EXCERPTED FROM

STEPHEN WOLFRAM

A NEW KIND OF SCIENCE

SECTION 12.9

Implications for Mathematics and Its Foundations

Implications for Mathematics and Its Foundations

Much of what I have done in this book has been motivated by trying to understand phenomena in nature. But the ideas that I have developed are general enough that they do not apply just to nature. And indeed in this section what I will do is to show that they can also be used to provide important new insights on fundamental issues in mathematics.

At some rather abstract level one can immediately recognize a basic similarity between nature and mathematics: for in nature one knows that fairly simple underlying laws somehow lead to the rich and complex behavior we see, while in mathematics the whole field is in a sense based on the notion that fairly simple axioms like those on the facing page can lead to all sorts of rich and complex results.

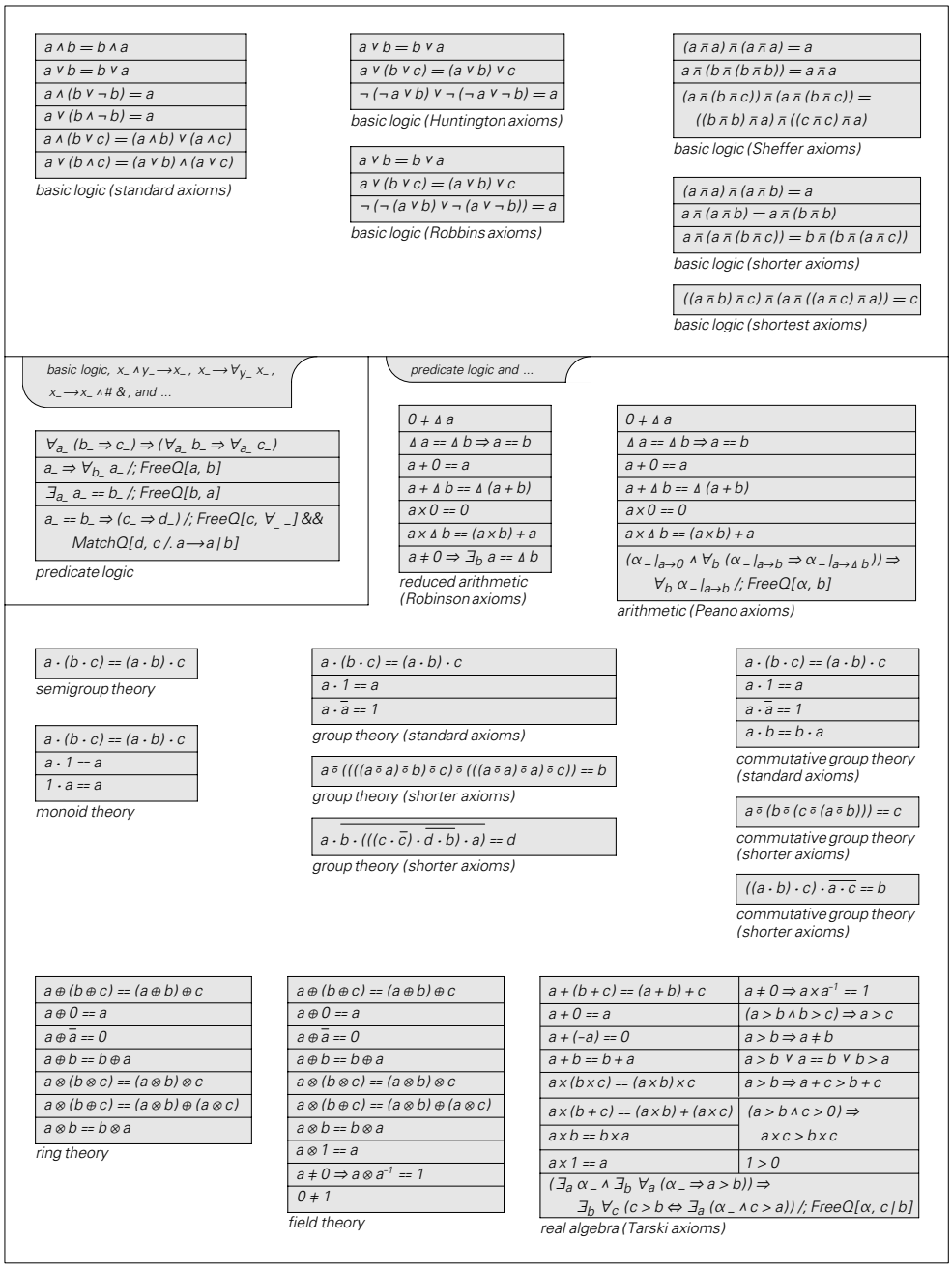
So where does this similarity come from? At first one might think that it must be a consequence of nature somehow intrinsically following mathematics. For certainly early in its history mathematics was specifically set up to capture certain simple aspects of nature.

But one of the starting points for the science in this book is that when it comes to more complex behavior mathematics has never in fact done well at explaining most of what we see every day in nature.

