

# **Proofs and the tools they use**

**Heather Macbeth**

**Imperial College London / Fordham University**

**Autoformalization for the Working Mathematician**

**ICERM, 27 April 2025**

# Disclaimer

In this talk, “AI-generated proofs” are Lean proofs taken from recent articles/technical reports in the AI literature.

It’s commendable that these authors provide so many examples of the Lean proofs their systems construct — many authors don’t.

But I quote mostly to criticize! So I will not “name and shame” the systems I quote.

# The coming dystopia

Maybe, someday soon, mathematicians will have easy access to systems which can prove some lemmas at the push of a button.

What if these proofs are terrible?

Will such technologies **impede** our understanding of mathematics?

Alex Kontorovich, *AI to Assist Mathematical Reasoning* workshop, National Academy of Science, 2023:

**Holy Grail:** AI solves the Riemann hypothesis.

**Nightmare:** AI solves the Riemann hypothesis in 1,000,000 lines of code, incomprehensible to humans. Mathematicians spend the rest of their lives trying to understand what it means.

# AMC 12b, 2021, problem 4

Ms. Blackwell gives an exam to two classes. The mean of the scores of the students in the morning class is 84, and the afternoon class's mean score is 70. The ratio of the number of students in the morning class to the number of students in the afternoon class is  $\frac{3}{4}$ . What is the mean of the scores of all the students?

- (A) 74    (B) 75    (C) 76    (D) 77    (E) 78

## AI-generated solution

```
have h2 : (m : ℝ) / (a : ℝ) = (3 : ℝ) / 4 := by
  exact_mod_cast h1
have h3 : (a : ℝ) ≠ 0 := by
  have ha : (a : ℝ) > (0 : ℝ) := by
    exact_mod_cast h0.right
  linarith
have h4 : (m : ℝ) = (3 : ℝ) / 4 * (a : ℝ) := by
  have h5 : (a : ℝ) ≠ 0 := h3
  field_simp at h2 ⊢
  linarith
have h5 : (↑m + ↑a : ℝ) ≠ 0 := by
  have ha : (a : ℝ) > 0 := by exact_mod_cast h0.right
  have hm : (m : ℝ) ≥ 0 := by
    exact_mod_cast (show (m : ℝ) ≥ (0 : ℝ) by norm_num)
  nlinarith [h4]
have h6 : (84 : ℝ) * ↑m + 70 * ↑a = 76 * (↑m + ↑a) := by
  have h7 : (m : ℝ) = (3 : ℝ) / 4 * (a : ℝ) := h4
  linarith
have h8 : (84 * (m : ℝ) + 70 * (a : ℝ)) / ((m : ℝ) + (a : ℝ)) = (76 : ℝ) := by
  have h9 : (84 : ℝ) * ↑m + 70 * ↑a = 76 * (↑m + ↑a) := h6
  have h10 : (↑m + ↑a : ℝ) ≠ 0 := by
    exact_mod_cast h5
  field_simp at *
  linarith
exact h8
```

## Cleaned up by hand by me

```
cases h0
field_simp at *
linarith
```

“After clearing denominators, this is a linear equation.”

# AIME, 1987, problem 5

Find  $3x^2y^2$  if  $x$  and  $y$  are integers such that  $y^2 + 3x^2y^2 = 30x^2 + 517$ .

## AI-generated solution

```
have h1 : 3 * (x ^ 2 * y ^ 2) = 30 * x ^ 2 + 517 - y^2 := by linarith
have h2 : y^2 ≤ 517 := by
  nlinarith [sq_nonneg (x * y), sq_nonneg (x * y - 17), sq_nonneg (y - 17),
    sq_nonneg (x - 2), sq_nonneg (x + 2)]
have h3 : y ≤ 22 := by nlinarith [h2]
have h4 : y ≥ -22 := by
  nlinarith [sq_nonneg (y + 22), sq_nonneg (y + 21), sq_nonneg (y + 20),
    sq_nonneg (y + 19), sq_nonneg (y + 18), sq_nonneg (y + 17),
    sq_nonneg (y + 16), sq_nonneg (y + 15), sq_nonneg (y + 14),
    sq_nonneg (y + 13), sq_nonneg (y + 12), sq_nonneg (y + 11),
    sq_nonneg (y + 10), sq_nonneg (y + 9), sq_nonneg (y + 8),
    sq_nonneg (y + 7), sq_nonneg (y + 6), sq_nonneg (y + 5),
    sq_nonneg (y + 4), sq_nonneg (y + 3), sq_nonneg (y + 2),
    sq_nonneg (y + 1), sq_nonneg (y)]
interval_cases y <|>
try {
  simp_all
  <|>
  (
    ring_nf
    <|>
    omega
  )
}
```

