A Greatest Common Divisor Criterion of Certain Binomial Coefficients

Dakai Guo[1], Ruichen Qiu[1], Yichuan Cao[1], Ruyong Feng[1], Xiao-Shan Gao[1]

June 23, 2026

Abstract

The binomial greatest common divisor (gcd) criterion recorded as OEIS A080170 is proven. The criterion also appears as conjecture (17) in Ralf Stephan's list of OEIS conjectures. For $k \geq 2$, put

$$D(k) = \gcd_{2 \leq q \leq k+1} \binom{qk}{k}, \qquad n = k + 1.$$

If P is the largest prime-power component p^a exactly dividing n, then the criterion asserts

$$D(k) = 1 \iff \frac{n}{P} > P.$$

The proof is formalized in Lean and the Lean artifact is accepted as part of the Formal Conjectures project. Both the natural-language proof and the Lean formalization are generated by the MechMath Agent Team, an AI agent developed by the authors.

2020 Mathematics Subject Classification. Primary 11B65; Secondary 11A05, 05A10.

Keywords. binomial coefficients, greatest common divisors, Lucas' theorem, OEIS, formalized mathematics.

1 Introduction

Let

$$D(k) = \gcd_{2 \leq q \leq k+1} \binom{qk}{k}$$

for $k \geq 2$. The On-Line Encyclopedia of Integer Sequences (OEIS) is a database of integer sequences, together with descriptions, formulae, programs, references, and links [12]. OEIS A080170 records the values of k for which $D(k) = 1$, together with a conjectural description in terms of the factorization of $k + 1$ [10]. This binomial gcd criterion was formulated by Ralf Stephan as conjecture (17) in his collection, *Prove or Disprove: 100 Conjectures from the OEIS* [13]. The purpose of this note is to prove that description.

The theorem fits into a broader line of work on common divisors of selected binomial coefficients. A classical starting point is Ram's theorem that

$$\gcd_{0<i<n} \binom{n}{i}$$

is p if n is a power of the prime p, and is 1 otherwise [11]. The same row-gcd criterion appears, for example, in the work of Brown and Peterson and in the modern formulation of Lü and Panov [1, 5]. Other work changes the selected set of binomial coefficients: Joris, Oestreicher, and Steinig studied gcds of certain finite regions of Pascal's triangle [4]; McTague studied gcds of the subfamily $\binom{n}{q}, \binom{n}{2q}, \binom{n}{3q}, \ldots$ [7], and Hong

[1]State Key Laboratory of Mathematical Sciences, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, and University of Chinese Academy of Sciences. This work is supported by the Strategic Priority Research Program of Chinese Academy of Sciences under Grant XDA0480502 and XDA0480503.

studied further families selected by arithmetic restrictions on the lower index [3]. The present problem is different: the lower entry is fixed at k, while the upper entry varies through the multiples qk.

For a positive integer n and a prime p, write $p^a \| n$ when $p^a \mid n$ and $p^{a+1} \nmid n$. We call such p^a an exact prime-power component of n, and define

$$\operatorname{ppart}(n) = \max\{p^a : p^a \| n\}$$

for $n > 1$. The main result is the following theorem, which proves OEIS A080170.

Theorem 1. *Let $k \geq 2$ and put $n = k + 1$. Then*

$$\gcd_{2 \leq q \leq k+1} \binom{qk}{k} = 1 \quad \Longleftrightarrow \quad \frac{n}{\operatorname{ppart}(n)} > \operatorname{ppart}(n).$$

The proof first shows, by a finite-difference argument, that every prime divisor of $D(k)$ divides $k + 1$. For primes $p \mid k + 1$, Lucas' theorem reduces the problem to a digitwise stabilizer statement for a finite box of base-p digits. The resulting p-primary criterion is $p \mid D(k)$ if and only if $n/p^a \leq p^a$, and the theorem follows by checking the largest exact prime-power component of n.

This problem also has a formalization motivation. Problem A080170 is not only an OEIS problem from Stephan's collection, but also a member of `FC100OpenSet1`, a distinguished subset of 100 open research problems in the Formal Conjectures benchmark. The update associated with this proof therefore changes the category count for that subset from 96 open and 4 solved entries to 95 open and 5 solved entries. Thus, the result provides a test case in which a previously open research-level formal statement receives both a mathematical proof and a machine-checkable Lean proof.

Formal Conjectures is an open Lean 4 repository and benchmark of formalized mathematical conjectures [2, 14]. The project paper reports a benchmark with 2615 formalized mathematical problem statements, including 1029 open research conjectures and 836 solved problems for proof autoformalization [2]. Its stated goals include providing benchmark problems for automated theorem proving and autoformalization, clarifying the exact meaning of conjectures through formalization, and identifying definitions that should enter mathlib [14].