Yet at some level there is still all sorts of complexity in mathematics. And indeed if one looks at a presentation of almost any piece of modern mathematics it will tend to seem quite complex. But the point is that this complexity typically has no obvious relationship to anything we see in nature. And in fact over the past century what has been done in mathematics has mostly taken increasing pains to distance itself from any particular correspondence with nature.

So this suggests that the overall similarity between mathematics and nature must have a deeper origin. And what I believe is that in the end it is just another consequence of the very general Principle of Computational Equivalence that I discuss in this chapter.

For both mathematics and nature involve processes that can be thought of as computations. And then the point is that all these computations follow the Principle of Computational Equivalence, so



Axiom systems for traditional mathematics. It is from the axiom systems on this page and the next that most of the millions of theorems in the literature of mathematics have ultimately been derived. Note that in several cases axiom systems are given here in much shorter forms than in standard mathematics textbooks. (See also the definitions on the next page.)

predicate logic and ...

$(a \wr b \wr a) \Rightarrow a == b$
$((a \wr b \wr c) \wedge (b \wr d \wr c)) \Rightarrow (a \wr b \wr d)$
$((a \wr b \wr c) \wedge (a \wr b \wr d) \wedge a \neq b) \Rightarrow ((a \wr c \wr d) \vee (a \wr d \wr c))$
$\overline{a\,b} \equiv \overline{b\,a}$
$\overline{a\,b} \equiv \overline{c\,c} \Rightarrow a == b$
$(\overline{a\,b} \equiv \overline{c\,d} \wedge \overline{a\,b} \equiv \overline{e\,f}) \Rightarrow \overline{c\,d} \equiv \overline{e\,f}$
$\exists_a (((b \wr c \wr d) \wedge (e \wr d \wr f)) \Rightarrow ((b \wr a \wr e) \wedge (f \wr c \wr a)))$
$\exists_a \exists_b (((c \wr d \wr e) \wedge (f \wr d \wr g) \wedge c \neq d) \Rightarrow ((c \wr g \wr a) \wedge (c \wr f \wr b) \wedge (a \wr e \wr b)))$
$(\overline{a\,b} \equiv \overline{c\,d} \wedge \overline{b\,e} \equiv \overline{d\,f} \wedge \overline{a\,g} \equiv \overline{c\,h} \wedge \overline{b\,g} \equiv \overline{d\,h} \wedge (a \wr b \wr e) \wedge (c \wr d \wr f) \wedge a \neq b) \Rightarrow$ $\overline{e\,g} \equiv \overline{f\,h}$
$\exists_a ((b \wr c \wr a) \wedge \overline{c\,a} \equiv \overline{d\,e})$
$\exists_a \exists_b \exists_c (\neg (a \wr b \wr c) \wedge \neg (b \wr c \wr a) \wedge \neg (c \wr a \wr b))$
$(\overline{a\,b} \equiv \overline{a\,c} \wedge \overline{d\,b} \equiv \overline{d\,c} \wedge \overline{e\,b} \equiv \overline{e\,c} \wedge b \neq c) \Rightarrow ((a \wr d \wr e) \vee (d \wr e \wr a) \vee (e \wr a \wr d))$
$\exists_a \forall_b \forall_c ((\alpha\_ \wedge \beta\_) \Rightarrow (a \wr b \wr c)) \Rightarrow$ $\exists_d \forall_b \forall_c ((\alpha\_ \wedge \beta\_) \Rightarrow (b \wr d \wr c)) \,/;\, \text{FreeQ}[\alpha, a \mid c \mid d] \,\&\&\, \text{FreeQ}[\beta, a \mid b \mid d]$

Euclidean plane geometry

$a \cdot (b \cdot c) == (a \cdot b) \cdot c$
$a \cdot \triangleright a == a$
$\triangleleft\, a \cdot a == a$
$a \cdot \square == \square$
$\square \cdot a == \square$
$\triangleright \square == \square$
$\triangleleft \square == \square$
$(a \neq \square \wedge b \neq \square) \Rightarrow (a \cdot b \neq \square \Leftrightarrow \triangleright a == \triangleleft b)$

elementary category theory

$\forall_a (a \in b \Leftrightarrow a \in c) \Rightarrow b == c$	*(extensionality)*
$\neg\, a \in \emptyset$	*(empty set)*
$a \in \{b, c\} \Leftrightarrow (a == b \vee a == c)$	*(pairing)*
$a \in \cup b \Leftrightarrow \exists_c (c \in b \wedge a \in c)$	*(union)*
$\exists_a \forall_b (b \in a \Leftrightarrow \forall_c (c \in b \Rightarrow c \in d))$	*(power set)*
$\exists_a \forall_b (b \in a \Leftrightarrow (b \in c \wedge \alpha\_)) \,/;\, \text{FreeQ}[\alpha, a]$	*(subset)*
$\exists_a (\emptyset \in a \wedge \forall_b (b \in a \Rightarrow \cup \{b, \{b\}\} \in a))$	*(infinity)*
$\forall_a (a \in b \Rightarrow \forall_c \forall_d ((\alpha\_|_{e \to c} \wedge \alpha\_|_{e \to d}) \Rightarrow c == d)) \Rightarrow$ $\exists_f \forall_g (g \in f \Leftrightarrow \exists_a (a \in b \wedge \alpha\_|_{e \to g})) \,/;\, \text{FreeQ}[\alpha, c \mid d \mid f \mid g]$	*(replacement)*
$(\neg\, \emptyset \in a \wedge \forall_b \forall_c ((b \in a \wedge c \in a \wedge b \neq c) \Rightarrow b \cap c == \emptyset)) \Rightarrow$ $\exists_d \forall_b (b \in a \Rightarrow \exists_e d \cap b == \{e\})$	*(choice)*
$a \neq \emptyset \Rightarrow \exists_b (b \in a \wedge a \cap b == \emptyset)$	*(regularity)*

