Proving at Scale for Universal Algebra

João Araújo
Universidade Nova de Lisboa, Portugal

Jan Hůla
Czech Technical University in Prague, Czechia

Mikoláš Janota
Czech Technical University in Prague, Czechia

Edmond W. H. Lee
Nova Southeastern University, USA

Bartosz Naskręcki
Adam Mickiewicz University, Poznań, Poland
Warsaw University of Technology, Warsaw, Poland

Abstract

We introduce SEMIBASE, a project that computes and formally certifies finite identity bases for small semigroups. Deciding finite basability is undecidable for finite algebras and remains open for finite semigroups. The task requires a proof that a candidate basis is complete, or a proof that none exists, rather than a single first-order validity query. LLM-guided agents search for these proofs; a referee agent rebuilds them from source, and the Lean kernel checks the resulting corpus in a final audit. Humans choose targets and approve final outcomes. We certify every semigroup of order at most 6: all 1,309 semigroups of order at most 5 and all 15,973 of order 6, including proofs that the four known nonfinitely based semigroups have no finite basis. The bases for order 6 define 505 distinct varieties, whose complete inclusion order we determine by finite evaluation. The resulting catalogue is a machine-checked account of results scattered across the literature and a tested foundation for order 7.

1 Introduction

Tarski [1968] asks the following question about a given finite algebra: can all identities that hold in it be described by a finite set of identities? Consider, for example, the group $\mathbb{Z}_2$ with addition modulo 2. In addition to the group axioms, it satisfies the identity $x + x = 0$. It also satisfies $x + x + x + x = 0$, but this is a simple consequence of the former; in fact, every identity of the group follows from it. The group $\mathbb{Z}_2$ therefore has a finite basis consisting of a single identity, assuming the group axioms. Oates and Powell [1964] show that finite groups always have a finite basis. McKenzie [1996] shows that given a finite algebra as input, deciding whether it has a finite basis is *undecidable*.

The project SEMIBASE tackles the finite-basis problem for finite semigroups, a setting that has been studied extensively; see the surveys of Volkov [2001] and Araújo et al. [2023]. The smallest nonfinitely based semigroups have 6 elements; in fact, there are only 4 distinct nonfinitely based semigroups of that order as shown by Lee et al. [2012]. A complete catalogue of nonfinitely based semigroups of order 7 is not yet known. In general, it is not known whether the existence of a finite basis is decidable, but it is believed to be undecidable. This provides a setting for our project to explore the frontiers of what is known about small semigroups.

The project is building a catalogue in which each covered semigroup is equipped with either a certified finite basis or a certified proof that it does not have one. Proofs are carried out in Lean 4 [Moura and Ullrich, 2021]. Currently, the project covers all 1,309 semigroups of order at most 5 and all 15,973 semigroups of order 6. The work was done by language-model agents. At our direction they wrote

Preprint.

Table 1: Number of distinct semigroups per order, and how many of them are nilpotent.

Order	Semigroups	Nilpotent	3-nilpotent
1	1	1	0
2	4	1	0
3	18	2	1
4	126	10	8
5	1,160	93	84
6	15,973	2,813	2,660
7	836,021	616,830	609,797

the scripted pipeline that settled most of the catalogue; they proposed candidate bases and searched for counter-models to them, wrote the proofs shared by families of semigroups and the generators of the individual proofs, and ran every build. Humans chose which semigroups to attack, approved changes to the repository and restarted stalled agents, with about 1,100 prompts over three months (Appendix C). Mathematicians working on the finite basis problem stay in the loop [Avigad, 2026].

To our knowledge, SEMIBASE is unique in using an autonomous, agent-driven workflow to pursue a systematic classification of finite identity bases for small semigroups. A useful comparison is the Equational Theories Project (ETP), a large-scale, machine-assisted study of equational reasoning whose results are checked in Lean. ETP determined the implication relations among laws on magmas by combining human-generated proofs and constructions with automated theorem provers and other computational tools [Bolan et al., 2025]. Subsequent experiments showed that Vampire proves every valid implication in this benchmark and refutes the vast majority of the invalid ones [Janota, 2025]. The projects differ primarily in how their search spaces are organized. ETP studies a fixed finite graph of implications between individual laws, whereas SEMIBASE organizes instances by finite semigroup and asks whether the semigroup's full equational theory has a finite basis. This requires producing a candidate basis and certifying its completeness, or proving that no finite basis exists. COPRA [Thakur et al., 2023] runs a language model inside a backtracking proof search and retrieves lemmas from a library, and LEGO-Prover [Wang et al., 2023] grows its own library of verified lemmas; our family proofs and shared library play the role of such a library at the scale of a catalogue. Studies of language models as proof checkers find their verdicts sensitive to the prompt and to the form of the proof [Naik et al., 2026, Mazdarani and Toxtli, 2026]. Our referee agent is therefore not trusted to verify: it triages work and settles disputes, but a class is accepted only when the Lean kernel re-checks its proof in the whole-corpus audit.

Our contributions are the following. (i) Lean proofs, rebuilt from source in a single audit, that each semigroup of order at most 6 has a finite basis or, for four semigroups of order 6, none; to our knowledge this is the first machine-checked account of a classification that so far existed only across the literature [Lee and Li, 2011, Lee and Zhang, 2015, Araújo et al., 2023]. (ii) The complete inclusion order of the 505 varieties generated by the finitely based semigroups of order 6, determined by finite evaluation of their certified bases, and 289 hard unit-equality problems encountered during proof search with Vampire [Bártek et al., 2025], prepared for the TPTP library [Sutcliffe, 2017]. (iii) A description of the agent workflow and its acceptance procedure, with measured effort, failure rates and the case where the agents stalled.

2 Preliminaries

A *semigroup* is a structure with an associative binary operation. In the remainder of the paper we denote the operation by $*$ and assume that it operates on the elements $1, \ldots, n$. The number of elements that the semigroup operates on, n, is called the *order* of the semigroup. Whenever convenient we omit the $*$; for instance, $x * x = x$ is abbreviated as $xx = x$ or $x^2 = x$.

A finite semigroup is represented explicitly by its multiplication table. Table 1 gives the number of *distinct* semigroups, where *distinct* means that isomorphic and anti-isomorphic semigroups are counted only once (we ignore element relabelling and transpositions of the multiplication table).