## Cleaned up by hand by me

```
have h2 : y^2 ≤ 517 := by nlinarith
have h3 : y ≤ 22 := by nlinarith
have h4 : y ≥ -22 := by nlinarith
interval_cases y <|> omega
```

“The equation implies that  $y^2 \leq 517$ .

“So  $-22 \leq y \leq 22$ .

“Now case-bash.”

# IMO, 1962, problem 2

Determine all real numbers  $x$  which satisfy the inequality:

$$\sqrt{\sqrt{3-x} - \sqrt{x+1}} > \frac{1}{2}$$

Mis-stated in the **miniF2F** benchmark as,

Determine all real numbers  $x$  which satisfy the inequality:

$$\sqrt{3-x} - \sqrt{x+1} > \frac{1}{2}$$

Furthermore statement is in **easy mode**: we reveal that the solution is

$\left[-1, 1 - \frac{\sqrt{31}}{8}\right)$  before asking for a proof.

# (A variation of) IMO, 1962, problem 2

Prove that if a real number  $x$  satisfies the inequality  $\sqrt{3-x} - \sqrt{x+1} > \frac{1}{2}$ , then  $x \in \left[-1, 1 - \frac{\sqrt{31}}{8}\right)$ .

## AI-generated solution

```
constructor
· linarith [h1]
· have h3 : Real.sqrt (3 - x) - Real.sqrt (x + 1) > 1 / 2 := by linarith
  have h4 : Real.sqrt (3 - x) > Real.sqrt (x + 1) + 1 / 2 := by linarith
  have h5 : Real.sqrt (x + 1) ≥ 0 := Real.sqrt_nonneg (x + 1)
  have h6 : Real.sqrt (3 - x) ≥ 0 := Real.sqrt_nonneg (3 - x)
  have h7 : Real.sqrt (3 - x) > 0 := by linarith [h6, h5]
  have h8 : (Real.sqrt (3 - x)) ^ 2 > (Real.sqrt (x + 1) + 1 / 2) ^ 2 := by
    apply sq_lt_sq'
    all_goals linarith [h4, h5, h7]
  have h9 : (Real.sqrt (3 - x)) ^ 2 = 3 - x := by
    rw [Real.sq_sqrt]
    linarith
  have h10 :
    (Real.sqrt (x + 1) + 1 / 2) ^ 2 = (Real.sqrt (x + 1)) ^ 2 + Real.sqrt (x + 1) + 1 / 4 := by
      ring_nf
      <=> simp [Real.sqrt_nonneg]
  rw [h9, h10] at h8
  have h11 : (Real.sqrt (x + 1)) ^ 2 = x + 1 := by
    rw [Real.sq_sqrt]
    linarith
  have h12 : Real.sqrt (x + 1) < 7 / 4 - 2 * x := by nlinarith [h8, h11, Real.sqrt_nonneg (x + 1)]
  have h13 : x < 1 - Real.sqrt 31 / 8 := by
    have h14 : Real.sqrt (x + 1) + 2 * x < 7 / 4 := by linarith [h12]
    have h15 : Real.sqrt 31 ^ 2 = 31 := by
      rw [Real.sq_sqrt]
      norm_num
    nlinarith [h14, Real.sq_sqrt (show 0 ≤ x + 1 by linarith),
      Real.sq_sqrt (show 0 ≤ 3 - x by linarith), Real.sqrt_nonneg (3 - x),
      Real.sqrt_nonneg (x + 1), Real.sq_sqrt (show 0 ≤ (31 : ℝ) by norm_num),
      Real.sq_sqrt (show 0 ≤ (8 : ℝ) by norm_num)]
  linarith [h13]
```

## Cleaned up by hand by me

```
constructor
· linarith
have h5 : Real.sqrt (x + 1) ≥ 0 := Real.sqrt_nonneg (x + 1)
have := Real.sq_sqrt (x := x + 1) (by linarith)
have := Real.sq_sqrt (x := 3 - x) (by linarith)
have := Real.sq_sqrt (x := 31) (by linarith)
have h12 : Real.sqrt (x + 1) < 7 / 4 - 2 * x := by nlinarith
nlinarith
```

“The key intermediate observation is that  $\sqrt{x+1} < \frac{7}{4} - 2x$ .”

