}\big).$$

Discard a if this range is empty or if a has more than two prime factors ≥ 11. For every pair of distinct candidates a, b whose combined count of primes ≥ 11 is at most 3 and whose M-ranges overlap, let m be the minimum of the overlap. Since bin capacities increase monotonically with M, it suffices to check 3-bin packing at $M = m$.

Whenever $24\,696ab \leq m^3$, the greedy inequality above already guarantees 3-bin packability; for every remaining pair, `large/verify_pairs.cpp` performs exact backtracking over all placements of the prime factors of ab into $(\lfloor m/7 \rfloor, \lfloor m/8 \rfloor, \lfloor m/9 \rfloor)$. The inequality screens 44 408 626 pairs and exact backtracking verifies all 6 249 278 remaining pairs, with zero failures.

Consequently, in any counterexample with $M \geq 12\,348$, all seven elements take the form

$$a_i = c_i u_i v_i, \qquad 1 \leq c_i \leq 6,$$

where $u_i, v_i > \sqrt{M/7}$ are large primes (possibly equal), forcing $c_i < 7$ from $u_i v_i > M/7$.

Reserving capacity for the final four slots. Sort the cofactors $c_1, \ldots, c_7$ in nondecreasing order and let $c_{(3)}$ denote the third smallest cofactor. Every large prime p divides at least three elements of A, hence at least one element $a_j = c_j u_j v_j$ with $c_j \geq c_{(3)}$. Since its partner large prime in a_j exceeds $\sqrt{M/7}$, we have $c_j p\sqrt{M/7} < a_j \leq M$, so $p^2/M < 7/c_j^2 \leq 7/c_{(3)}^2$. Combined with $c_i p\sqrt{M/7} < a_i \leq M$ inside a_i itself, and noting that the smaller of $\{u_i, v_i\}$ satisfies $c_i \min(u_i, v_i)^2 \leq a_i \leq M$, the ratios u_i^2/M and v_i^2/M are bounded respectively by

$$R_i = \frac{7}{\max(c_i, c_{(3)})^2}, \qquad S_i = \min\left(R_i, \frac{1}{c_i}\right),$$

where S_i applies to the smaller prime factor. As shown in Section 5, five positions suffice to pay five pairs of large prime factors, leaving four remaining large prime tokens. Taking the four largest values among the fourteen bounds $\{R_i, S_i\}_{i=1}^7$ in nonincreasing order, the j-th largest remaining prime w_j has w_j^2/M bounded by the j-th largest bound.

We may assign these four remaining primes to slots $k = 6, 7, 8, 9$ in any permutation. If slot k receives a prime w subject to upper bound R, then $w \leq \sqrt{MR}$, so slot k can simultaneously absorb an extra cofactor divisor up to $\left\lfloor \sqrt{12\,348/(k^2 R)} \right\rfloor$, since the product of w and this extra factor is at most $\sqrt{MR} \cdot \sqrt{M/(k^2 R)} = M/k$. Across all 792 unordered multisets $(c_1, \ldots, c_7) \in \{1, \ldots, 6\}^7$, `large/verify_cofactors.py` enumerates every permutation of the four largest upper bounds in exact rational and integer square-root arithmetic and verifies that the prime factors of $\prod_{i=1}^7 c_i$ pack into the four extra-factor capacities.

5 Certificate for Five Pairings

View the seven large-prime pairs $\{u_i, v_i\}$ as the edges of a multigraph (allowing parallel edges and self-loops). For any connected component with $e \in \{1, \ldots, 7\}$ edges, we require

$$t(1..7) = (1, 2, 2, 3, 4, 4, 5)$$

pairings, respectively. If an edge uv appears with multiplicity m, the corresponding elements of A have distinct cofactors c, so $muv \leq M$ (in fact $c \geq m$ for the largest cofactor among them), ensuring at least m multiples of uv in I. Moreover, any two primes $\leq \sqrt{M}$ satisfy $uv \leq M$ and can be paired at least once even if not joined by an original edge.

Pairings are performed sequentially under two rules:

- **Distinct primes $u \neq v$.** At least one endpoint must be fresh (not yet used in prior pairings). Suppose we pick ℓ distinct multiples of uv in I, where ℓ does not exceed the guaranteed multiplicity of uv or the remaining degree demands of u and v. Say u is fresh; if k of the ℓ chosen positions collide with previously selected positions, the net increase in positions is $\ell - k$, all ℓ positions supply u^ℓ, and each of the k colliding positions only needs a fresh multiple of v to repair it, adding at most k positions for a net cost of at most ℓ new positions.
- **Self-loop u^2.** The vertex u must have prior paired count at most 1. Taking a multiple of u^2 either adds one new position or, if already selected, lacks at most one factor of u, repaired by one fresh multiple of u.