Formal Conjectures is an open benchmark for verified mathematical discovery [2, 14]. The present theorem is one of the OEIS problems selected for that benchmark, originating from Stephan's list of one hundred OEIS conjectures [13]. More specifically, `gcdCondition_iff_primePowerCondition`, in the `OeisA80170` namespace, is listed in the subset file `FC100OpenSet1.lean`. The associated pull request changes its tag from `research open` to `research solved` and updates the verified category counts for that 100-problem subset from 96 open and 4 solved entries to 95 open and 5 solved entries. Thus, the Lean development turns the A080170 entry from an open formal-conjecture instance into a solved one.

Both the natural-language proof and the Lean formalization are generated by the MechMath Agent Team developed by the authors [8]. MechMath Agent Team is a large language model driven agent designed to generate proofs for mathematical theorems expressed in both natural language and formal language in Lean.

The rest of the paper is organized as follows. In Section 2, the proof of Theorem 1 is given. In Section 3, the Lean formalization of the proof is described. In Section 4, conclusions are presented..

2 Proof of Theorem 1

In this section, the proof of Theorem 1 is presented in several steps. The only standard external input is Lucas's theorem [6]; for broader background on Lucas-type congruences, see the survey [9]. For completeness, we include the short polynomial proof of the non-vanishing criterion used here.

2.1 Preliminaries

For an integer x and a positive integer M, let $[x]_M$ denote the least nonnegative residue of x modulo M.

Lemma 2 (Newton interpolation by forward differences)**.** *Let f be a polynomial of degree at most d, and define*

$$\Delta f(x) = f(x + 1) - f(x).$$

For every integer a and every x,

$$f(x) = \sum_{r=0}^{d} \binom{x-a}{r} \Delta^r f(a).$$

Proof. Let

$$g(x) = \sum_{r=0}^{d} \binom{x-a}{r} \Delta^r f(a).$$

Both f and g have degree at most d. It is enough to show that they agree at $x = a + j$ for $j = 0, \ldots, d$.

For $r \geq 0$,

$$\Delta^r f(a) = \sum_{i=0}^{r} (-1)^{r-i} \binom{r}{i} f(a+i).$$

Thus, at $x = a + j$,

$$g(a+j) = \sum_{r=0}^{j} \binom{j}{r} \sum_{i=0}^{r} (-1)^{r-i} \binom{r}{i} f(a+i).$$

The coefficient of $f(a+i)$ in this expression is

$$\sum_{r=i}^{j} \binom{j}{r} (-1)^{r-i} \binom{r}{i} = \binom{j}{i} \sum_{u=0}^{j-i} (-1)^u \binom{j-i}{u}.$$

This coefficient is 1 if $i = j$ and 0 otherwise. Therefore $g(a+j) = f(a+j)$ for $j = 0, \ldots, d$. Hence $f = g$. □

Lemma 3 (Lucas non-vanishing criterion)**.** *Let p be prime, and write*

$$N = \sum_i N_i p^i, \qquad K = \sum_i K_i p^i$$

with $0 \leq N_i, K_i < p$. Then

$$\binom{N}{K} \not\equiv 0 \pmod{p} \quad \Longleftrightarrow \quad K_i \leq N_i \text{ for every } i.$$

Proof. Work in the polynomial ring $\mathbb{F}_p[X]$. Since $(1+X)^p = 1 + X^p$ in characteristic p,

$$(1+X)^N = \prod_i (1+X)^{N_i p^i} = \prod_i (1+X^{p^i})^{N_i}.$$

The coefficient of X^K in the last product is

$$\prod_i \binom{N_i}{K_i},$$

where a factor is interpreted as 0 if $K_i > N_i$. Each nonzero factor is a nonzero element of $\mathbb{F}_p$, because $0 \leq K_i \leq N_i < p$. Therefore the coefficient, which is $\binom{N}{K} \bmod p$, is nonzero if and only if $K_i \leq N_i$ for all i. □

2.2 Primes in the gcd

Lemma 4. *Let $k \geq 1$ and*

$$D(k) = \gcd_{2 \leq q \leq k+1} \binom{qk}{k}.$$

If a prime p divides $D(k)$, then $p \mid k+1$.