set theory

$\exists_{a\_}\, b\_ \longrightarrow \neg\, \forall_a \neg\, b$
$a\_ \Rightarrow b\_ \longrightarrow \neg\, a \vee b$
$a\_ \Leftrightarrow b\_ \longrightarrow (a \Rightarrow b) \wedge (b \Rightarrow a)$
$a\_ \cap b\_ == c\_ \longrightarrow$ $\forall_n (n \in c \Leftrightarrow (n \in a \wedge n \in b))$
$\{a\_\} \longrightarrow \{a, a\}$
$a\_|_{b\_ \to c\_} \longrightarrow \forall_b (b == c \Rightarrow a)$
$a\_ \subseteq b\_ \longrightarrow \forall_n (n \in a \Rightarrow n \in b)$
$a\_ \neq b\_ \longrightarrow \neg\, (a == b)$

definitions

$\wedge$	*and*
$\vee$	*or*
$\neg$	*not*
$\bar{\wedge}$	*nand*
$\forall$	*for all*
$\exists$	*there exists*
$\vartriangle$	*next integer*
$\cdot$	*composition*
$\bar{\circ}$	*inverse composition*
$\overline{\square}$	*inverse*
$\oplus$	*generalized addition*
$\otimes$	*generalized multiplication*
$\square^{-1}$	*reciprocal*
$\triangleleft$	*left identity morphism*
$\triangleright$	*right identity morphism*
$\square$	*morphism mismatch*
$(\square \wr \square \wr \square)$	*is between*
$\overline{\square\,\square} \equiv \overline{\square\,\square}$	*segments are congruent*
$\in$	*element of*
$\emptyset$	*empty set*
$\{\square, \square\}$	*pair*
$\cup$	*union*
$\mathbb{X}$	*set of all points*
$\mathcal{O}$	*set of all open sets*
$\mathbb{R}$	*set of all real numbers*

typical interpretations

set theory and ...

$a \in \mathcal{O} \Rightarrow a \subseteq \mathbb{X}$
$\emptyset \in \mathcal{O} \wedge \mathbb{X} \in \mathcal{O}$
$(a \in \mathcal{O} \wedge b \in \mathcal{O} \wedge a \cap b == c) \Rightarrow c \in \mathcal{O}$
$a \subseteq \mathcal{O} \Rightarrow \cup a \in \mathcal{O}$

general topology

real algebra with all objects restricted to $\mathbb{R}$
$(a \subseteq \mathbb{R} \wedge a \neq \emptyset \wedge \exists_b (b \in \mathbb{R} \wedge \forall_c (c \in a \Rightarrow c > b))) \Rightarrow$ $\exists_b (b \in \mathbb{R} \wedge \forall_d (d \in \mathbb{R} \Rightarrow (d > b \Leftrightarrow \exists_c (c \in a \wedge d > c))))$

real analysis

Further axiom systems for traditional mathematics. The typical interpretations are relevant for applications, though not for formal derivation of theorems. The last two axioms listed for set theory are usually considered optional.

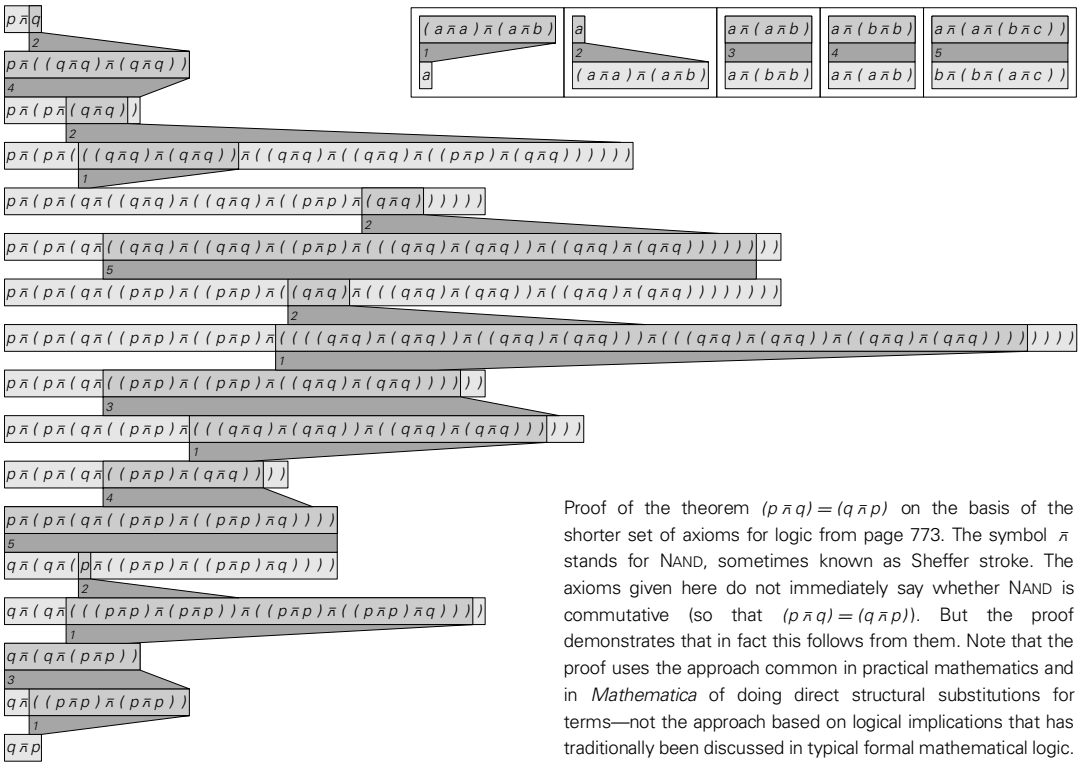
that they ultimately tend to be equivalent in their computational sophistication—and thus show all sorts of similar phenomena.