Fix a countably infinite set $\mathcal{X}$ of variables and let $\mathcal{X}^+$ be the nonempty words over $\mathcal{X}$ constructed by the operator $*$. An *identity* is an equality $u \approx v$ with $u, v \in \mathcal{X}^+$. A semigroup S *satisfies* $u \approx v$, written $S \models u \approx v$, if $\varphi(u) = \varphi(v)$ for every homomorphism $\varphi\colon \mathcal{X}^+ \to S$, that is, for every map $\mathcal{X} \to S$ extended multiplicatively to words. The *equational theory* of S is the set $\mathrm{Id}(S) = \{\, u \approx v \mid S \models u \approx v \,\}$ of all identities that hold in S. A set of identities $\Sigma \subseteq \mathrm{Id}(S)$ is an *identity basis* of S if every identity in $\mathrm{Id}(S)$ is derivable from Σ by Birkhoff's rules of equational deduction [Birkhoff, 1935]; equivalently, if every semigroup that satisfies Σ satisfies all of $\mathrm{Id}(S)$. Semigroup S is *finitely based* if it has a finite identity basis and *nonfinitely based* otherwise.

A semigroup has a *zero element* z if $\forall x.\ xz = zx = z$; additionally it is *nilpotent* if there is a $k \in \mathbb{N}$ such that $\forall x_1, \ldots, x_k.\ x_1 \ldots x_k = z$. Nilpotent semigroups are a sixth of those of order 6 but three quarters of those of order 7 (see Table 1). They are the easy case for our purposes, since a nilpotent semigroup of degree k satisfies $x_1 \ldots x_k \approx y_1 \ldots y_k$ and is therefore finitely based (intuitively, only short identities matter).

Example 1. Let S_ℓ be the semigroup on $\{1, 2\}$ given by left projection, $x * y = x$; associativity holds because $(x * y) * z = x = x * (y * z)$. Under any assignment a word evaluates to the value of its leftmost variable, so $S_\ell \models u \approx v$ exactly when u and v begin with the same variable. The identity $xy \approx x$ is a basis: it rewrites any word to its first variable, after which the two sides of an identity of $\mathrm{Id}(S_\ell)$ coincide. Note that S_ℓ has no zero element, since $1 * 2 = 1$ and $2 * 1 = 2$, and is thus not nilpotent, yet it is still finitely based. Appendix F prints the Lean proof of this example.

A semigroup T belongs to the *variety generated by* S, denoted $\mathsf{V}(S)$, exactly when it satisfies every identity of S [Birkhoff, 1935]. Any basis of S therefore defines $\mathsf{V}(S)$.

3 The proof factory

SEMIBASE produces its proofs with a small team of language-model agents (Figure 2 in the appendix). Three workers do the mathematics. A coordinator assigns semigroups to them, allocates compute and runs independent rebuilds. A referee certifies candidate bases, settles disputes between agents and decides what enters the catalogue. The workers and the coordinator are Codex agents; the referee is a Claude agent, so the reviewer and the reviewed come from different model families. Humans at occasional intervals choose which semigroups to attack next and restart stalled agents. Each agent appends records to a shared log and can exchange files with other agents through per-agent mailboxes; every record names the earlier records it builds on, so every proof can be traced back through the log to the human decision that started it.

Who did what. The pipeline that sealed the first 14,989 classes is a set of scripts, written by the agents at our direction in June and July, that runs without a language model in the loop. It applies bases found by a bounded search over short identities or taken from the literature. In this sense it is the deterministic baseline: it settled 94% of the classes and stopped. The 984 classes it left open needed individual work: the agents proposed candidate bases, searched for counter-models and wrote completeness proofs, for a family of semigroups or for a single one. Published bases entered as inputs where they existed, as for the neighbour of L below. Humans wrote no proofs: they chose targets, approved pushes and restarted agents, with 440 prompts to the Codex agents on 72 days and 674 to the referee on 61 days (Appendix C).

From a candidate to a proof. A candidate basis is first screened for counter-models: a semigroup that satisfies the candidate but not some identity of the target refutes it. A candidate that survives is written in Lean 4 as a theorem for the semigroup and for its *transpose*, that is, anti-isomorphic copy. Showing that the candidate identities hold on the table is a simple check; the main effort is in proving that every identity of the table follows from the candidates. The proofs that succeeded all have the same shape. A normal form for words is fixed, together with a few rewriting rules derived from the basis, and every word is shown to rewrite to its normal form. The normal form is then shown to be determined by finitely many features of the word that the table can see, such as the order in which letters first occur, the order in which they last occur, and the parity of each letter. Semigroups that share a basis share this proof; only the finite check on each table differs. We consider a given class *sealed* once the referee has rebuilt its proof from scratch together with the whole corpus, checked in the kernel that the theorem's table is the catalogue table and that only the standard axioms of Lean are used, and recorded the theorem names and file hashes in a certificate. Both loops of Figure 2 were

busy: from 8 August to 6 September, 76 candidates were refuted before any Lean was written, and 143 of 443 independent kernel checks sent a proof back to its author; after two failed attempts, later three, a worker had to stop and report to the referee.

The neighbour of L. The hardest class shows why the problem is difficult at all. It is the semigroup C_8 of Section 19 of Lee and Zhang [2015] (3,842 in SMALLSEMI), and it differs from L, one of the four nonfinitely based semigroups of order 6 [Lee et al., 2012], in exactly two entries of its multiplication table. That difference is enough to cut off the infinite family of identities that makes L nonfinitely based [Zhang and Luo, 2011]. The class is finitely based, with the basis of their Proposition 19.1. The first candidate, found by bounded search and approved by the referee, had nine identities in at most three variables. A worker refuted it with a semigroup of order 20, larger than any the screens examine, that satisfies all nine but not $xykxty \approx xxkxtyyx$, an identity of the class. The agents switched to the published basis, but its completeness proof did not close for another week, until a structural analysis by the referee supplied a test for equality of words in the class and a normal form. With these, one agent completed the proof overnight, reusing much of the earlier attempt; it was the last proof of the campaign (Appendix D).

Where the effort went. Two decisions saved the most work: sharing one basis across a family of semigroups, and looking for counter-models before writing Lean. Two mistakes cost the most: searching for bases with a word-length bound that was too small, and checking classes one batch at a time instead of all together.