Proof. Set

$$F(x) = \binom{kx}{k}.$$

This is a polynomial in x of degree k, and

$$F(0) = 0, \qquad F(1) = 1, \qquad F(q) = \binom{qk}{k}.$$

Assume that p divides $D(k)$. Then

$$F(2) \equiv F(3) \equiv \cdots \equiv F(k+1) \equiv 0 \pmod{p}.$$

For $r = 0, \ldots, k$,

$$\Delta^r F(1) = \sum_{j=0}^{r} (-1)^{r-j} \binom{r}{j} F(1+j).$$

Modulo p, all terms with $j \geq 1$ vanish, while $F(1) = 1$. Hence

$$\Delta^r F(1) \equiv (-1)^r \pmod{p}.$$

By Lemma 2, evaluated at $x = 0$ and $a = 1$,

$$F(0) = \sum_{r=0}^{k} \binom{-1}{r} \Delta^r F(1).$$

Since $\binom{-1}{r} = (-1)^r$, we obtain

$$0 = F(0) \equiv \sum_{r=0}^{k} (-1)^r (-1)^r = k+1 \pmod{p}.$$

Thus $p \mid k+1$. □

2.3 Digit boxes

Fix a prime p and an integer $L \geq 1$. Put $P = p^L$. If $0 \leq c < P$ has base-p expansion

$$c = \sum_{i=0}^{L-1} c_i p^i, \qquad 0 \leq c_i < p,$$

define the digit box

$$\mathcal{D}_c = \left\{ \sum_{i=0}^{L-1} y_i p^i : \ 0 \leq y_i \leq c_i \text{ for every } i \right\}.$$

Thus $0, c \in \mathcal{D}_c$, and every element of $\mathcal{D}_c$ lies in $[0, c]$.

Lemma 5 (Gap bound)**.** *Let $P = p^L$ and $0 \leq c < P$. If the elements of $\mathcal{D}_c$ are listed in increasing order, every gap between two consecutive elements is at most P/p.*

Proof. We argue by induction on L. For $L = 1$, the set is $\{0, 1, \ldots, c\}$, so all gaps are $1 = P/p$.

Assume $L > 1$ and write $P = pH$, where $H = p^{L-1}$. Write

$$c = dH + r, \qquad 0 \leq d < p, \quad 0 \leq r < H.$$

Then

$$\mathcal{D}_c = \bigcup_{a=0}^{d} (aH + \mathcal{D}_r).$$

Inside each block $aH + \mathcal{D}_r$, the induction hypothesis gives gaps at most $H/p \leq H = P/p$. The last element of $aH + \mathcal{D}_r$ is $aH + r$, and the first element of the next block is $(a+1)H$. The gap between them is

$$(a+1)H - (aH + r) = H - r \leq H = P/p.$$

Therefore all consecutive gaps are at most P/p. □

Lemma 6 (No nonzero translation)**.** *Let $P = p^L$ and $0 \le c < P - P/p$. Suppose $0 \le S \le c$ and*

$$[z + S]_P \in \{0, 1, \dots, c\} \qquad \text{for every } z \in \mathcal{D}_c.$$

Then $S = 0$.

Proof. Assume $S > 0$.

If $S < P - c$, then $c \in \mathcal{D}_c$ gives

$$[c + S]_P = c + S \in \{c + 1, \dots, P - 1\},$$

contradicting the hypothesis.

It remains to consider $S \ge P - c$. Since $S \le c$, the open interval

$$I = (c - S,\ P - S)$$

is contained in $[0, c]$. Its length is

$$(P - S) - (c - S) = P - c > P/p.$$

If I contained no element of $\mathcal{D}_c$, then the consecutive elements of $\mathcal{D}_c$ immediately surrounding I would have a gap of length greater than P/p, contradicting Lemma 5. Hence there is some $z \in \mathcal{D}_c$ with

$$c - S < z < P - S.$$

Then

$$c < z + S < P,$$

so $[z + S]_P \notin \{0, 1, \dots, c\}$, again contradicting the hypothesis. Thus $S = 0$. □

Theorem 7 (Digit-box stabilizer)**.** *Let $P = p^L$ and $0 < c < P - P/p$. Let s be coprime to p. Then*

$$[sy]_P \in \{1, \dots, c\} \quad \text{for every } y \in \mathcal{D}_c \setminus \{0\}$$

if and only if

$$s \equiv 1 \pmod{p^{L - \operatorname{ord}_p(c)}}.$$

Proof. First assume $s \equiv 1 \pmod{p^{L-\operatorname{ord}_p(c)}}$. Put $v = \operatorname{ord}_p(c)$. The lowest v base-p digits of c are 0, so every $y \in \mathcal{D}_c$ is divisible by p^v. Hence $(s - 1)y$ is divisible by $p^{L-v}p^v = P$, and

$$[sy]_P = y.$$

For $y \ne 0$ this lies in $\{1, \dots, c\}$.

Conversely, assume

$$[sy]_P \in \{1, \dots, c\} \qquad (y \in \mathcal{D}_c \setminus \{0\}).$$

We prove the desired congruence by induction on L.

