

Gödel’s Incompleteness Theorem

Aayush Bajaj

July 31, 2026

Contents

1	The promise: Hilbert’s programme	1
2	The setting: what counts as a “system”	2
3	Arithmetisation: statements become numbers	3
4	Diagonalisation: the sentence that talks about itself	3
5	The first theorem: the pincer	4
6	The second theorem: consistency is the price	5
7	What it does not say	5
8	Aftermath	6
9	See also	6

In 1931 a twenty-five-year-old in Vienna published a paper proving that mathematics cannot fully formalise itself: any consistent formal system rich enough to do arithmetic contains true statements it cannot prove, and — worse — cannot even prove its own consistency (Gödel, Kurt, 1931). This page walks the actual argument: what the theorem says precisely, the two technical inventions that power it (Gödel numbering and diagonalisation), and — just as important — the long list of things it does *not* say.

The paper’s title promises ‘undecidable propositions of Principia Mathematica and related systems’ — the ‘related systems’ clause is doing heavy lifting: it means every system you will ever build

1 The promise: Hilbert’s programme

By the 1920s the foundations of mathematics had survived two frights — the paradoxes of naive set theory (Russell’s barber-shaped hole in Frege) and the intuitionist revolt against non-constructive proof. David Hilbert’s response was an engineering project: formalise all of mathematics in one axiomatic system, then prove — using only “finitary”, uncontroversial reasoning — that the system is **consistent** (it never proves a contradiction) and **complete** (it settles every statement one way or the other). Proof itself would become an object of mathematical study, and mathematics would certify its own reliability by purely mechanical means.

Hilbert, 1900: ‘in mathematics there is no ignorabimus’ — no ‘we shall not know’. Gödel’s paper is the ignorabimus, typeset.

Gödel’s two theorems close both doors. The first says no such system is complete; the second says the consistency proof Hilbert wanted is exactly the kind of statement the system cannot deliver about itself.



Figure 1: Kurt Gödel in 1925, as a student in Vienna — six years before the incompleteness paper. (Photograph in the public domain.)

2 The setting: what counts as a “system”

The theorems are about **formal theories**: a language of **first-order logic**, a set of axioms, and the usual deduction rules. Three properties matter:

- **Consistent**: there is no formula φ with both $T \vdash \varphi$ and $T \vdash \neg\varphi$. An inconsistent theory proves everything, so it measures nothing.
- **(Negation-)complete**: for every sentence φ in the language, $T \vdash \varphi$ or $T \vdash \neg\varphi$. The theory leaves no question open.¹
- **Effectively axiomatisable**: an algorithm can decide what is and is not an axiom. This is the “no cheating” clause — the set of all true sentences of arithmetic is a complete consistent theory, but there is no mechanical way to list it, so it is not a formal system in any useful sense.

The last ingredient is arithmetic strength: the theory must interpret enough of the arithmetic of $\mathbb{N}$ — addition, multiplication, a little induction. **Peano Arithmetic** (PA) is the canonical example, but weak fragments (Robinson’s Q) already suffice, and anything stronger (ZFC, your favourite proof assistant’s kernel) is automatically in scope.

First incompleteness theorem. Let T be a consistent, effectively axiomatisable theory that interprets enough arithmetic. Then there is a sentence G_T in the language of T such that $T \not\vdash G_T$ and $T \not\vdash \neg G_T$ — and G_T is true in the standard model $\mathbb{N}$.

¹“Complete” is the most overloaded word in logic: negation-completeness of a *theory* (at issue here), semantic completeness of a *proof calculus* (Gödel 1929), refutation-completeness of *resolution*, NP-completeness of a *problem*. Four different concepts on one syllable.

Note the shape of the trade-off: you may have consistency, effective axioms, arithmetic strength, or completeness — any three, never all four.

3 Arithmetisation: statements become numbers

The first invention is bureaucratic and the whole proof rests on it. Assign every symbol of the language a number — say $\neg \mapsto 1$, $\forall \mapsto 2$, $= \mapsto 3$, and so on — and encode a formula (a sequence of symbols) as a single number using prime factorisation:

$$\ulcorner s_1 s_2 \dots s_k \urcorner = 2^{c(s_1)} \cdot 3^{c(s_2)} \dots p_k^{c(s_k)}, \quad (1)$$

where $c(s_i)$ is the code of the i -th symbol and p_k is the k -th prime. Unique factorisation (**number theory** earning its keep) makes the coding reversible: from the number you can mechanically recover the formula. A proof — a finite sequence of formulas — gets coded the same way, one level up.