And what we will see in this section is while some of these phenomena correspond to known features of mathematics—such as Gödel's Theorem—many have never successfully been recognized.

But just what basic processes are involved in mathematics?

Ever since antiquity mathematics has almost defined itself as being concerned with finding theorems and giving their proofs. And in any particular branch of mathematics a proof consists of a sequence of steps ultimately based on axioms like those of the previous two pages.

The picture below gives a simple example of how this works in basic logic. At the top right are axioms specifying certain fundamental equivalences between logic expressions. A proof of the equivalence $p \barwedge q = q \barwedge p$ between logic expressions is then formed by applying these axioms in the particular sequence shown.

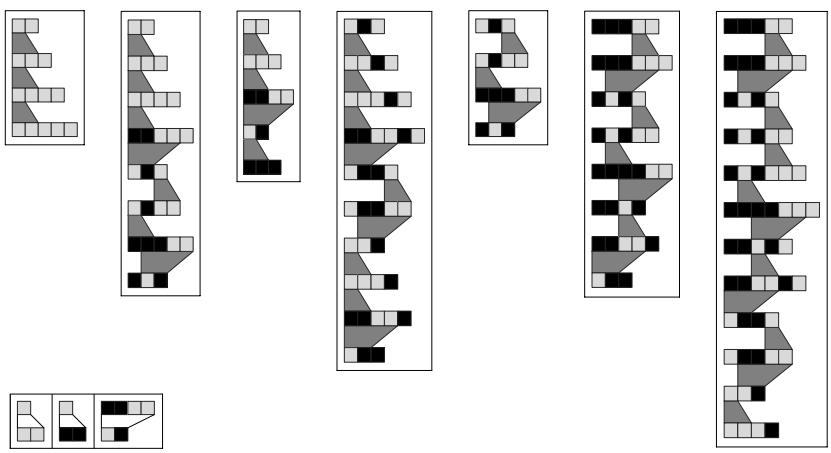


Proof of the theorem $(p \barwedge q) = (q \barwedge p)$ on the basis of the shorter set of axioms for logic from page 773. The symbol $\barwedge$ stands for NAND, sometimes known as Sheffer stroke. The axioms given here do not immediately say whether NAND is commutative (so that $(p \barwedge q) = (q \barwedge p)$). But the proof demonstrates that in fact this follows from them. Note that the proof uses the approach common in practical mathematics and in *Mathematica* of doing direct structural substitutions for terms—not the approach based on logical implications that has traditionally been discussed in typical formal mathematical logic.

In most kinds of mathematics there are all sorts of additional details, particularly about how to determine which parts of one or more previous expressions actually get used at each step in a proof. But much as in our study of systems in nature, one can try to capture the essential features of what can happen by using a simple idealized model.

And so for example one can imagine representing a step in a proof just by a string of simple elements such as black and white squares. And one can then consider the axioms of a system as defining possible transformations from one sequence of these elements to another—just like the rules in the multiway systems we discussed in Chapter 5.