If $L = 1$, then $P = p$ and $0 < c < p - 1$. For $p = 2$ there is no such c. For odd p, $\mathcal{D}_c = \{0, 1, \dots, c\}$. Multiplication by s is injective on the nonzero residues modulo p, so the image of $\{1, \dots, c\}$ is exactly $\{1, \dots, c\}$. Taking sums modulo p gives

$$s\,\frac{c(c+1)}{2} \equiv \frac{c(c+1)}{2} \pmod{p}.$$

Because $0 < c < p - 1$, the factor $c(c + 1)/2$ is nonzero modulo p, so $s \equiv 1 \pmod{p}$. This is the asserted congruence for $L = 1$.

Now let $L > 1$ and write $P = pP'$, where $P' = p^{L-1}$. Decompose

$$c = c_0 + pC, \qquad s = s_0 + pS,$$

with $0 \le c_0 < p$, $1 \le s_0 < p$, and $0 \le S < P'$.

If $c_0 = 0$, then $c = pC$ and

$$\mathcal{D}_c = p\mathcal{D}_C.$$

For every nonzero $z \in \mathcal{D}_C$, the element pz belongs to $\mathcal{D}_c \setminus \{0\}$. The hypothesis gives

$$[s(pz)]_P = p\,[sz]_{P'} \in \{1, \dots, pC\}.$$

Thus

$$[sz]_{P'} \in \{1, \dots, C\} \qquad (z \in \mathcal{D}_C \setminus \{0\}).$$

Also $C < P' - P'/p$, since $pC < P - P/p = (p-1)P'$. By induction,

$$s \equiv 1 \pmod{p^{(L-1)-\mathrm{ord}_p(C)}}.$$

As $\mathrm{ord}_p(c) = 1 + \mathrm{ord}_p(C)$, this is exactly

$$s \equiv 1 \pmod{p^{L-\mathrm{ord}_p(c)}}.$$

It remains to handle $c_0 > 0$. Then $\mathrm{ord}_p(c) = 0$, so we must prove $s \equiv 1 \pmod{P}$.

First consider elements pz with $z \in \mathcal{D}_C \setminus \{0\}$. The same division by p as above gives

$$[sz]_{P'} \in \{1, \dots, C\} \qquad (z \in \mathcal{D}_C \setminus \{0\}).$$

If $C > 0$, induction and the already-proved forward direction imply

$$[sz]_{P'} = z \qquad (z \in \mathcal{D}_C).$$

If $C = 0$, this conclusion is also true, because $\mathcal{D}_C = \{0\}$.

Now fix $a \in \{1, \dots, c_0\}$ and $z \in \mathcal{D}_C$. The element $a + pz$ lies in $\mathcal{D}_c \setminus \{0\}$. Write

$$s_0 a = r_a + p e_a, \qquad 0 \le r_a < p.$$

Using $[sz]_{P'} = z$, the quotient after the lowest base-p digit of $[s(a+pz)]_P$ is

$$[z + Sa + e_a]_{P'}.$$

Since $[s(a+pz)]_P \le c_0 + pC$, this quotient is at most C. Hence

$$[z + Sa + e_a]_{P'} \in \{0, \dots, C\} \qquad (z \in \mathcal{D}_C).$$

Putting $z = 0$ shows that

$$R_a := [Sa + e_a]_{P'} \in \{0, \dots, C\},$$

and the last display becomes

$$[z + R_a]_{P'} \in \{0, \dots, C\} \qquad (z \in \mathcal{D}_C).$$

By Lemma 6, applied with modulus P' and digit bound C, we have $R_a = 0$. Therefore

$$Sa + e_a \equiv 0 \pmod{P'} \qquad (1 \le a \le c_0).$$

For $a = 1$ we have $e_1 = 0$, since $s_0 < p$. Thus $S \equiv 0 \pmod{P'}$. Because $0 \le S < P'$, it follows that $S = 0$. Then the congruence above gives $e_a \equiv 0 \pmod{P'}$ for every a. Since $0 \le e_a < p \le P'$, we get $e_a = 0$ for all $a = 1, \dots, c_0$. In particular

$$s_0 c_0 < p.$$

Finally take $a = c_0$ and $z = C$, so that $a + pz = c$. We have $S = 0$, hence $s = s_0$, and $[sC]_{P'} = C$. Since $s_0 c_0 < p$, there is no carry from the lowest digit, so

$$[sc]_P = s_0 c_0 + pC.$$

The hypothesis says this is at most $c = c_0 + pC$. Therefore $s_0 c_0 \le c_0$. Because $c_0 > 0$, we conclude $s_0 = 1$. Together with $S = 0$, this gives $s = 1$, i.e. $s \equiv 1 \pmod{P}$. This completes the induction and the proof. $\square$

2.4 The zero-run lemma

Lemma 8 (Zero-run lemma)**.** *Let p be prime, let $m \geq 1$, and let M be coprime to p. Assume $p \nmid m+1$. Let $P = p^L$ be the least power of p satisfying $m < P$. If*

$$\binom{m+tM}{m} \equiv 0 \pmod{p} \qquad (t = 1, \ldots, m),$$

then

$$M \equiv -1 \pmod{P}.$$

Proof. Put

$$c = P - 1 - m.$$

Since $P/p \leq m < P$, we have

$$0 \leq c < P - P/p.$$

The assumption $p \nmid m+1$ gives $p \nmid c$, because $m + 1 = P - c$. Hence $c > 0$.