# AIME II, 2001, problem 3

Given that

$$x_1 = 211,$$

$$x_2 = 375,$$

$$x_3 = 420,$$

$$x_4 = 523, \text{ and}$$

$$x_n = x_{n-1} - x_{n-2} + x_{n-3} - x_{n-4} \text{ when } n \geq 5,$$

find the value of  $x_{531} + x_{753} + x_{975}$ .

# AI-generated solution

```
-- First, compute  $x_5$  using the recurrence relation
have h5 : x 5 = 267 := by
  have h5 := h6 5 (by norm_num)
  simp_all
-- Next, observe that the sequence repeats every 10 terms
-- We will prove that for all  $n \geq 1$ ,  $x (n + 10) = x n$ 
have period_10 :  $\forall n \geq 1, x (n + 10) = x n$  := by
  intro n hn
  induction n with
  | zero =>
    simp_all
  | succ n ih =>
    simp_all [Nat.succ_eq_add_one]
    linarith [h6 10 (by linarith), h6 11 (by linarith), h6 12 (by linarith),
             h6 13 (by linarith), h6 14 (by linarith), h6 15 (by linarith),
             h6 16 (by linarith), h6 17 (by linarith), h6 18 (by linarith),
             h6 19 (by linarith)]
-- Compute  $x_{531}$ 
have x531_eq : x 531 = x 1 := by
  simp [h1, h2, h3, h4, h5, period_10] using (show 531 % 10 = 1 by norm_num)
-- Compute  $x_{753}$ 
have x753_eq : x 753 = x 3 := by
  rw [show 753 = 10 * 75 + 3 by norm_num]
  simp [period_10, h3, h6, h5, h2, h1]
-- Compute  $x_{975}$ 
have x975_eq : x 975 = x 5 := by
  norm_num [period_10, x531_eq, x753_eq]
-- Sum the values
have sum_eq : x 531 + x 753 + x 975 = x 1 + x 3 + x 5 := by
  norm_num [h1, h2, h3, h4, h5, x531_eq, x753_eq, x975_eq]
-- Using the periodicity condition, we simplify the sum to  $x_1 + x_3 + x_5$ 
linarith [h1, h3, h5]
```

# Cleaned up by hand by me

```
-- First, compute  $x_5$  using the recurrence relation
have h5 : x 5 = 267 := by
  simp_all
-- Next, observe that the sequence repeats every 10 terms
-- We will prove that for all  $n \geq 1$ ,  $x (n + 10) = x n$ 
have period_10 :  $\forall n \geq 1, x (n + 10) = x n$  := by
  intro n hn
  induction n with
  | zero =>
    simp_all
  | succ n ih =>
    simp_all
    linarith
-- Compute  $x_{531}$ 
have x531_eq : x 531 = x 1 := by simp [period_10]
-- Compute  $x_{753}$ 
have x753_eq : x 753 = x 3 := by
  simp [period_10, h3]
-- Compute  $x_{975}$ 
have x975_eq : x 975 = x 5 := by
  norm_num [period_10, x531_eq, x753_eq]
-- Sum the values
linarith
```