4 Results

Coverage. All 1,309 semigroups of order at most 5 and all 15,973 semigroups of order 6 are sealed. The four semigroups of order 6 known to be nonfinitely based, L, B_2^1, A_2^g and A_2^1 [Lee et al., 2012] (numbers 3,843, 8,564, 8,878 and 13,747 in SMALLSEMI [Distler and Mitchell, 2024]), carry Lean proofs that no finite basis exists; every other sealed class carries a certified basis. The four proofs formalize the published arguments [Perkins, 1969, Trahtman, 1987, Sapir, 1988, Zhang and Luo, 2011, Lee and Zhang, 2015]: for every k, an identity of the semigroup does not follow from identities in at most k variables, because a property of words separating its two sides survives every derivation step with them (Appendix E).

4.1 Proof size

The certificates differ in size by four orders of magnitude. Most classes are covered by a family proof: one basis and one proof shared by every semigroup in the family, with only the check on each multiplication table repeated. The remaining 445 semigroups of order 6 have a generated proof of their own. The common step is a transfer from a semigroup certified earlier: if a semigroup S with basis B embeds in a direct power of the target T, or is a quotient of a subsemigroup of T, then every identity of T holds in S and so follows from B; a check on the table that T satisfies B then makes B a basis for T. Others combine the bases of the factors of a subdirect product, adapt the basis of a semigroup that T inflates, or enumerate the identities of a nilpotent T. Figure 1 (left) shows the sizes of these generated proofs, all files of a proof counted together. The median proof has a few hundred lines. Nine exceed 100,000 lines and together hold close to half of all generated proof text; the largest runs to 333,557. Family proofs, which an agent writes as ordinary Lean, are shorter, none above 10,000 lines; Table 2 in the appendix gives the quantiles by kind of file. Lines of Lean measure the size of a certificate, not the effort of finding it: the largest proofs are generated mechanically, while the 2,918 new lines for the neighbour of L took four weeks. The mathematics sits in the family proofs (median 170 lines) and the shared library.

4.2 Basis size and the order of varieties

Figure 1 (right) shows the sizes of the bases certified for the 15,524 classes sealed by family proofs. These bases fall into 486 syntactically distinct lists of between 1 and 64 identities, median 4. Over four fifths of the classes have a basis of at most five identities, 4,056 have a basis of two and 53 a basis of one; at the other end, 74 classes were sealed with 26 identities and a few with up to 64.

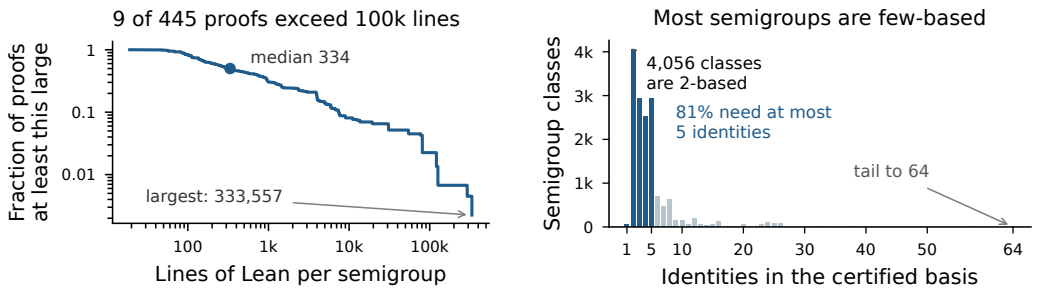

Figure 1: Left: share of the 445 generated proofs at least as long as a given number of lines, all files of a proof counted together (log axes). Right: the 15,524 classes sealed by family proofs, by the size of their basis as certified (22 of all 536 certified bases repeat an identity up to renaming), before the reduction of Section 4.2; at most five identities highlighted.

The certified bases need not be minimal, and distinct bases may define the same variety. Vampire removes identities that follow from the others, shrinking the 536 certified bases from 7,919 identities to 1,781; in 509 of the reduced bases every remaining identity is shown independent of the others by a counter-model, although an irredundant basis need not have minimum size. For the 143,380 pairs of bases, a semigroup of the order-6 catalogue separates all but 31, and Vampire proves each of those 31 pairs equivalent. The bases therefore define exactly 505 distinct varieties.

Finite evaluation completes the inclusion order (Appendix G): 13,325 strict inclusions, 1,317 covering pairs and 87 maximal varieties. The four directions left undecided by Vampire all hold. The 289 hardest proof-search problems are prepared for the TPTP library [Sutcliffe, 2017]. The Vampire proofs and table evaluations have not been formalized in Lean.

5 Summary and Conclusions

We have introduced the project SEMIBASE, which catalogues and formally certifies finite identity bases of small semigroups. The finite basis problem, originally proposed by Tarski [1968], has a longstanding tradition in universal algebra and is undecidable in general; the semigroup case remains open. Identity bases also matter because they can be seen as syntactic counterparts of varieties.

We show that the problem benefits from being attacked *en masse* rather than one semigroup at a time, because some semigroups share bases and some of the proof techniques recur. At this stage of the project, all 1,309 semigroups of order at most 5 and all 15,973 of order 6 carry a Lean certificate—either a finite basis or a proof that none exists. The Lean development, the certificate of the final audit and scripts that check both are archived on Zenodo [Naskręcki, 2026].

Agents produced the proofs, a referee and a whole-corpus audit made them trustworthy, and humans chose targets and approved pushes but wrote no proofs. All the proofs are kernel-checked in Lean 4. The catalogue is itself an object of study: Vampire has reduced its bases, finite evaluation has completed the inclusion order of the 505 varieties, and 289 hard unit-equality problems provide benchmarks for automated provers. The hardness of the proofs is strongly long-tailed: many of the proofs are almost routine, but a huge amount of resources must be dedicated to the last remaining few. This also suggests where to look for interesting proof techniques and problems; eventually we are likely to hit a case where the agents fail.

The next target is order 7, with 836,021 semigroups, 219,191 of them not nilpotent, an order of magnitude more than order 6. Order 6 took three months, about 2,260 agent sessions and 1,500 core-hours of Lean builds, and its cost sat in the tail: 15,583 of its 15,973 classes were sealed by 14 August, the remaining 390 took 23 more days, and the last needed a structural analysis before a proof could be written. Order 7 therefore needs the techniques used so far streamlined so that the tail, not only the bulk, is discharged automatically. The workflow carries over to any catalogue whose claims can be checked on each instance and stated in a proof assistant.