By Lemma 3, the congruence $\binom{m+tM}{m} \not\equiv 0 \pmod{p}$ is equivalent to the assertion that the base-p addition of m and tM has no carry. Since $m < P$, this is equivalent to

$$[tM]_P \in \mathcal{D}_c.$$

The hypothesis therefore says

$$[tM]_P \notin \mathcal{D}_c \qquad (t = 1, \ldots, m).$$

Let U be the inverse of M modulo P. Since $p \nmid c$, the element 1 belongs to $\mathcal{D}_c$. Thus U cannot lie in $\{1, \ldots, m\}$; also $U \not\equiv 0 \pmod{P}$. Hence there is a unique $s \in \{1, \ldots, c\}$ such that

$$U \equiv -s \pmod{P}.$$

For any $y \in \mathcal{D}_c \setminus \{0\}$, the residue $[Uy]_P$ cannot lie in $\{1, \ldots, m\}$; otherwise, if $[Uy]_P = t$ with $1 \leq t \leq m$, then $[tM]_P = y \in \mathcal{D}_c$, a contradiction. Therefore

$$[Uy]_P \in \{m+1, \ldots, P-1\} = \{P-c, \ldots, P-1\}.$$

Since $U \equiv -s \pmod{P}$, this is equivalent to

$$[sy]_P \in \{1, \ldots, c\} \qquad (y \in \mathcal{D}_c \setminus \{0\}).$$

By Theorem 7, and because $p \nmid c$, we get $s \equiv 1 \pmod{P}$. Since $1 \leq s \leq c < P$, this means $s = 1$. Hence $U \equiv -1 \pmod{P}$, and therefore $M \equiv -1 \pmod{P}$. □

2.5 The primary criterion

Lemma 9. *Let $n \geq 2$, let $p^a \| n$, and write*

$$A = p^a, \qquad n = Ab, \qquad p \nmid b.$$

Put

$$G_n = \gcd_{2 \leq q \leq n} \binom{q(n-1)}{n-1}.$$

Then

$$p \mid G_n \quad \Longleftrightarrow \quad b \leq A.$$

Proof. Let $K = n - 1 = Ab - 1$.

First suppose $p \nmid \binom{qK}{K}$. By Lemma 3, the lower a base-p digits of qK must dominate the lower a base-p digits of K. But

$$K = Ab - 1 \equiv -1 \pmod{A},$$

so those lower a digits of K are all $p - 1$. Hence the lower a digits of qK are also all $p - 1$, i.e.

$$qK \equiv -1 \pmod{A}.$$

As $K \equiv -1 \pmod{A}$, this gives $q \equiv 1 \pmod{A}$.

Thus any possible nonzero Lucas witness among $2 \le q \le n = Ab$ has the form

$$q = 1 + At, \qquad 1 \le t \le b - 1.$$

For such q,

$$qK = (1 + At)(Ab - 1) = A\big(b + t(Ab - 1)\big) - 1.$$

After the common lower a digits, Lucas' criterion compares the base-p digits of

$$b - 1 \quad \text{and} \quad b - 1 + t(Ab - 1).$$

Assume first that $b \le A$. For $1 \le t \le b - 1$,

$$b - 1 + t(Ab - 1) \equiv b - t - 1 \pmod{A},$$

and $0 \le b - t - 1 < b - 1 \le A - 1$. If the base-$p$ digits of $b - 1$ were all bounded by those of $b - 1 + t(Ab - 1)$, then in particular the lowest a digits would represent a number at least $b - 1$. But those lowest a digits represent $b - t - 1 < b - 1$. Therefore Lucas' criterion fails for every $t = 1, \dots, b - 1$. Hence every coefficient in the gcd is divisible by p, so $p \mid G_n$.

Conversely, assume $p \mid G_n$. Then for every $t = 1, \dots, b - 1$,

$$\binom{(1 + At)(Ab - 1)}{Ab - 1} \equiv 0 \pmod{p}.$$

Equivalently, after removing the common lower a digits as above,

$$\binom{(b - 1) + t(Ab - 1)}{b - 1} \equiv 0 \pmod{p} \qquad (t = 1, \dots, b - 1).$$

Set

$$m = b - 1, \qquad M = Ab - 1.$$

If $m = 0$, then $b = 1 \le A$ and there is nothing to prove. Otherwise $m \ge 1$, $p \nmid m + 1 = b$, and $p \nmid M$. Let $P = p^L$ be the least p-power with $m < P$. By Lemma 8,

$$M \equiv -1 \pmod{P}.$$

Thus

$$Ab - 1 \equiv -1 \pmod{P},$$

so $P \mid Ab$. Since $p \nmid b$, this implies $P \mid A$. Finally $b = m + 1 \le P \le A$. Therefore $b \le A$. □

2.6 Completion of the proof

Proof of Theorem 1. Let

$$D(k) = \gcd_{2 \le q \le k+1} \binom{qk}{k}, \qquad n = k + 1.$$

The gcd is a positive integer. By Lemma 4, every prime divisor of $D(k)$ divides n.