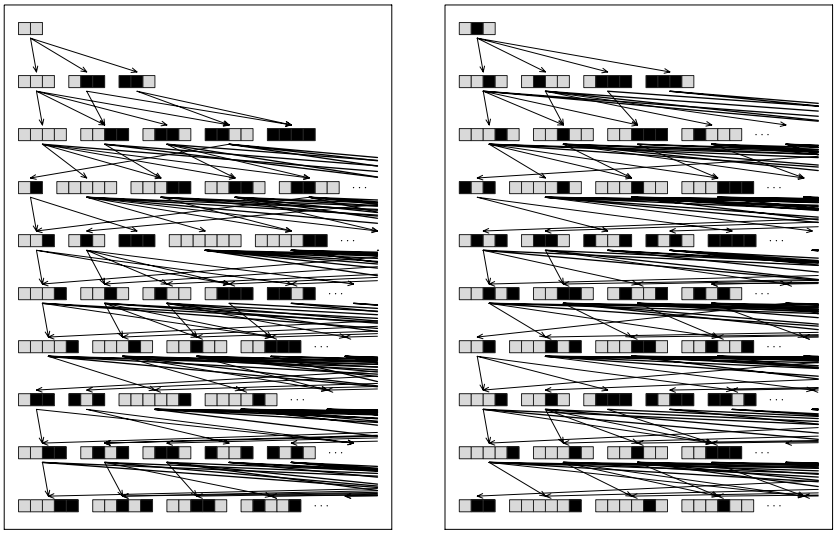
The pictures below show how proofs of theorems work with this setup. Each theorem defines a connection between strings, and proving the theorem consists in finding a series of transformations—each associated with an axiom—that lead from one string to another.



Simple idealizations of proofs in mathematics. The rules on the left in effect correspond to axioms that specify valid transformations between strings of black and white elements. The proofs above then show how one string—say □□—can be transformed into another—say □□□□□—by using the axioms. Typically there are many different proofs that can be given of a particular theorem; here in each case the ones shown are examples of the shortest possible proofs. The system shown is an example of a general substitution system of the kind discussed on page 497. Note that the fifth theorem □■□ → ■□■ occurs in effect as a lemma in the second theorem □□ → ■□■.

But just as in the multiway systems in Chapter 5 one can also consider an explicit process of evolution, in which one starts from a

particular string, then at each successive step one applies all possible transformations, so that in the end one builds up a whole network of connections between strings, as in the pictures below.



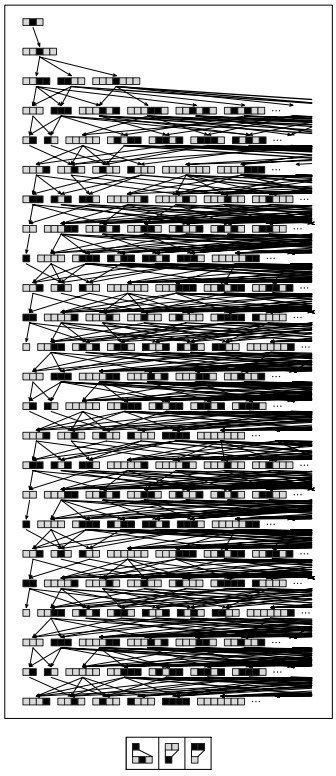
The result of applying the same transformations as on the facing page—but in all possible ways, corresponding to the evolution of a multiway system that represents all possible theorems that can be derived from the axioms. With the axioms used here, the total number of strings grows by a factor of roughly 1.7 at each step; on the last steps shown there are altogether 237 and 973 strings respectively.

In a sense such a network can then be thought of as representing the whole field of mathematics that can be derived from whatever set of axioms one is using—with every connection between strings corresponding to a theorem, and every possible path to a proof.

But can networks like the ones above really reflect mathematics as it is actually practiced? For certainly the usual axioms in every traditional area of mathematics are significantly more complicated than any of the multiway system rules used above.

But just like in so many other cases in this book, it seems that even systems whose underlying rules are remarkably simple are already able to capture many of the essential features of mathematics.

An obvious observation in mathematics is that proofs can be difficult to do. One might at first assume that any theorem that is easy



Three examples of multiway systems that show the analog of long proofs. In each case a string consisting of a single white element is eventually generated—but this takes respectively 12, 28 and 34 steps to happen. The first multiway system actually generates all strings in the end (not least since it yields the lemmas ■ → ■■ and ■ → □)—and in fact strings of length $n > 2$ appear after at most $2n + 7$ steps. The second multiway system generates only the $n + 1$ strings where black comes before white—and all of these strings appear after at most $7n$ steps. The third multiway system generates a complicated collection of strings; the numbers of lengths up to 8 are 1, 2, 4, 8, 14, 22, 34, 45. All the strings generated have an even number of black elements.

to state will also be easy to prove. But experience suggests that this is far from correct. And indeed there are all sorts of well-known examples—such as Fermat's Last Theorem and the Four-Color Theorem—in which a theorem that is easy to state seems to require a proof that is immensely long.

So is there an analog of this in multiway systems? It turns out that often there is, and it is that even though a string may be short it may nevertheless take a great many steps to reach.

If the rules for a multiway system always increase string length then it is inevitable that any given string that is ever going to be generated must appear after only a limited number of steps. But if the rules can both increase and decrease string length the story is quite different, as the picture on the facing page illustrates. And often one finds that even a short string can take a rather large number of steps to produce.

But are all these steps really necessary? Or is it just that the rule one has used is somehow inefficient, and there are other rules that generate the short strings much more quickly?

Certainly one can take the rules for any multiway system and add transformations that immediately generate particular short strings. But the crucial point is that like so many other systems I have discussed in this book there are many multiway systems that I suspect are computationally irreducible—so that there is no way to shortcut their evolution, and no general way to generate their short strings quickly.