Acknowledgements

The research was supported by the European Union under the project ROBOPROX (reg. no. CZ.02.01.01/00/22_008/0004590) and is part of the RICAIP project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857306. Computational resources were supported by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254).

References

João Araújo, João Pedro Araújo, Peter J. Cameron, Edmond W.H. Lee, and Jorge Raminhos. A survey on varieties generated by small semigroups and a companion website. *Journal of Algebra*, 635:698–735, 2023. ISSN 0021-8693. doi:10.1016/j.jalgebra.2023.06.030.

Jeremy Avigad. Mathematicians in the age of AI, 2026. URL https://arxiv.org/abs/2603.03684.

Filip Bártek, Ahmed Bhayat, Robin Coutelier, Márton Hajdú, Matthias Hetzenberger, Petra Hozzová, Laura Kovács, Jakob Rath, Michael Rawson, Giles Reger, Martin Suda, Johannes Schoisswohl, and Andrei Voronkov. The Vampire diary. In Ruzica Piskac and Zvonimir Rakamaric, editors, *Computer Aided Verification - 37th International Conference, CAV 2025, Zagreb, Croatia, July 23-25, 2025, Proceedings, Part III*, volume 15933 of *Lecture Notes in Computer Science*, pages 57–71. Springer, 2025. doi:10.1007/978-3-031-98682-6_4.

Garrett Birkhoff. On the structure of abstract algebras. *Mathematical Proceedings of the Cambridge Philosophical Society*, 31(4):433–454, 1935. ISSN 0305-0041. doi:10.1017/S0305004100013463.

Matthew Bolan, Joachim Breitner, Jose Brox, Nicholas Carlini, Mario Carneiro, Floris van Doorn, Martin Dvorak, Andrés Goens, Aaron Hill, Harald Husum, Hernán Ibarra Mejia, Zoltan A. Kocsis, Bruno Le Floch, Amir Livne Bar-on, Lorenzo Luccioli, Douglas McNeil, Alex Meiburg, Pietro Monticone, Pace P. Nielsen, Emmanuel Osalotioman Osazuwa, Giovanni Paolini, Marco Petracci, Bernhard Reinke, David Renshaw, Marcus Rossel, Cody Roux, Jérémy Scanvic, Shreyas Srinivas, Anand Rao Tadipatri, Terence Tao, Vlad Tsyrklevich, Fernando Vaquerizo-Villar, Daniel Weber, and Fan Zheng. The equational theories project: Advancing collaborative mathematical research at scale, 2025. URL https://arxiv.org/abs/2512.07087.

Stanley N Burris and H P Sankappanavar. *A Course in Universal Algebra*. Graduate Texts in Mathematics. Springer, New York, NY, 1981. URL https://www.math.uwaterloo.ca/~snburris/htdocs/UALG/univ-algebra2012.pdf. Online Millennium Edition, 2012 update.

Andreas Distler and James D. Mitchell. *Smallsemi — a library of small semigroups*, 2024. URL https://gap-packages.github.io/smallsemi/. Version 0.7.1, GAP package.

Mikoláš Janota. Experimental results for Vampire on the equational theories project, 2025. URL https://arxiv.org/abs/2508.15856.

Edmond W. H. Lee and Jian Rong Li. Minimal non-finitely based monoids. *Dissertationes Mathematicae*, 475:1–65, 2011. ISSN 1730-6310. doi:10.4064/dm475-0-1.

Edmond W. H. Lee and Wen Ting Zhang. Finite basis problem for semigroups of order six. *LMS Journal of Computation and Mathematics*, 18(1):1–129, 2015. doi:10.1112/S1461157014000412.

Edmond W. H. Lee, Jian Rong Li, and Wen Ting Zhang. Minimal non-finitely based semigroups. *Semigroup Forum*, 85(3):577–580, 2012. ISSN 1432-2137. doi:10.1007/s00233-012-9434-y.

Fateme Mazdarani and Carlos Toxtli. Beyond the answer key: Robustness evaluation of large language models for step-level mathematical verification, 2026. URL https://arxiv.org/abs/2608.28725.

Ralph McKenzie. Tarski's finite basis problem is undecidable. *International Journal of Algebra and Computation*, 06(01):49–104, February 1996. ISSN 1793-6500. doi:10.1142/s0218196796000040.

Leonardo de Moura and Sebastian Ullrich. *The Lean 4 Theorem Prover and Programming Language*, pages 625–635. Springer International Publishing, 2021. ISBN 9783030798765. doi:10.1007/978-3-030-79876-5_37.

Aaditya Naik, Guruprerana Shabadi, Rajeev Alur, and Mayur Naik. Do we need frontier models to verify mathematical proofs?, 2026. URL https://arxiv.org/abs/2604.02450.

Bartosz Naskręcki. SemiBase: finite identity bases of the semigroups of order six. Zenodo, 2026. doi:10.5281/zenodo.23109225.

Sheila Oates and M.B Powell. Identical relations in finite groups. *Journal of Algebra*, 1(1):11–39, April 1964. ISSN 0021-8693. doi:10.1016/0021-8693(64)90004-3.

Peter Perkins. Bases for equational theories of semigroups. *Journal of Algebra*, 11(2):298–314, 1969. doi:10.1016/0021-8693(69)90058-1.

Tibor Radó. On non-computable functions. *Bell System Technical Journal*, 41(3):877–884, May 1962. ISSN 0005-8580. doi:10.1002/j.1538-7305.1962.tb00480.x.

Mark V. Sapir. Problems of Burnside type and the finite basis property in varieties of semigroups. *Mathematics of the USSR-Izvestiya*, 30(2):295–314, 1988. doi:10.1070/IM1988v030n02ABEH001012.

Geoff Sutcliffe. The TPTP problem library and associated infrastructure. *Journal of Automated Reasoning*, 59(4):483–502, 2017. doi:10.1007/s10817-017-9407-7.