The specific coding is irrelevant, and nobody computes these numbers — they are astronomically large. What matters is that coding and decoding are computable.

Now the pivotal observation: syntactic properties of formulas become *arithmetic* properties of their code numbers. “ n codes a well-formed formula”, “ m codes a proof of the formula coded by n ” — these are statements about numbers, checkable by primitive-recursive bookkeeping, and Gödel showed the checking is expressible by formulas of arithmetic itself. Define

$$\text{Prov}_T(n) \equiv \exists m \text{Prf}_T(m, n), \quad (2)$$

where $\text{Prf}_T(m, n)$ formalises “ m codes a T -proof of the formula coded by n ”. The loop has closed: arithmetic talks about numbers, numbers encode formulas, so arithmetic talks about formulas — including its own.

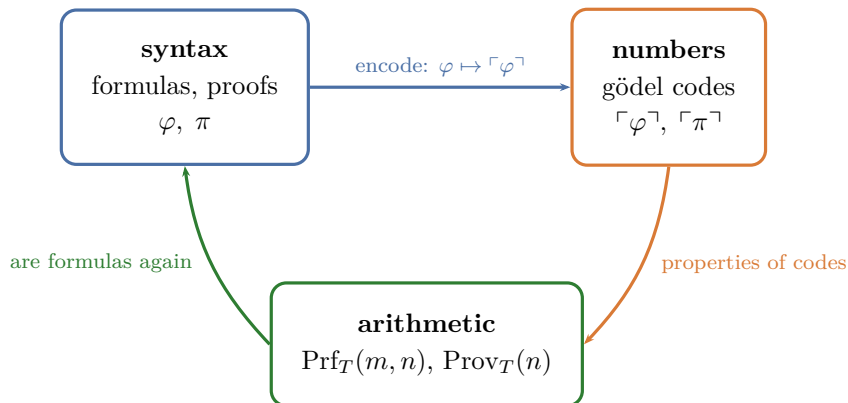


Figure 2: The arithmetisation loop. Arithmetic talks about numbers; Gödel numbering makes numbers stand for formulas and proofs; so arithmetic ends up talking about itself.

4 Diagonalisation: the sentence that talks about itself

The second invention manufactures self-reference without any hand-waving about “this sentence”. It is the same diagonal move Cantor used to show $\mathbb{R}$ uncountable (**sets**), pointed at provability instead of decimal digits.

Diagonal lemma. For any formula $\psi(x)$ with one free variable there is a sentence σ such that $T \vdash \sigma \leftrightarrow \psi(\ulcorner \sigma \urcorner)$. σ “says” that ψ holds of σ ’s

own code.

The construction is short and feels like a card trick: consider the operation “substitute a formula’s own code into itself”, show it is computable hence arithmetically expressible, and apply it to itself.

Now take $\psi(x) := \neg \text{Prov}_T(x)$. The diagonal lemma hands us the **Gödel sentence**:

$$T \vdash G_T \leftrightarrow \neg \text{Prov}_T(\ulcorner G_T \urcorner). \quad (3)$$

G_T asserts, via the coding, “ G_T is not provable in T ”. Note this is a perfectly concrete (if enormous) statement about natural numbers — the self-reference lives entirely in our interpretation of the codes.

5 The first theorem: the pincer

Suppose $T \vdash G_T$. Then there is an actual proof, so $\text{Prf}_T(m, \ulcorner G_T \urcorner)$ holds for some concrete m , and T — which can verify finite computations — proves $\text{Prov}_T(\ulcorner G_T \urcorner)$. But $T \vdash G_T$ with the fixed-point equivalence also gives $T \vdash \neg \text{Prov}_T(\ulcorner G_T \urcorner)$. Contradiction: T is inconsistent. So if T is consistent, $T \not\vdash G_T$.