All single-prime repair factors satisfy $u, v \leq M/9$, so fresh multiples always exist within the first 9 positions.

If a connected component has more than $t(e)$ vertices, greedily taking $t(e)$ edges along a spanning tree that introduce new vertices pays $t(e)$ pairs in $t(e)$ positions directly. Otherwise the component has $1 \leq v \leq t(e)$ vertices, of which $1 \leq s \leq v$ are at most $\sqrt{M}$ (every original edge $u_i v_i \leq M$ must

have at least one endpoint $\leq \sqrt{M}$). Exhaustively enumerating all e-edge labeled connected multigraphs without isolated vertices where every edge has an endpoint in $\{1, \ldots, s\}$, and testing recursive pairing bounded by vertex degrees (self-loops counted twice), `large/verify_multigraphs.py` confirms that all cases succeed (Table 2).

Edges e	1	2	3	4	5	6	7
Required pairs $t(e)$	1	2	2	3	4	4	5
Labeled component configs	1	6	10	107	1414	4165	85 799

Table 2: Exhaustive multigraph verification (`large/verify_multigraphs.py`).

For every integer partition $e_1 + \cdots + e_r = 7$, the sum $\sum_j t(e_j)$ is at least 5. Executing pairings component-by-component and stopping as soon as five pairs are reached uses at most five positions and leaves exactly four large prime tokens. Since connected components have disjoint vertex sets, single-prime repairs avoid all previously chosen positions across components. Assigning the remaining four large primes and all cofactors c_i via Section 4 yields $5 + 4 = 9$ positions, completing the proof for $M \geq 12\,348$.

6 Intermediate Exhaustive Enumeration: $513 \leq M \leq 12\,347$

By Section 4, for each $M \in [513, 12\,347]$ we define an incompatibility graph whose vertices are eligible integers $a \leq M$, connecting a and b whenever ab fails to pack into capacities $\lfloor M/7 \rfloor$, $\lfloor M/8 \rfloor$, $\lfloor M/9 \rfloor$. Any counterexample must form a 7-clique containing M. Only candidates with $a > M/7$ need to be retained: if $a \leq M/7$, any other element $b \leq M$ factors into two divisors $\leq \lfloor M/9 \rfloor$ just as in Section 2 (if b has no prime factor ≥ 9, the minimal small-prime product reaching 9 is at most $8 \cdot 7 = 56 \leq \lfloor 513/9 \rfloor$), so ab packs into three bins.

For each M, `gap/enumerate.cpp` extends cliques in descending order starting from M. Candidates may only contain large primes ($p \geq 11$) satisfying $9p \leq M < 7p^2$. Each occurring large prime must divide at least three chosen moduli, and each modulus holds at most two, so at most four distinct large primes can appear; branches whose remaining candidates cannot supply these incidences are pruned. Within each M, cliques yielding identical target products are deduplicated. Across all 11 835 values of M, exactly 4 290 admit M as an eligible maximum, generating 183 795 332 leaf 7-cliques and 6 957 902 distinct targets, all resolved by five methods (Table 3).

Resolution method (deduplicated by target product per M)	Targets resolved
Appending one position to a 6-modulus cover (`one_modulus`)	11 671
Independent per-prime valuation bounds (`separate_primes`)	312 845
Appending two positions to a 5-modulus cover (`two_moduli`)	3 731 280
Five large-prime pairs plus 4-slot tail packing (`five_pairs`)	2 886 748
Explicit cumulative-GCD slot sequence (`terminal.py`)	15 358
Total distinct targets	6 957 902

Table 3: Resolution breakdown for $513 \leq M \leq 12\,347$ (`gap/verify.py`).