For a prime divisor p of n, write $p^a \| n$, put $A = p^a$, and write $n = Ab$. Lemma 9, with $G_n = D(k)$, gives

$$p \mid D(k) \iff b \le A \iff \frac{n}{p^a} \le p^a.$$

Consequently,

$$D(k) = 1 \iff \frac{n}{p^a} > p^a \text{ for every exact prime-power component } p^a \| n.$$

It remains only to observe that this family of inequalities is equivalent to the single inequality involving the largest exact prime-power component $P = \operatorname{ppart}(n)$. If $n/P > P$ and $A = p^a \| n$ is any other exact component, then $A \le P$, and hence

$$\frac{n}{A} \ge \frac{n}{P} > P \ge A.$$

Conversely, if the inequality fails for some exact component A, then $n/A \le A \le P$, so

$$\frac{n}{P} \le \frac{n}{A} \le P.$$

Thus all component inequalities hold if and only if $n/P > P$. This proves the theorem. □

3 The Lean formalization

The theorem has also been formalized in Lean in `FormalConjectures/OEIS/80170.lean` in the `formal-conjectures` repository, under the statement `OeisA80170.gcdCondition_iff_primePowerCondition`. The file imports `FormalConjectures.Util.ProblemImports` and `Mathlib.Algebra.IsPrimePow`. This section describes the organization of that formal proof. It is included because the Lean development is a substantial part of the result: it fixes the exact formal meaning of the OEIS statement, builds the auxiliary infrastructure used in the proof, and checks the final assembly in the Lean kernel. The natural-language proof in this article and the Lean formalization were both generated by the MechMath Agent Team [8].

3.1 Benchmark statement and compatibility layer

The Formal Conjectures file starts from the benchmark predicates rather than from the notation used in Theorem 1. In the displays below, we use line-broken ASCII renderings of the Lean signatures; the linked source contains the exact Unicode syntax. The benchmark-facing statement is:

```
def GCDCondition (k : Nat) : Prop :=
  (Finset.range k).gcd
    (fun i => Nat.choose ((i + 2) * k) k) = 1

def PrimePowerCondition (k : Nat) : Prop :=
  let P := ((Nat.divisors k).filter IsPrimePow).max.getD 0
  k / P > P

@[category research solved, AMS 11]
theorem gcdCondition_iff_primePowerCondition
    (k : Nat) (hk : 2 <= k) :
    GCDCondition k <-> PrimePowerCondition (k + 1)
```

Thus the formal theorem proves the equivalence between the gcd predicate at k and the prime-power predicate at $k+1$. The informal proof, however, is stated in terms of

$$D(k) = \gcd_{2\le q\le k+1} \binom{qk}{k} \qquad \text{and} \qquad n = k+1.$$

The formal proof therefore introduces the paper-level definitions:

```
def D (k : Nat) : Nat :=
  (Finset.Icc 2 (k + 1)).gcd
    fun q => Nat.choose (q * k) k

def ppart (n : Nat) : Nat :=
  n.primeFactors.sup
    fun p => p ^ n.factorization p

def digitBox (p L c : Nat) : Finset Nat :=
  (Finset.range (p ^ L)).filter
    fun y => forall i, i < L ->
      y / p ^ i % p <= c / p ^ i % p
```

The main theorem is obtained only after two compatibility bridges. The first bridge identifies the benchmark gcd over `Finset.range k` with the paper's gcd over `Finset.Icc 2 (k + 1)`. The second bridge identifies the benchmark's maximum over prime-power divisors with the exact component $\mathrm{ppart}(n) = \max_{p^a \| n} p^a$.

```
theorem gcdCondition_iff_D_eq_one (k : Nat) :
    GCDCondition k <-> D k = 1

theorem max_primePow_divisor_eq_ppart
    (n : Nat) (hn : 2 <= n) :
    ((Nat.divisors n).filter IsPrimePow).max.getD 0
```

```
      = ppart n

theorem primePowerCondition_iff_ppart
    (n : Nat) (hn : 2 <= n) :
    PrimePowerCondition n <-> n / ppart n > ppart n
```

These statements are mostly bookkeeping from a mathematical point of view, but they are essential in Lean: the finite indexing set, the shift from k to $k+1$, the exact exponent `n.factorization p`, and the maximum over prime-power divisors all have to be expressed as the same objects before the final theorem can call the proof of Theorem 1.