Alfred Tarski. Equational logic and equational theories of algebras. In H. Arnold Schmidt, K. Schütte, and H.-J. Thiele, editors, *Contributions to Mathematical Logic*, volume 50 of *Studies in Logic and the Foundations of Mathematics*, pages 275–288. Elsevier, 1968. doi:10.1016/S0049-237X(08)70531-7.

Amitayush Thakur, George Tsoukalas, Yeming Wen, Jimmy Xin, and Swarat Chaudhuri. An in-context learning agent for formal theorem-proving, 2023. URL https://arxiv.org/abs/2310.04353.

Avraham N. Trahtman. Some finite infinitely basable semigroups. *Ural. Gos. Univ. Mat. Zap.*, 14(2): 128–131, 1987. In Russian.

M. V. Volkov. The finite basis problem for finite semigroups. *Scientiae Mathematicae japonicae*, 53(1):171–199, 01 2001. ISSN 13460862. Updated version https://s.csproject.org/MATHJAP_revisited.pdf.

Haiming Wang, Huajian Xin, Chuanyang Zheng, Lin Li, Zhengying Liu, Qingxing Cao, et al. LEGO-Prover: Neural theorem proving with growing libraries, 2023. URL https://arxiv.org/abs/2310.00656.

Wen Ting Zhang and Yan Feng Luo. A new example of a minimal nonfinitely based semigroup. *Bulletin of the Australian Mathematical Society*, 84(3), 2011. doi:10.1017/S0004972711002541.

A The proof factory, step by step

Figure 2 shows the steps from a multiplication table to a sealed class, arranged by who performs them; Section 3 describes them.

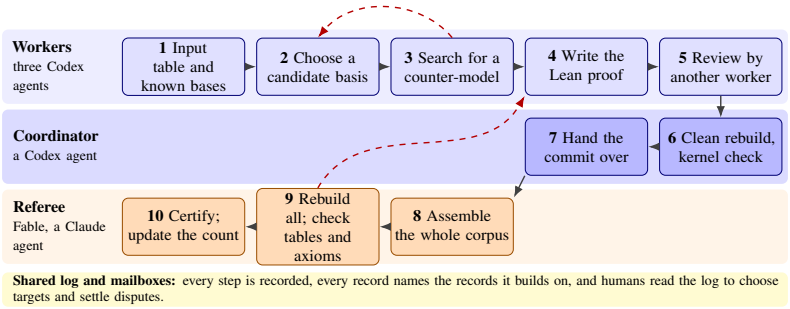

Figure 2: The steps from a multiplication table to a sealed class, arranged by who performs them (blue: Codex agents; orange: the referee). The dashed red arrows return work to an earlier step: a refuted candidate goes back to step 2, a proof that fails the audit back to step 4.

B How the Lean proofs are organised

Figure 3 shows how the Lean code is layered and what a single proof module contains. The acceptance repository, which adds the audit and wrapper modules to the proof repository counted in Section 4.1, holds 9,329 files and 6.94 million lines, on Lean 4.28 without Mathlib. The shared library is small; nearly all of the volume is proofs and audits about individual semigroups.

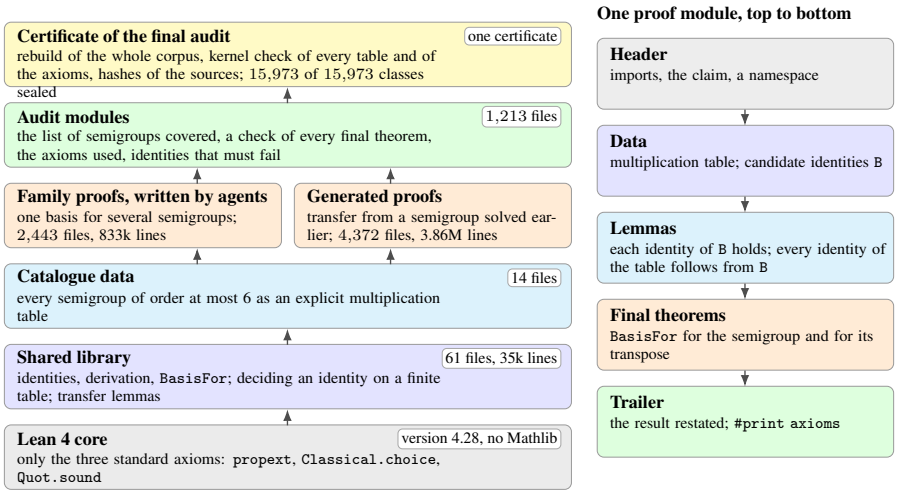

Figure 3: Left: the layers of the Lean code, each built on the one below. Right: the parts of a single proof module.

What a proof module contains. Every module, whether it treats one semigroup or a family of semigroups with a common basis, has the same shape. The header imports the shared library and the family's own lemma files, states the claim in a comment and opens a namespace named after the semigroup. The data are the multiplication table and the candidate identities. The lemmas show that each candidate identity holds on the table, which Lean checks by computation, and that every identity the table satisfies follows from the candidates; this second half is proved through normal forms or by decomposing the semigroup into a subdirect product of smaller semigroups that were treated before. The final theorems state `BasisFor` for the semigroup and for its transpose. A short trailer restates the result and prints the axioms used. Generated modules differ only in the middle:

they show that the target has the same identities as a semigroup solved earlier and reuse its basis; their heavy computations are split into several files.

Shared files. The core library is imported by everything. Its central file, `Equational.lean`, has 91 lines; it defines identities, satisfaction, models, derivation and `BasisFor`, and proves that derivation is sound. Each family keeps its own lemma files next to the proofs of its members, and each module imports only what it needs. A batch is therefore built from a fixed commit, and its audit module lists exactly which semigroups it covers.

The campaign. The campaign covers the 15,973 semigroups of order 6. A scripted pipeline first sealed 14,989 of them and left 984 open. The agents then took the remainder in 102 batches and sealed 943 more: 29 batches of ten or more semigroups with a common strategy, and 73 small batches down to single hard cases. The last 41 were delivered in the final week and certified together by the whole-corpus audit, after which all 15,973 classes are sealed. Under the batch process used until 5 September, a batch with a single semigroup took 37 minutes from submission to sealing and an independent verification job 2 to 20 minutes. The largest single build, the 333,557-line proof of one semigroup, took more than a day on one core.