# Much of the AI-generated solution is irrelevant

```
-- First, compute x5 using the recurrence relation
have h5 : x 5 = 267 := by
  have h5 := h6 5 (by norm_num)
  simp_all
-- Next, observe that the sequence repeats every 10 terms
-- We will prove that for all n ≥ 1, x (n + 10) = x n
have period_10 : ∀ n ≥ 1, x (n + 10) = x n := by
  intro n hn
  induction n with
  | zero =>
    simp_all
  | succ n ih =>
    simp_all [Nat.succ_eq_add_one]
    linarith [h6 10 (by linarith), h6 11 (by linarith), h6 12 (by linarith),
             h6 13 (by linarith), h6 14 (by linarith), h6 15 (by linarith),
             h6 16 (by linarith), h6 17 (by linarith), h6 18 (by linarith),
             h6 19 (by linarith)]
  simp_all
  linarith
-- Compute x531
have x531_eq : x 531 = x 1 := by
  simpa [h1, h2, h3, h4, h5, period_10] using (show 531 % 10 = 1 by norm_num)
have x531_eq : x 531 = x 1 := by simp [period_10]
-- Compute x753
have x753_eq : x 753 = x 3 := by
  rw [show 753 = 10 * 75 + 3 by norm_num]
  simp [period_10, h3, h6, h5, h2, h1]
  simp [period_10, h3]
-- Compute x975
have x975_eq : x 975 = x 5 := by
  norm_num [period_10, x531_eq, x753_eq]
-- Sum the values
have sum_eq : x 531 + x 753 + x 975 = x 1 + x 3 + x 5 := by
  norm_num [h1, h2, h3, h4, h5, x531_eq, x753_eq, x975_eq]
-- Using the periodicity condition, we simplify the sum to x1 + x3 + x5
linarith [h1, h3, h5]
linarith
```

This model does not understand the full strength of the automation it is using (`linarith`, `simp`).

This is a common theme among all the previous examples.

# Automation style affects understanding

The AI systems that I quoted just now are trained only to produce proofs — they are not rewarded for producing short or “good” proofs.

Maybe, with a little bit of effort, AI systems could trim the obvious cruft from their proofs.

**Relevant literature:** *ImProver: Agent-Based Automated Proof Optimization*. R. Ahuja, J. Avigad, P. Tetali, S. Welleck, *ICLR 2025*

But even after streamlining, there are still many ways to prove a statement.

Some are more comprehensible to humans than others. What are good patterns; is it possible to reward AI for producing them?

# AIME, 1990, problem 4

Find the positive solution to

$$\frac{1}{x^2 - 10x - 29} + \frac{1}{x^2 - 10x - 45} - \frac{2}{x^2 - 10x - 69} = 0$$

**Easy mode:** system is told that the solution is 13.

## AI-generated solution

```
have h5 : x^2 - 10 * x - 29 ≠ 0 := h1
have h6 : x^2 - 10 * x - 45 ≠ 0 := h2
have h7 : x^2 - 10 * x - 69 ≠ 0 := h3
have h_eq : (x - 13) * (x + 3) = 0 := by
  field_simp [(show x^2 - 10 * x - 29 ≠ 0 by assumption),
    (show x^2 - 10 * x - 45 ≠ 0 by assumption),
    (show x^2 - 10 * x - 69 ≠ 0 by assumption)] at h4
  ring_nf at h4
  nlinarith [sq_nonneg (x - 13), sq_nonneg (x + 3),
    sq_nonneg (x^2 - 10 * x - 29), sq_nonneg (x^2 - 10 * x - 45),
    sq_nonneg (x^2 - 10 * x - 69)]
cases' (mul_eq_zero.mp h_eq) with h13 h3
· -- x - 13 = 0
  linarith
· -- x + 3 = 0, so x = -3, contradicting 0 < x
  linarith
```

## Alternative solution by me

```
field_simp at h4
have hx : x + 3 ≠ 0 := by linarith
apply mul_left_cancel0 hx
polyrith
```

“Clear denominators.

“Then do a Gröbner basis calculation to show that  $(x - 13)(x + 3)$  is in the ideal generated by the equations we know.”