3.2 Formalizing the proof infrastructure

The formalization follows the proof section in two main preparatory blocks. The first block proves the finite-difference reduction. Its general tool is `newton_interpolation`, a Newton interpolation formula over $\mathbb{Q}$ for forward differences. It is then specialized to the polynomial attached to $\binom{xk}{k}$, yielding the divisibility theorem used in Lemma 4.

```
theorem newton_interpolation
    (f : Polynomial Rat) (d : Nat) (hf : f.natDegree <= d)
    (a : Int) (x : Rat) :
    -- Newton expansion of f.eval x by forward differences at a

theorem prime_dvd_succ_of_dvd_D
    (k p : Nat) (hk : 1 <= k)
    (hp : p.Prime) (hdiv : p | D k) :
    p | k + 1
```

The second block formalizes the Lucas-theoretic criterion. The paper states this in base-p digit language; the Lean statement uses the explicit digit formula $N/p^i \bmod p$ at every position i.

```
theorem lucas_nonvanishing
    (p : Nat) (hp : p.Prime) (N K : Nat) :
    not (p | Nat.choose N K) <->
      forall i, K / p ^ i % p <= N / p ^ i % p
```

The proof calls Mathlib's formalized Lucas theorem inside this lemma. After `lucas_nonvanishing` is available, congruence questions about binomial coefficients can be converted into digit inequalities and then into membership statements for `digitBox`.

3.3 Digit boxes and the zero-run lemma

The longest local development is the digit-box part. The final stabilizer statement in `80170.lean` is a multiplicative stabilizer theorem: under the hypotheses of Theorem 7, multiplication by s preserves the nonzero part of the digit box if and only if s is congruent to 1 modulo the indicated power of p.

```
theorem digitBox_stabilizer
    (p L c s : Nat) (hp : p.Prime) (hL : 1 <= L)
    (hc0 : 0 < c) (hc : c < p ^ L - p ^ L / p)
    (hs : Nat.Coprime s p) :
    (forall y, y in digitBox p L c -> y != 0 ->
       1 <= s * y % p ^ L
       and s * y % p ^ L <= c)
    <->
    s == 1 [MOD p ^ (L - c.factorization p)]
```

The proof of this theorem is decomposed into smaller finite-set and digit lemmas. The basic membership and digit estimates are handled by `mem_digitBox`, `mem_digitBox_succ`, and `le_of_mem_digitBox`. The gap

and translation arguments are isolated as `gap_bound` and `no_nonzero_translation`; these lemmas supply the finite combinatorial core needed by `digitBox_stabilizer`.

The zero-run lemma then turns a run of binomial divisibilities into a modular conclusion. Its formal statement is:

```
theorem zero_run
    (p m M L : Nat) (hp : p.Prime) (hm : 1 <= m)
    (hM : Nat.Coprime M p) (hm1 : not (p | m + 1))
    (hLm : p ^ (L - 1) <= m) (hmL : m < p ^ L)
    (h : forall t, t in Finset.Icc 1 m ->
         p | Nat.choose (m + t * M) m) :
    (M : ZMod (p ^ L)) = -1
```

The conversions leading to `zero_run` are also named explicitly in the file:

```
lemma noCarry_iff_mem_box
lemma not_dvd_choose_iff_no_carry
lemma not_dvd_choose_iff_mem_box
```

These lemmas are where the informal phrase "no carry in base p" becomes a statement about finite-set membership in `digitBox`. This extra layer is one of the reasons the formal proof is much longer than the paper proof: every use of a residue modulo p^L, every digit inequality, and every conversion between a binomial congruence and a carry condition has to be stated and typed explicitly.

3.4 Primary criterion and final assembly

The local p-primary criterion is the central formal theorem connecting the digit-box work back to the gcd:

```
theorem primary_criterion
    (n p b : Nat) (hn : 2 <= n) (hp : p.Prime)
    (ha : 1 <= n.factorization p)
    (hb : n = p ^ n.factorization p * b)
    (hpb : not (p | b)) :
    p | D (n - 1) <-> b <= p ^ n.factorization p
```

The exponent in this statement is not a separate variable. It is the exact exponent `n.factorization p`, so the hypotheses encode $n = p^a b$, $a \geq 1$, and $p \nmid b$ directly in Lean. The proof uses `lucas_nonvanishing` for the easy direction and `zero_run` for the converse direction.