Proof lengths. Section 4.1 plots the size of the generated proofs per semigroup. Table 2 gives the quantiles at the certified commit, per semigroup and by kind of file. The distribution has a heavy tail. Among the 445 semigroups with a generated proof of their own, the median proof has 334 lines but the mean is 7,227. Nine semigroups need more than 100,000 lines each and together hold close to half of all generated proof text; the largest, at 333,557 lines, is the proof whose build took a day. Family proofs, written by agents, are much shorter (median 170 lines, none above 10,000), because a shared basis and a subdirect decomposition replace the direct transfer that a generated proof carries out. The typical proof can be read by a person; the largest cannot, a growth in certificate size that recalls the busy-beaver problem [Radó, 1962]. The last classes to be sealed belong to this tail.

Kind of unit (lines of Lean)	n	median	p75	p90	p99	max	total
Generated proof of one semigroup (module and its parts)	445	334	1,466	7,398	126,350	333,557	3.22M
Family proof file, written by an agent	2,443	170	399	842	2,478	9,568	833k
Audit module	1,213	32	63	302	1,669	3,626	230k
Shared library file	61	207	382	538	18,584	18,584	35k

Generated proofs by size (lines)	< 100	100–300	300–1k	1k–3k	3k–10k	10k–30k	30k–100k	≥ 100k
semigroups	75	142	94	40	59	7	19	9

Table 2: Proof lengths at the certified commit of the final audit: quantiles over units of each kind (top), and the number of generated per-semigroup proofs in each size class (bottom).

C Effort and resources

Table 3 collects what the campaign cost, measured from the repository history, the agents' session logs, the shared log and the accounting of the two clusters that ran the Lean builds. Entries marked "about" are rounded. The shared log was introduced for the finishing campaign on 8 August, so the counts taken from it cover the last month only.

Duration	9 June to 6 September 2026: 90 days, with commits on 84 of them
Commits	4,923 on all branches, 3,588 of them in the history of the certified commit
Workers and coordinator	Codex agents (Codex CLI, GPT-5-family models): about 2,260 sessions, of which about 660 in June, 1,510 in July and 90 later; over 100 GB of session logs
Referee	a Claude agent in Claude Code: 43,236 model turns and 41.9 million output tokens up to 6 September
Human prompts	440 to the Codex agents on 72 days, 674 to the referee on 61 days
Shared log, 8 August to 6 September	4,046 records: 1,609 from the coordinator, 1,242 from the workers, 385 from the referee, the rest job records
Candidates refuted by a counter-model before any Lean was written	76
Independent kernel checks	443 jobs: 291 passed, 143 failed and sent the proof back to its author, 9 ended otherwise
Referee records that reject, withdraw or hold a proposal	about 106 of 385
Lean builds on the WMI cluster	18,222 jobs, about 1,500 core-hours
Largest single build	S6_12824, a proof of 333,557 lines: more than 24 hours on one core
Final audit	about 7,200 modules; seven runs over two days, the last, incremental, 4 h 6 min on 4 cores
Independent re-certification	a fresh clone rebuilt with 96 parallel jobs in 2 h: all 15,973 classes, no failures

Table 3: Effort and resources of the order-6 campaign.

The long tail. Table 4 shows the number of sealed classes over the finishing campaign. The scripted pipeline left 984 classes open; by 14 August only 390 remained, and those took another 23 days. The last 41 were certified together by the final audit.

Date	Sealed	Date	Sealed	Date	Sealed
scripted stage	14,989	28 August	15,776	2 September	15,928
14 August	15,583	29 August	15,856	3 September	15,930
25 August	15,649	30 August	15,886	4 September	15,932
26 August	15,711	31 August	15,894	6 September	15,973
27 August	15,720	1 September	15,921		

Table 4: Classes of order 6 sealed, at the last census of each day on which the census changed. The final audit of 6 September certified all 15,973 together.

Nilpotent classes. The records do not let us split the compute between nilpotent and other classes. Nilpotent classes are easy mathematically but not always cheap: the second-largest generated proof, 291,860 lines, certifies the nilpotent class S6_2582, whose basis has 3,513 identities.

D Worked example: the neighbour of L

The semigroup. The class S6_3842, the semigroup C_8 of Section 19 of Lee and Zhang [2015], has the elements $0, ex, x, ed, d, e$ and the table below, row times column. The idempotents d and e generate $\{0, ed, d, e\}$ with $de = 0$, and $\{0, x, ex\}$ is a null ideal with $dx = x$, $xe = x$ and $xd = 0$. The two shaded entries are the whole difference from L (S6_3843), in which $d \cdot e$ and $ed \cdot e$ are not zero.

$\cdot$	0	ex	x	ed	d	e
0	0	0	0	0	0	0
ex	0	0	0	0	0	ex
x	0	0	0	0	0	x
ed	0	0	ex	0	ed	0
d	0	0	x	0	d	0
e	0	ex	ex	ed	ed	e

Why it is finitely based. What separates the two semigroups is the product $x\,y^2\,x$: under $x \mapsto e$, $y \mapsto d$ it is $ede \neq 0$ in L but 0 in S6_3842, because $ed \cdot e = 0$. As a result S6_3842 satisfies identities that fail in L, such as $xyxy \approx xyyx$ and the exchange of adjacent squares $x\,y^2z^2t^2\,x \approx x\,z^2y^2t^2\,x$. The family $x\,y_1^2 \cdots y_n^2\,x \approx x\,y_n^2 \cdots y_1^2\,x$, which no finite set of identities of L derives (Appendix E), holds in both semigroups, but in S6_3842 it follows from a basis whose identities have at most five variables.

The basis. The certified basis is Proposition 19.1 of Lee and Zhang [2015]: the twelve schemata below, in which each of H, K, T is independently either absent or one further variable, expand to 38 identities.