And what I believe is that essentially the same phenomenon operates in almost every area of mathematics. Just like in multiway systems, one can always add axioms to make it easier to prove particular theorems. But I suspect that ultimately there is almost always computational irreducibility, and this makes it essentially inevitable that there will be short theorems that only allow long proofs.

In the previous section we saw that computational irreducibility tends to make infinite questions undecidable. So for example the question of whether a particular string will ever be generated in the evolution of a multiway system—regardless of how long one waits—is in general undecidable. And similarly it can be undecidable whether

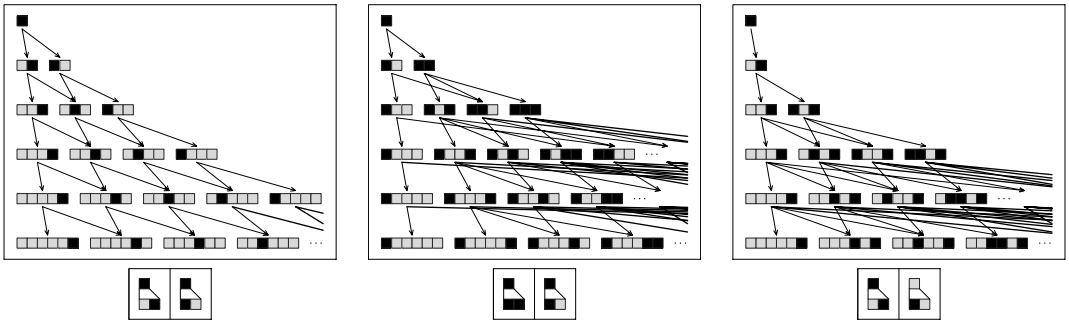
any proof—regardless of length—exists for a specific result in a mathematical system with particular axioms.

So what are the implications of this?

Probably the most striking arise when one tries to apply traditional ideas of logic—and particularly notions of true and false.

The way I have set things up, one can find all the statements that can be proved true in a particular axiom system just by starting with an expression that represents "true" and then using the rules of the axiom system, as in the picture on the facing page.

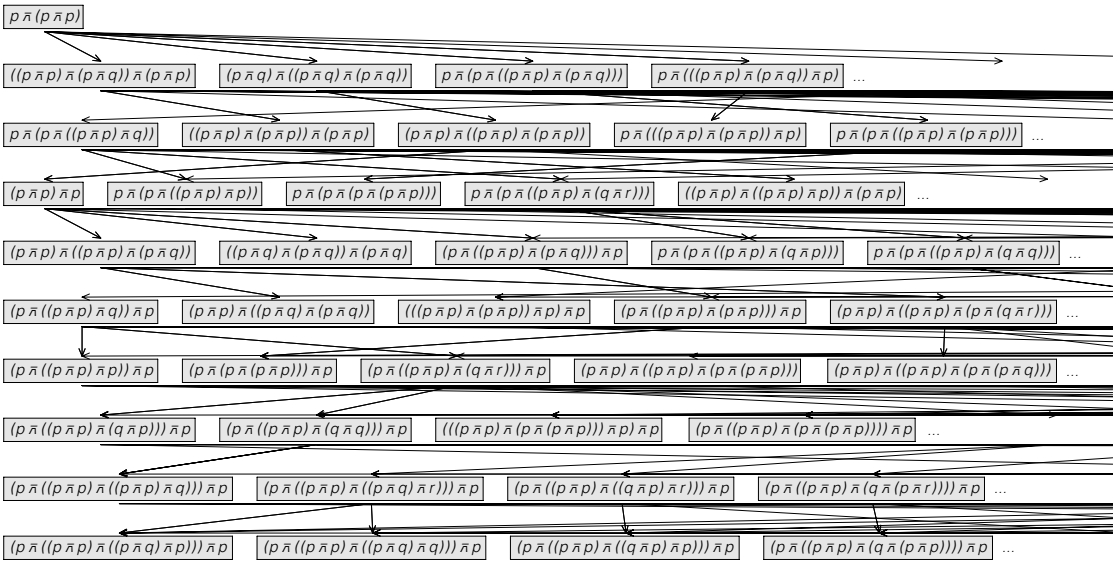
In a multiway system, one can imagine identifying "true" with a string consisting of a single black element. And this would mean that every string in networks like the ones below should correspond to a statement that can be proved true in the axiom system used.



Multiway systems starting from a single black element that represents TRUE. All strings that appear can be thought of as statements that are true according to the axioms represented by the multiway system rules. One can take negation to be the operation that interchanges black and white. This then means that the first multiway system represents an inconsistent axiom system, since on step 2, both □■ and its negation ■□ appear. The other two multiway systems are consistent, so that they never generate both a string and its negation. The third one, however, is incomplete, since for example it never generates either □□ or its negation ■■. The second one, however, is both complete and consistent: it generates all strings that begin with ■, but none that begin with □.

But is this really reasonable? In traditional logic there is always an operation of negation which takes any true statement, and makes it into a false one, and vice versa. And in a multiway system, one possible way negation might work is just to reverse the colors of the elements in a string. But this then leads to a problem in the first picture above.