Suppose instead $T \vdash \neg G_T$, i.e. $T \vdash \text{Prov}_T(\ulcorner G_T \urcorner)$. By the previous paragraph no proof of G_T exists, so for every concrete m the theory proves $\neg \text{Prf}_T(m, \ulcorner G_T \urcorner)$ — yet it also proves $\exists m \text{Prf}_T(m, \ulcorner G_T \urcorner)$. The theory affirms an existential while refuting every instance. Gödel’s original argument outlaws this by assuming **ω -consistency**; Rosser’s 1936 refinement replaces the Gödel sentence with a cleverer one (“for every proof of me there is a shorter proof of my negation”) and needs only plain consistency.²

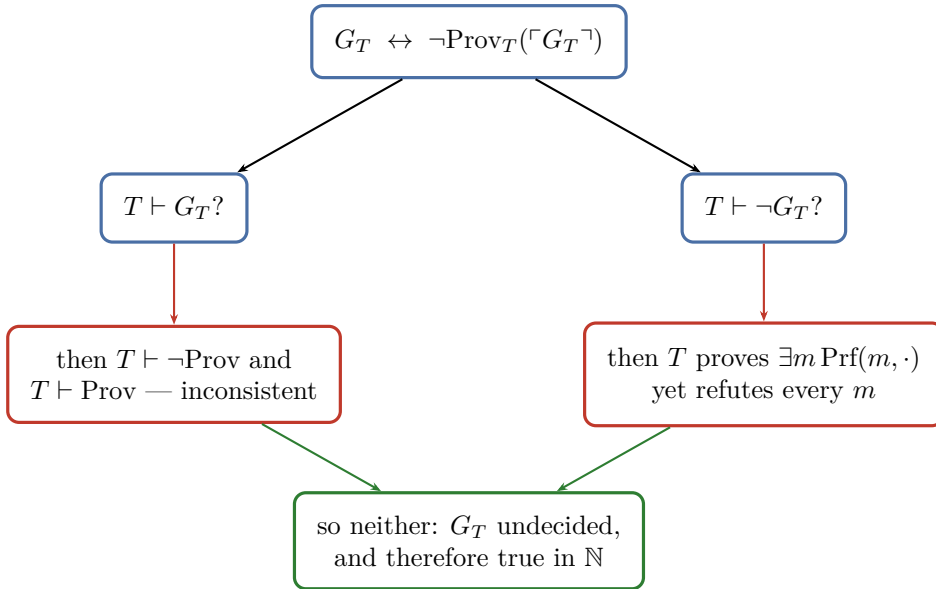


Figure 3: The pincer. Either decision on the Gödel sentence collapses the theory, so a consistent theory makes no decision — and undecided-but-true is exactly incompleteness.

²A theory is **ω -inconsistent** if it proves $\exists x \varphi(x)$ while also proving $\neg \varphi(\bar{n})$ for every numeral $\bar{n}$. Such a theory can be consistent — it just has no model whose “numbers” are only the standard ones. Rosser’s trick makes the whole argument immune to this subtlety, which is why modern statements of the theorem require only consistency.

Programmers meet the same trick as quines — programs that print their own source. The diagonal lemma is the quine construction, done in arithmetic.

And now the sting in the tail: since $T \not\vdash G_T$, no number actually codes a proof of G_T — which is precisely what G_T asserts: **the Gödel sentence is true**. Truth in $\mathbb{N}$ and provability in T come apart, permanently: adding G_T as a new axiom just produces a stronger theory T' with its own Gödel sentence $G_{T'}$. The hole moves; it never closes.

'True' here is not mystical: it means true in the standard model $\mathbb{N}$, the structure the axioms were written to describe.

6 The second theorem: consistency is the price

The second theorem is the first one, proved *inside* the system. Look at what we established: “if T is consistent then $T \not\vdash G_T$ ”. Write $\text{Con}(T)$ for the arithmetised consistency statement $\neg \text{Prov}_T(\ulcorner 0 = 1 \urcorner)$. The entire pincer argument uses only reasoning that T itself can carry out — provided Prov_T satisfies the three **Hilbert–Bernays–Löb derivability conditions**:

1. If $T \vdash \varphi$ then $T \vdash \text{Prov}_T(\ulcorner \varphi \urcorner)$ — the theory notices its own proofs;
2. $T \vdash \text{Prov}_T(\ulcorner \varphi \rightarrow \psi \urcorner) \rightarrow (\text{Prov}_T(\ulcorner \varphi \urcorner) \rightarrow \text{Prov}_T(\ulcorner \psi \urcorner))$ — provability respects modus ponens;
3. $T \vdash \text{Prov}_T(\ulcorner \varphi \urcorner) \rightarrow \text{Prov}_T(\ulcorner \text{Prov}_T(\ulcorner \varphi \urcorner) \urcorner)$ — condition 1, internalised.