The paper theorem is then formalized as `a080170`:

```
theorem a080170 (k : Nat) (hk : 2 <= k) :
    D k = 1 <->
      (k + 1) / ppart (k + 1) > ppart (k + 1)
```

Finally, the benchmark theorem is a short assembly step:

```
@[category research solved, AMS 11]
theorem gcdCondition_iff_primePowerCondition
    (k : Nat) (hk : 2 <= k) :
    GCDCondition k <-> PrimePowerCondition (k + 1) := by
  rw [gcdCondition_iff_D_eq_one k, a080170 k hk,
    primePowerCondition_iff_ppart (k + 1) (by omega)]
```

This last block clearly shows the role of the compatibility layer: after the benchmark predicates are rewritten into $D(k) = 1$ and $(k+1)/\operatorname{ppart}(k+1) > \operatorname{ppart}(k+1)$, the kernel-checked proof is exactly the formal theorem `a080170`. The pinned source file contains no `sorry`, and the final theorem is marked as a solved research benchmark in the Formal Conjectures development.

3.5 Remarks and availability

The complete Lean proof code is available at the pinned commit URL

`https://github.com/guodk/formal-conjectures/blob/0720658844d76a50d48e4baa152eef14d4462907/FormalConjectures/OEIS/80170.lean#L1823`.

The corresponding pull request to the Formal Conjectures repository is

`https://github.com/google-deepmind/formal-conjectures/pull/4253`.

The pinned code link is used to identify the exact artifact discussed in this paper, while the pull request records the review context in the upstream project.

4 Conclusions

In this paper, a binomial gcd criterion is proven. The criterion is listed as OEIS A080170 and appears as conjecture (17) in Ralf Stephan's list of OEIS conjectures. Furthermore, the proof has been formalized in Lean, ensuring its correctness. The problem under consideration is also a member of `FC100OpenSet1`, a distinguished subset of 100 open research problems in the Formal Conjectures benchmark. Our Lean proof has been accepted by the Formal Conjectures project.

References

[1] Edgar H. Brown, Jr. and Franklin P. Peterson. Relations among characteristic classes. II. *Annals of Mathematics*, 81(2):356–363, 1965. `https://doi.org/10.2307/1970620`.

[2] Moritz Firsching, Paul Lezeau, Simone Mercuri, Márton Zsombor Horváth, Yaël Dillies, Christian Sönne, Eric Wieser, Fred Zhang, Thomas Hubert, Blaise Agüera y Arcas, and Pushmeet Kohli. Formal conjectures: An open and evolving benchmark for verified discovery in mathematics. `https://arxiv.org/abs/2605.13171`, 2026.

[3] Shaofang Hong. The greatest common divisor of certain binomial coefficients. *Comptes Rendus Mathematique*, 354(8):756–761, 2016. `https://doi.org/10.1016/j.crma.2016.06.001`.

[4] H. Joris, C. Oestreicher, and J. Steinig. The greatest common divisor of certain sets of binomial coefficients. *Journal of Number Theory*, 21(1):101–119, 1985. `https://doi.org/10.1016/0022-314X(85)90013-7`.

[5] Zhi Lü and Taras Panov. On toric generators in the unitary and special unitary bordism rings. *Algebraic & Geometric Topology*, 16(5):2865–2893, 2016. `https://doi.org/10.2140/agt.2016.16.2865`.

[6] Édouard Lucas. Sur les congruences des nombres euleriens et des coefficients differentiels des fonctions trigonometriques, suivant un module premier. *Bulletin de la Société Mathématique de France*, 6: 49–54, 1878.

[7] Cameron McTague. On the greatest common divisor of binomial coefficients $\binom{n}{q}, \binom{n}{2q}, \binom{n}{3q}, \ldots$. *The American Mathematical Monthly*, 124(4):353–356, 2017. Also available as `https://arxiv.org/abs/1510.06696`.

[8] MechMath Agent Team Authors. MechMath Agent Team. `https://mechmath.github.io/`, 2026.

[9] Romeo Meštrović. Lucas' theorem: its generalizations, extensions and applications (1878–2014). `https://arxiv.org/abs/1409.3820`, 2014.

[10] OEIS Foundation Inc. The on-line encyclopedia of integer sequences, A080170. Published electronically at `https://oeis.org/A080170`, 2026.

[11] B. Ram. Common factors of $n!/(m!(n-m)!)$ $(m = 1, 2, \ldots, n-1)$. *Journal of the Indian Mathematical Club*, 1:39–43, 1909.

[12] N. J. A. Sloane. The on-line encyclopedia of integer sequences. *Notices of the American Mathematical Society*, 50:912–915, 2003. Also available as https://arxiv.org/abs/math/0312448.

[13] Ralf Stephan. Prove or disprove. 100 conjectures from the OEIS. https://arxiv.org/abs/math/0409509, 2004.

[14] The Formal Conjectures Authors. The formal conjectures repository. https://github.com/google-deepmind/formal-conjectures, 2025.