For the picture implies that both □■ and its negation ■□ can be proved to be true statements. But this cannot be correct. And so what



The network of statements that can be proved true using the axiom system for logic from page 775. $p \barwedge (p \barwedge p)$ is the simplest representation for TRUE when logic is set up using the NAND operator $\barwedge$. Each arrow indicates an equivalence established by applying a single axiom. On each row only statements that have not appeared before are given. The statements are sorted so that the simplest are first. Note that some fairly simple statements do not show up for at least several rows. The total number of statements on successive rows grows faster than exponentially; for the first few it is 1, 6, 91, 2180, 76138. If continued forever the network would eventually include all possible true statements (tautologies) of logic (see also page 818). Other simple axiom systems for logic like those on page 808 yield networks similar to the one shown.

this means is that with the setup used the underlying axiom system is inconsistent. So what about the other multiway systems on the facing page? At least with the strings one can see in the pictures there are no inconsistencies. But what about with longer strings? For the particular rules shown it is fairly easy to demonstrate that there are never inconsistencies. But in general it is not possible to do this, for after some given string has appeared, it can for example be undecidable whether the negation of that particular string ever appears.

So what about the axiom systems normally used in actual mathematics? None of those on pages 773 and 774 appear to be inconsistent. And what this means is that the set of statements that can be proved true will never overlap with the set that can be proved false.

But can every possible statement that one might expect to be true or false actually in the end be proved either true or false?

In the early 1900s it was widely believed that this would effectively be the case in all reasonable mathematical axiom systems. For at the time there seemed to be no limit to the power of mathematics, and no end to the theorems that could be proved.

But this all changed in 1931 when Gödel's Theorem showed that at least in any finitely-specified axiom system containing standard arithmetic there must inevitably be statements that cannot be proved either true or false using the rules of the axiom system.

This was a great shock to existing thinking about the foundations of mathematics. And indeed to this day Gödel's Theorem has continued to be widely regarded as a surprising and rather mysterious result.

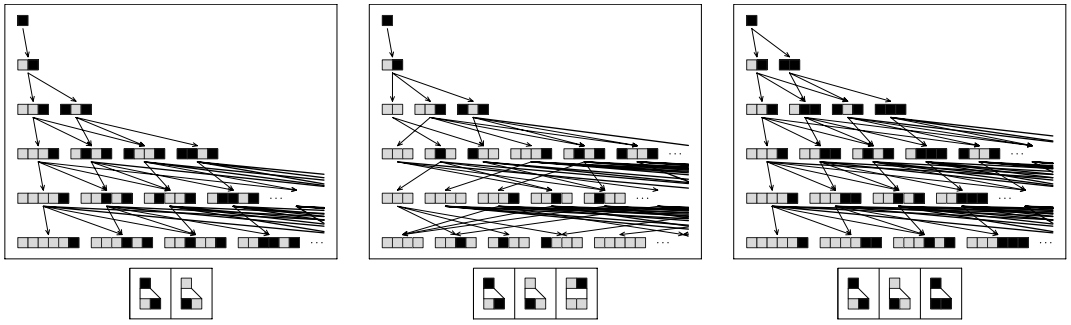
But the discoveries in this book finally begin to make it seem inevitable and actually almost obvious. For it turns out that at some level it can be viewed as just yet another consequence of the very general Principle of Computational Equivalence.

So what is the analog of Gödel's Theorem for multiway systems? Given the setup on page 780 one can ask whether a particular multiway system is complete in the sense that for every possible string the system eventually generates either that string or its negation.

And one can see that in fact the third multiway system is incomplete, since by following its rules one can never for example generate either □□ or its negation ■■. But what if one extends the rules by adding more transformations, corresponding to more axioms? Can one always in the end make the system complete?

If one is not quite careful, one will generate too many strings, and inevitably get inconsistencies where both a string and its negation appear, as in the second picture on the facing page. But at least if one only has to worry about a limited number of steps, it is always possible to set things up so as to get a system that is both complete and consistent, as in the third picture on the facing page.

And in fact in the particular case shown on the facing page it is fairly straightforward to find rules that make the system always complete and consistent. But knowing how to do this requires having behavior that is in a sense simple enough that one can foresee every aspect of it.



The effect of adding transformations to the rules for a multiway system. The first multiway system is incomplete, in the sense that for some strings, it generates neither the string nor its negation. The second multiway system yields more strings—but introduces inconsistency, since it can generate both ■□ and its negation □■. The third multiway system is however both complete and consistent: for every string it eventually generates either that string or its negation.

Yet if a system is computationally irreducible this will inevitably not be possible. For at any point the system will always in effect be able to do more things that one did not expect. And this means that in general one will not be able to construct a finite set of axioms that can be guaranteed to lead to ultimate completeness and consistency.

And in fact it turns out that as soon as the question of whether a particular string can ever be reached is undecidable it immediately follows that there must be either incompleteness or inconsistency. For to say that such a question is undecidable is to say that it cannot in general be answered by any procedure that is guaranteed to finish.

But if one had a system that was complete and consistent then it is easy to come up with such a procedure: one just runs the system until either one reaches the string one is looking for or one reaches its negation. For the completeness of the system guarantees that one must always reach one or the other, while its consistency implies that reaching one allows one to conclude that one will never reach the other.