Formalising the pincer yields $T \vdash \text{Con}(T) \rightarrow G_T$. So if T proved its own consistency it would prove G_T — which the first theorem forbids:

Second incompleteness theorem. If T is consistent (effectively axiomatisable, arithmetically strong), then $T \not\vdash \text{Con}(T)$.

This is the theorem that ends Hilbert’s programme on its own terms: any “finitary” consistency proof for PA would live inside PA (finitary reasoning being the weakest thing around), and PA cannot host it. Consistency certificates must always be purchased from a *stronger* theory — whose consistency is in turn on credit.

Gentzen, 1936: PA is consistent, proved by induction up to the ordinal ε_0 — genuinely illuminating, but the induction principle used is itself beyond PA. The credit line never clears.

7 What it does not say

No theorem attracts misuse like this one, so the fine print deserves its own section.

- **It does not say every system is incomplete.** Drop any hypothesis and completeness is back on the menu:

Theory	Complete?	Decidable?	Why it escapes
Presburger arithmetic ($\mathbb{N}, +$)	yes	yes	No multiplication — too weak to code proofs
Real closed fields (Tarski)	yes	yes	No way to define $\mathbb{N}$ inside $\mathbb{R}$
True arithmetic $\text{Th}(\mathbb{N})$	yes	no	Not effectively axiomatisable
PA, ZFC, ...	no	no	All hypotheses hold

- **It does not conflict with Gödel’s own completeness theorem (1929).** That result says first-order *logic* proves every semantically valid consequence; incompleteness says no effective axiom set captures all truths of the one intended model $\mathbb{N}$. The [logic](#) page keeps the two apart.
- **It does not say the Gödel sentence is interesting mathematics.** G_{PA} is a contrived numerical statement. But natural examples followed: Goodstein’s theorem and the Paris–Harrington principle are ordinary-looking combinatorial truths unprovable in PA.
- **It does not license “therefore minds beat machines” or “therefore nothing is certain”.** The Lucas–Penrose argument needs the premise that

we can *know* our own reasoning to be consistent — exactly what the second theorem makes suspect for any formal system, human-implemented or otherwise (Nagel, Ernest and Newman, James R., 2001).

8 Aftermath

Five years later Turing recast the whole story computationally: the halting problem is undecidable, and incompleteness falls out as a corollary — a complete effective theory of arithmetic would let you decide halting by searching for proofs (Hodges, Andrew, 1983). The two results are one phenomenon in different costumes: diagonalisation against provability, diagonalisation against computability. Provability itself became a subject — the modal logic GL axiomatises exactly what Prov_T can say about itself — and proof assistants live comfortably with the limitation: they verify mathematics daily while cheerfully declining to certify their own kernels.

The same diagonal also gives Löb's theorem: if $T \vdash \text{Prov}_T(\ulcorner \varphi \urcorner) \rightarrow \varphi$ then $T \vdash \varphi$. Believing your own press has consequences.

The theorem's popular reputation is as a wrecking ball, but inside mathematics it functions as a map: it marks where the boundary of each formal system lies, and guarantees the hierarchy of theories above PA — each proving the consistency of the ones below — never flattens into a final system. Mathematics did not collapse in 1931; it acquired a horizon.

9 See also

- **logic** — first-order logic, soundness, and the 1929 completeness theorem this page keeps being confused with
- **sets, functions and sequences** — Cantor's diagonal argument, the ancestor of the diagonal lemma
- **number theory** — unique factorisation, the substrate of Gödel numbering
- **classical algorithms** — computability, the halting problem's home turf

References

Gödel, Kurt (1931). *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I*, Monatshefte für Mathematik und Physik.

Hodges, Andrew (1983). *Alan Turing: The Enigma*, Simon and Schuster.

Nagel, Ernest and Newman, James R. (2001). *Gödel's Proof*, New York University Press.