The fourth criterion applies only when every element has two large prime factors, exactly as in Section 4. The fifth criterion outputs the original 7-tuple and verifies an explicit sequence $d_1, \ldots, d_s$ ($s \leq 9$) satisfying Section 2. For instance, the very first terminal row at $M = 516$ is

$$A = \{387, 425, 430, 459, 500, 510, 516\}, \qquad (d_1, \ldots, d_8) = (43, 43, 215, 375, 425, 289, 480, 486),$$

where $\prod_{j=1}^{8} d_j = \prod A$ and $\gcd(\prod_{i<j} d_i, d_j) \leq \lfloor 516/j \rfloor$ for all $1 \leq j \leq 8$ (so 8 positions even suffice

here). All 15 358 terminal certificates are archived in `gap/terminal_certificates.tsv.gz` and can be regenerated and verified from scratch.

Combining Section 3 ($M \leq 512$), Section 6 ($513 \leq M \leq 12\,347$), and Sections 4–5 ($M \geq 12\,348$), we conclude $\tau(A, I) \leq 9$ for all $M \geq 8$ (note that seven distinct integers ≥ 2 automatically force $M \geq 8$). Together with the explicit 9-position witness in Section 1, this establishes Theorem 1: $g(7) = 9$.

7 Lean 4 Formalization

To independently audit all logical reductions, case splits, and algebraic bounds, the architecture of this proof has been formalized in Lean 4 (`v4.36.0`) with Mathlib4 (`v4.36.0`) inside `formalization/` (8 files, 5 060 lines; Table 4). The entire project builds cleanly under `lake build` with zero occurrences of `sorry`, `admit`, `native_decide`, or custom global axioms.

Module (`Erdos708/`)	Lines	Key Proved Results
`Basic.lean`	243	Arbitrary-to-positive reduction, factorial bound $M \leq k$, $g_+(1) \leq 1$
`LowerBound.lean`	271	Five-digit witness ($\neg g(7) \leq 8$), 9-cover, exact-equality refutation
`Tools.lean`	421	Cumulative-GCD steps, $M = 516$ example, $g_+(2) \leq 2$, valuations
`SixBound.lean`	604	`SmallOrderInputs`, $M \geq 513$ analytic proof of $g(6) \leq 8$
`LargeReduction.lean`	1 372	Greedy 3-bin pack, two-prime factoring, kernel 792-multiset check
`FivePairing.lean`	1 916	Kernel partition bound (≥ 5), collision repairs, $5 + 4$ cover theorem
`Main.lean`	226	`FiniteCertificates` assembly and main theorem `erdos_708_g7_eq_9`

Table 4: Modular structure of the Lean 4 formalization (`Erdos708.lean`, 7 lines root import, 5 060 lines total).

Two small-scale exhaustive checks embedded within Sections 4–5 are discharged directly inside Lean's kernel via `decide`: all 792 cofactor multisets in `cofactor_792_kernel_check` (verifying the 4-slot tail capacity table) and all positive partitions of 7 in `partition_sum_ge_five` (verifying $\sum_j t(e_j) \geq 5$). The final exported theorem `Erdos708.erdos_708_g7_eq_9` depends exclusively on Lean's standard core axioms (`propext`, `Classical.choice`, `Quot.sound`), with all remaining external dependencies passed explicitly as two `Prop`-valued hypotheses: `SmallOrderInputs` ($g_+(3) \leq 4, g_+(4) \leq 5, g_+(5) \leq 6$) and `FiniteCertificates` (the six large-scale computational searches reproduced bit-for-bit by `replay.py`).

Acknowledgments and AI Disclosure

This repository was initiated, curated, and released by Qingzhen Hu. The proof strategy, C++17 and Python verification software, Lean 4 formalization, certificate generation, and English exposition were developed with AI assistance. All mathematical reductions can be verified line by line from the text and machine-checked in `formalization/`, and every finite search and explicit certificate is independently reproducible without floating-point arithmetic via `replay.py`. As noted above, this manuscript has not yet undergone independent third-party peer review.

References

[1] T. F. Bloom, *Erdős Problem #708*, https://www.erdosproblems.com/708 (originally posed in P. Erdős and J. Surányi, *Megjegyzések egy versenyfeladathoz*, Matematikai Lapok **10** (1959), 39–48).

[2] Tripodo, *Theorem 1.6 and historical bounds on* $g(1), \ldots, g(5)$, https://hexagonmath.org/2611.00694v1.

[3] C. Bergström, *Exact Small-n Values ($g(4) = 5, g(5) = 6, g(6) = 8$) and Constructive Lower Bounds ($g(7) \geq 9, g(8) \geq 11$) for Erdős Problem 708*, Zenodo Preprint (November 2026), https://doi.org/10.5281/zenodo.24366897.