# AI-generated solution

```
have h2 : Real.cos x ≠ 0 := by
  by_contra h
  have h3 : 1 / Real.cos x = 0 := by
    field_simp [h]
  have h4 : Real.tan x = 0 := by
    rw [Real.tan_eq_sin_div_cos]
    simp [h]
  have h5 : (22 : ℝ) / 7 = 0 := by linarith [h, h3, h4]
  norm_num at h5

have h_tan : Real.tan x = Real.sin x / Real.cos x := by
  rw [Real.tan_eq_sin_div_cos]

have h3 : Real.sin x ^ 2 + Real.cos x ^ 2 = 1 := Real.sin_sq_add_cos_sq x

have h4 : 1 / Real.cos x + Real.tan x = 22 / 7 := h
  rw [h_tan] at h4
have h5 : 1 / Real.cos x + Real.sin x / Real.cos x = 22 / 7 := h4
have h6 : (1 + Real.sin x) / Real.cos x = 22 / 7 := by
  have h7 : Real.cos x ≠ 0 := h2
  field_simp [h7] at h5
  nlinarith

have h7 : 7 * (1 + Real.sin x) = 22 * Real.cos x := by
  have h8 : Real.cos x ≠ 0 := h2
  field_simp at h6
  nlinarith

have h_sin : Real.sin x = (22 * Real.cos x - 7) / 7 := by linarith

have h8 : Real.sin x ^ 2 + Real.cos x ^ 2 = 1 := Real.sin_sq_add_cos_sq x
have h9 : Real.sin x = (22 * Real.cos x - 7) / 7 := h_sin

have h_cos_sq : Real.cos x ^ 2 = (308 / 533) ^ 2 := by
  have h10 : Real.cos x ≠ 0 := h2
  have h11 : Real.sin x = (22 * Real.cos x - 7) / 7 := h_sin
  rw [h11] at h3
  have h12 : ((22 * Real.cos x - 7) / 7) ^ 2 + Real.cos x ^ 2 = 1 := by
    linarith
  have h13 : (22 * Real.cos x - 7) ^ 2 + (7 * Real.cos x) ^ 2 = 49 := by
    nlinarith
  have h14 : (Real.cos x) * (533 * Real.cos x - 308) = 0 := by
    ring_nf at h13
    linarith
  cases' (mul_eq_zero.mp h14) with h_cos h15
  · -- Real.cos x = 0
    exfalso
    exact h10 (by linarith)
  · -- 533 * Real.cos x - 308 = 0
    have h16 : Real.cos x = 308 / 533 := by linarith
    nlinarith [h_cos]
have h_cos : Real.cos x = 308 / 533 := by nlinarith [h_cos_sq]

have h_sin' : Real.sin x = 435 / 533 := by
  have h9 : Real.sin x = (22 * Real.cos x - 7) / 7 := h_sin
  rw [h9, h_cos]
  norm_num

have h_csc_cot : (1 / Real.sin x : ℝ) + (1 / Real.tan x : ℝ)
  = (29 / 15 : ℝ) := by
  have h10 : Real.sin x ≠ 0 := by
    by_contra h
    have h11 : Real.sin x = 0 := by linarith
    rw [h11] at h3
    have h12 : Real.cos x ^ 2 = 1 := by nlinarith
    have h13 : Real.cos x = 1 ∨ Real.cos x = -1 := by
      have h14 : Real.cos x ^ 2 - 1 = 0 := by linarith
      have h15 : (Real.cos x - 1) * (Real.cos x + 1) = 0 := by
        ring_nf at h14
        linarith
      cases' (mul_eq_zero.mp h15) with
      | inl h16 => left; linarith
      | inr h17 => right; linarith
    cases h13 with
    | inl h14 =>
      have h15 : Real.cos x = 308 / 533 := h_cos
      linarith
    | inr h14 =>
      have h15 : Real.cos x = 308 / 533 := h_cos
      linarith
  have h11 : Real.tan x ≠ 0 := by
    by_contra h
    have h12 : Real.sin x = 0 := by
      rw [Real.tan_eq_sin_div_cos] at h
      have h13 : Real.cos x ≠ 0 := h2
      field_simp [h13] at h
      linarith
    contradiction
  have h12 : (1 / Real.sin x : ℝ) + (1 / Real.tan x : ℝ)
    = (29 / 15 : ℝ) := by
    have h13 : Real.tan x = Real.sin x / Real.cos x :=
      Real.tan_eq_sin_div_cos x
    rw [h13]
    have h14 : Real.sin x ≠ 0 := h10
    have h15 : Real.cos x ≠ 0 := h2
    field_simp [h14, h15, h_sin', h_cos]
    norm_num
  exact h12
have h_m : (m : ℝ) = (29 / 15 : ℝ) := by
have h12 : (1 / Real.sin x : ℝ) + (1 / Real.tan x : ℝ) = (29 / 15 : ℝ) :=
  h_csc_cot
have h13 : (1 / Real.sin x : ℝ) + (1 / Real.tan x : ℝ) = (m : ℝ) := by
  exact_mod_cast h
  rw [h13] at h12
  linarith
have h14 : m = (29 / 15 : ℝ) := by
have h15 : (m : ℝ) = (29 / 15 : ℝ) := h_m
have h16 : (m : ℝ) = (29 / 15 : ℚ) := by
  norm_num at h15
  exact_mod_cast h15
  exact_mod_cast h16
  rw [h14]
norm_num
```