$xxx \approx xx$
$xxyx \approx xyx$
$xyxx \approx xyx$
$xyxzx \approx xzxyx$
$xHyKxTy \approx xHxKxTyyx$
$xHyyx \approx yyxHyyx$
$xHyyxKx \approx xHyyxKyyx$
$xHyKxy \approx xHyKyx$
$xHyKxy \approx yHxKxy$
$xHyyxKy \approx xHyyxKx$
$yHxKyyx \approx xHxKyyx$
$xHyKyyx \approx xHxKyyx$

The first candidate consisted of $xx \approx xxx$, $xxyx \approx xyx$, $xyxx \approx xyx$, $xyxy \approx xyyx$, $xyxy \approx yxxy$, $xyxzx \approx xzxyx$, $xyxzy \approx xyzyx$, $xyxzy \approx xzyyx$ and $xyxzy \approx xyzxy$, all in at most three variables. It does not derive $xykxty \approx xxkxtyyx$, the instance of the fifth schema with H absent: a worker found a semigroup of order 20 that satisfies the nine identities but not this one. Replacing the ninth identity by the five-variable instance $xhykxty \approx xhxkxtyyx$ gives a basis equivalent to the published one; we checked the equivalence by explicit derivations in both directions, outside Lean.

The decisive observation. Two words are equal in S6_3842 exactly when they agree on a few combinatorial features. Call a letter *simple* if it occurs once, and cut a word into *components* wherever the letters before the cut and after it are disjoint. Words u and v are equal in the semigroup if and only if they have the same letters, the same simple letters and the same sequence of component contents, and, around each simple letter t, written $u = p\,t\,s$, the same contents of the trailing components of p that avoid the letters of s, and agree on whether p and s share a letter. A normal form follows: each component is rewritten to a canonical word determined by these features. Completeness then reduces to two facts, that the basis rewrites every word to its normal form and that words with the same features have the same normal form. The criterion was checked by computation on all words in up to four variables of bounded length before it was given to the agent as the plan of the proof.

How the proof was found.

8 August	The class enters the finishing campaign; a worker records its structure.
29 August	A worker's semigroup of order 20 refutes the nine-identity candidate; the referee withdraws it the same day and makes the published basis the only route.
29 August to 5 September	Parts of Lee and Zhang's argument are proved in Lean, but the completeness proof does not close.
5 September	The referee's structural analysis gives the criterion and the normal form above.
6 September, 01:10	A dedicated agent delivers the proof: 25 new modules with 2,918 lines and 169 theorems, reusing 63 modules (14,497 lines) of the earlier attempt.
6 September	The final audit certifies the class together with all others.

The certificate consists of two theorems, `BasisFor` for the table of S6_3842 with the 38 identities and for its transpose with the reversed identities, which use only the three standard axioms. Over the four weeks the shared log has 86 records about the class, from four agents.

E The four proofs that no finite basis exists

In Lean a semigroup is nonfinitely based when no finite list of identities over the variables $\mathbb{N}$ is a basis for it (Appendix F). The audit accepts a theorem as the proof for one of the four classes only if its statement is exactly `NonfinitelyBased` of the catalogue table, with no hypotheses, and if it uses only the three standard axioms.

All four proofs follow one scheme. They exhibit identities $u_n \approx v_n$ of the semigroup, one for every n, and a property of words that separates u_n from v_n. For every bound k, once n is large enough compared with k, no derivation step that applies an identity in at most k variables can change the property, so $u_n \approx v_n$ does not follow from such identities. The identities of a finite set have at most k variables for some k, so the set is not a basis. The proofs differ in the family and in the property (Table 5).

Class	Identities $u_n \approx v_n$	Property preserved by short steps	Lean
L S6_3843	$x\,y_1^2 \cdots y_n^2\,x \approx x\,y_n^2 \cdots y_1^2\,x$	words equal in L to u_n fall into two classes, with the squares in the forward or in the reverse order, and no short step crosses between them	6 files, 5,337 lines
B_2^1 S6_8564	$x\,y_1 \cdots y_n\,x\,y_n \cdots y_1 \approx$ $x\,y_n \cdots y_1\,x\,y_1 \cdots y_n$	a pattern of the occurrences of the letters; the argument uses that $xytyx$ and $xtyxy$ are isoterms, equal in B_2^1 only to themselves	8 files, 12,747 lines
A_2^1 S6_13747	$X\,y\,X'\,y\,X \approx X\,y\,X'\,y\,X\,y\,X'\,y\,X$, where $X = x_1 \cdots x_n$ and X' is its reverse	the word u_n itself: no short step changes it	8 files, 10,502 lines
A_2^g S6_8878	$(x_1^2 \cdots x_n^2)^2 \approx (x_1^2 \cdots x_n^2)^3$	the parity with which a word winds around the cycle $x_1 \to x_2 \to \cdots \to x_n \to x_1$	2 files, 2,625 lines

Table 5: The four nonfinitely based semigroups of order 6. A short step applies an identity whose number of variables is small compared with n. Line counts exclude audit modules and the 404 lines of shared definitions.

Sources. The families and arguments are those of Zhang and Luo [2011] for L, of Perkins [1969] in the reconstruction of Sapir [1988] for B_2^1, and of Trahtman [1987] and Sapir [1988] for A_2^1. For L the formalization reconstructs the proof of Lemma 2 of Zhang and Luo [2011], on which their reduction to connected identities rests. For A_2^g the proof characterizes the identities of the semigroup by a marked directed graph of the word together with a parity, and the winding argument works on that characterization. The four semigroups are those of the main theorem of Lee and Zhang [2015]. Where the catalogue lists a published table with its elements in another order (B_2^1 and A_2^g), the proof checks the relabelling and shows that both tables satisfy the same identities.