# AIME, 1991, problem 9

Suppose that  $\sec x + \tan x = \frac{22}{7}$  and that  $\csc x + \cot x = \frac{m}{n}$ , where  $\frac{m}{n}$  is in lowest terms. Find  $m + n$ .

**Easy mode:** system is told that the solution is 44

## Alternative solution by me

```
have h_tan : Real.tan x = Real.sin x / Real.cos x := by
  rw [Real.tan_eq_sin_div_cos]
```

```
have h2 : Real.cos x ≠ 0 := by
  by_contra h
  simp [h] at *
  linarith
field_simp at h0 h_tan
```

```
have h3 : Real.sin x ^ 2 + Real.cos x ^ 2 = 1 := Real.sin_sq_add_cos_sq x
```

```
have : Real.tan x ≠ 0 := by
  by_contra h
  simp only [h] at *
  nlinarith
have : Real.sin x ≠ 0 := by
  by_contra h
  nlinarith
field_simp at h1
```

```
have h_m : 15 * (m : ℝ) = 29 := by polyrith
norm_cast at h_m
have h14 : m = 29 / 15 := by linarith
rw [h14]
norm_num
```

“Note that  $\cos x$ ,  $\tan x$  and  $\sin x$  must all be nonzero.

“Clear denominators.

“Then do a Gröbner basis calculation to show that  $15\frac{m}{n} - 29$  is in the ideal generated by the equations we know.”

# IMO, 1983, problem 6

Let  $a$ ,  $b$  and  $c$  be the lengths of the sides of a triangle. Prove that

$$a^2b(a - b) + b^2c(b - c) + c^2a(c - a) \geq 0.$$

Determine when equality occurs.

## AI-generated solution

```
have h4 : a + b - c > 0 := by linarith
have h5 : b + c - a > 0 := by linarith
have h6 : c + a - b > 0 := by linarith

nlinarith [sq_nonneg (a - b), sq_nonneg (b - c), sq_nonneg (c - a),
  h4, h5, h6, mul_nonneg (show 0 ≤ a + b - c by linarith) (sq_nonneg (a - b)),
  mul_nonneg (show 0 ≤ b + c - a by linarith) (sq_nonneg (b - c)),
  mul_nonneg (show 0 ≤ c + a - b by linarith) (sq_nonneg (c - a))]
```

## Alternative solution by me

```
have h4 : a + b - c > 0 := by linarith
have h5 : b + c - a > 0 := by linarith
have h6 : c + a - b > 0 := by linarith
have : 0 ≤ (a + b - c) * (c + a - b) * (c - a) ^ 2
  + (b + c - a) * (a + b - c) * (a - b) ^ 2
  + (c + a - b) * (b + c - a) * (b - c) ^ 2 := by positivity
linarith
```

“Clearly the expression

$$(a + b - c)(c + a - b)(c - a)^2 + (b + c - a)(a + b - c)(a - b)^2 + (c + a - b)(b + c - a)(b - c)^2$$

is nonnegative. Expand and halve it.”

# Why aren't AI systems using automation in this way?

These proofs are in a very specific style: they consist of rather few steps, each of which is a (potentially computationally-expensive) call to a standard algorithm with a precise scope.

For humans, this style is hard to *write* but easy to *read*.

**Relevant literature:** *Algorithm and abstraction in formal mathematics*. H. Macbeth, ICMS 2024

I haven't seen many AI-generated Lean proofs which are in this style, or even which make much use of the key tactics for this style.

- Maybe because some of these tactics are fairly new?
  - Gröbner basis tactic (`polyrith`), Bhatia-Lewis 2022
  - `positivity` tactic, Carneiro-Macbeth 2022
- For `polyrith`, maybe because it is “self-replacing”, so it doesn't appear in any Lean training data?
- Maybe because humans rarely write proofs in this style, so it doesn't appear in any natural-language training data?