F A representative proof

Every certificate refers to the definitions below, from the shared library (`Equational.lean` and `Nonfinite.lean`, abridged to the definitions). A word is a nonempty list of variables, `G.eval` evaluates a word in `G` under a valuation of its variables, and `Derives` is equational deduction with substitution of words for variables.

```
def SatisfiedBy (e : Identity α) (G : Semigroup S) : Prop :=
  ∀ valuation : α → S, G.eval valuation e.lhs = G.eval valuation e.rhs

def Models (G : Semigroup S) (basis : List (Identity α)) : Prop :=
  ∀ e, e ∈ basis → e.SatisfiedBy G

/-- Equational derivability for semigroup words over one infinite variable type. -/
inductive Derives (basis : List (Identity α)) : Word α → Word α → Prop
  | fromBasis {e : Identity α} : e ∈ basis → Derives basis e.lhs e.rhs
  | refl (u : Word α) : Derives basis u u
  | symm {u v : Word α} : Derives basis u v → Derives basis v u
  | trans {u v w : Word α} :
```

```
      Derives basis u v → Derives basis v w → Derives basis u w
  | prepend (p : Word α) {u v : Word α} :
      Derives basis u v → Derives basis (p ++ u) (p ++ v)
  | appendRight {u v : Word α} :
      Derives basis u v → (q : Word α) → Derives basis (u ++ q) (v ++ q)
  | subst {u v : Word α} :
      Derives basis u v → (σ : α → Word α) →
        Derives basis (u.bind σ) (v.bind σ)

/-- A finite list is a basis when it is sound and derives every valid identity. -/
def BasisFor (G : Semigroup S) (basis : List (Identity α)) : Prop :=
  Models G basis ∧
    ∀ e : Identity α, e.SatisfiedBy G → Derives basis e.lhs e.rhs

def FinitelyBasedOver (G : Semigroup S) (α : Type u) : Prop :=
  ∃ basis : List (Identity α), BasisFor G basis

/-- Failure of finite basability relative to a chosen variable type. -/
def NonfinitelyBasedOver (G : Semigroup S) (α : Type u) : Prop :=
  ¬FinitelyBasedOver G α

/-- The standard nonfinite-basis predicate, using `Nat` variables. -/
def NonfinitelyBased (G : Semigroup S) : Prop :=
  NonfinitelyBasedOver G Nat
```

The proof below is complete. It is the module `Examples/LeftZeroTwo.lean` of the certified tree, for the left-zero semigroup S_ℓ of Example 1, up to its theorem for the semigroup itself; the remaining 39 lines derive the theorem for the transpose from it. A family proof of order 6 has the same parts at a larger scale: the table, the basis, soundness by computation, a normal form reached by derivation (here the first letter of a word, `leftZeroDerivesHead`), and completeness by a valuation that tells different normal forms apart.

```
import SemigroupBasis.FiniteTable
import SemigroupBasis.Opposite

namespace SemigroupBasis.Examples

open SemigroupBasis

def leftZeroTwo : FiniteTable where
  order := 2
  mul := fun a _ => a
  assoc := by intros; rfl

def xy : Word (Fin 2) := ⟨0, [1]⟩
def x : Word (Fin 2) := Word.singleton 0
def leftZeroLaw : Identity (Fin 2) := ⟨xy, x⟩

theorem leftZeroLaw_valid : leftZeroLaw.SatisfiedBy leftZeroTwo.semigroup :=
  leftZeroTwo.checkIdentity_sound leftZeroLaw (by decide)

def natXY : Word Nat := ⟨0, [1]⟩
def natX : Word Nat := Word.singleton 0
def leftZeroBasisLaw : Identity Nat := ⟨natXY, natX⟩
def leftZeroBasis : List (Identity Nat) := [leftZeroBasisLaw]

private def instantiateTwo (u v : Word Nat) : Nat → Word Nat
  | 0 => u
  | _ => v

theorem leftZeroTwo_eval (valuation : Nat → Fin 2) (w : Word Nat) :
    leftZeroTwo.semigroup.eval valuation w = valuation w.head := by
  cases w with
  | mk head tail =>
      induction tail with
      | nil => rfl
      | cons y ys ih =>
          simp only [Semigroup.eval, List.foldl_cons]
          exact ih

theorem leftZeroBasis_models : Models leftZeroTwo.semigroup leftZeroBasis := by
  intro e he
  simp only [leftZeroBasis, List.mem_singleton] at he
  subst e
  intro valuation
  rfl

theorem leftZeroDerivesHead (w : Word Nat) :
```

```
    Derives leftZeroBasis w (Word.singleton w.head) := by
  cases w with
  | mk head tail =>
      cases tail with
      | nil => exact Derives.refl _
      | cons next rest =>
          let suffix : Word Nat := ⟨next, rest⟩
          have hbase : Derives leftZeroBasis natXY natX :=
            Derives.fromBasis (e := leftZeroBasisLaw) <| by
              change leftZeroBasisLaw ∈ [leftZeroBasisLaw]
              exact List.Mem.head []
          have h := Derives.subst hbase
            (instantiateTwo (Word.singleton head) suffix)
          simpa [natXY, natX, instantiateTwo, Word.bind, Word.append,
            Word.singleton, suffix] using h

theorem leftZeroBasis_complete : BasisFor leftZeroTwo.semigroup leftZeroBasis := by
  refine ⟨leftZeroBasis_models, ?_⟩
  intro e hsatisfied
  have hheads : e.lhs.head = e.rhs.head := by
    apply Decidable.byContradiction
    intro hne
    let valuation : Nat → Fin 2 := fun z =>
      if z = e.lhs.head then 0 else 1
    have heval := hsatisfied valuation
    rw [leftZeroTwo_eval, leftZeroTwo_eval] at heval
    simp [valuation, Ne.symm hne] at heval
  exact Derives.trans (leftZeroDerivesHead e.lhs) <| by
    rw [hheads]
    exact Derives.symm (leftZeroDerivesHead e.rhs)
```

G Computing the inclusion order

The following finite inclusion test follows from Burris and Sankappanavar [1981, Chapter II, Lemmas 11.3 and 11.8].

Proposition 1. *Let S, T be finite semigroups and let B_T be a finite identity basis of T. Then*

$$\mathsf{V}(S) \subseteq \mathsf{V}(T) \quad \Longleftrightarrow \quad S \models B_T.$$

Consequently, this inclusion is decidable by finite evaluation of B_T on S.

Proof. Since $V(S)$ is the smallest variety containing S, the inclusion $V(S) \subseteq V(T)$ holds if and only if $S \in V(T)$. As B_T is an identity basis of T, this is equivalent to $S \models B_T$. Each identity in B_T involving k variables can be checked on all $|S|^k$ assignments. Since S and B_T are finite, this gives a terminating decision procedure. □

Evaluating each certified basis on a generator of each of the 505 varieties confirms all 13,321 inclusions proved by Vampire or transitivity and adds exactly the four directions that remained undecided after 12 hours of search each. The complete order has 13,325 strict inclusions, 1,317 covering pairs and 87 maximal varieties. Explicit equational derivations have since been found for three of the four directions; one remains to be constructed. The inclusion itself is settled by the finite check and the certified completeness of the bases.

The bases are certified in Lean. The finite table evaluations and Vampire's equational derivations have not been formalized in Lean.