So the result of this is that if the evolution of a multiway system is computationally irreducible—so that questions about its ultimate behavior are undecidable—the system cannot be both complete and consistent. And if one assumes consistency then it follows that there must be strings where neither the string nor its negation can be

reached—corresponding to the fact that statements must exist that cannot be proved either true or false from a given set of axioms.

But what does it take to establish that such incompleteness will actually occur in a specific system?

The basic way to do it is to show that the system is universal.

But what exactly does universality mean for something like an axiom system? In effect what it means is that any question about the behavior of any other universal system can be encoded as a statement in the axiom system—and if the answer to the question can be established by watching the evolution of the other universal system for any finite number of steps then it must also be able to be established by giving a proof of finite length in the axiom system.

So what axiom systems in mathematics are then universal?

Basic logic is not, since at least in principle one can always determine the truth of any statement in this system by the finite—if perhaps exponentially long—procedure of trying all possible combinations of truth values for the variables that appear in it.

And essentially the same turns out to be the case for pure predicate logic, in which one just formally adds "for all" and "there exists" constructs. But as soon as one also puts in an abstract function or relation with more than one argument, one gets universality.

And indeed the basis for Gödel's Theorem is the result that the standard axioms for basic integer arithmetic support universality.

Set theory and several other standard axiom systems can readily be made to reproduce arithmetic, and are therefore also universal. And the same is true of group theory and other algebraic systems like ring theory.

If one puts enough constraints on the axioms one uses, one can eventually prevent universality—and in fact this happens for commutative group theory, and for the simplified versions of both real algebra and geometry on pages 773 and 774.

But of the axiom systems actually used in current mathematics research every single one is now known to be universal.

From page 773 we can see that many of these axiom systems can be stated in quite simple ways. And in the past it might have seemed

hard to believe that systems this simple could ever be universal, and thus in a sense be able to emulate essentially any system.

But from the discoveries in this book this now seems almost inevitable. And indeed the Principle of Computational Equivalence implies that beyond some low threshold almost any axiom system should be expected to be universal.

So how does universality actually work in the case of arithmetic?

One approach is illustrated in the picture on the next page. The idea is to set up an arithmetic statement that can be proved true if the evolution of a cellular automaton from a given initial condition makes a given cell be a given color at a given step, and can be proved false if it does not.

By changing numbers in this arithmetic statement one can then in effect sample different aspects of the cellular automaton evolution. And with the cellular automaton being a universal one such as rule 110 this implies that the axioms of arithmetic can support universality.

Such universality then implies Gödel's Theorem and shows that there must exist statements about arithmetic that cannot ever be proved true or false from its normal axioms.

So what are some examples of such statements?

The original proof of Gödel's Theorem was based on considering the particular self-referential statement "this statement is unprovable".

At first it does not seem obvious that such a statement could ever be set up as a statement in arithmetic. But if it could then one can see that it would immediately follow that—as the statement says—it cannot be proved, since otherwise there would be an inconsistency.

And in fact the main technical difficulty in the original proof of Gödel's Theorem had to do with showing—by doing what amounted to establishing the universality of arithmetic—that the statement could indeed meaningfully be encoded as a statement purely in arithmetic.

But at least with the original encoding used, the statement would be astronomically long if written out in the notation of page 773. And from this result, one might imagine that unprovability would never be relevant in any practical situation in mathematics.

But does one really need to have such a complicated statement in order for it to be unprovable from the axioms of arithmetic?

$$(-3x_6 + x_7 + x_8)^2 + (2^{1+x_3(1+x_1+2x_3)}x_2 - 2x_4 - x_{10} + x_{11})^2 + (-2x_8 - x_9 + x_{10} + x_{11})^2 + (1 - 2^{(1+x_3)(x_1+2x_3)} + x_4 + x_{12})^2 +$$
$$(1 - 2^{x_1} + x_2 + x_{13})^2 + (1 - 2^{x_1} + x_5 + x_{14})^2 + (-x_4 + 2^{x_3}x_5 + 2^{x_1+2x_3}x_6 + 2^{x_1+x_3}x_{15} + x_{16})^2 + (1 - 2^{x_3} + x_{15} + x_{17})^2 +$$
$$(1 - 2^{x_3} + x_{16} + x_{18})^2 + (-x_6 - 2x_7 + x_9 + x_{19})^2 + (-(2 + 2^{x_6})^{x_6} + (1 + 2^{x_6})^{x_7}(1 + 2x_{20} + (1 + 2^{x_6})x_{21}) + x_{22})^2 + (1 - (1 + 2^{x_6})^{x_7} + x_{22} + x_{23})^2 +$$
$$(1 - 2^{x_6} + 2x_{20} + x_{24})^2 + (-(2 + 4^{x_6})^{2x_6} + (1 + 4^{x_6})^{x_7}(1 + 2x_{25} + (1 + 4^{x_6})x_{26}) + x_{27})^2 + (1 - (1 + 4^{x_6})^{x_7} + x_{27} + x_{28})^2 +$$
$$(1 - 4^{x_6} + 2x_{25} + x_{29})^2 + (-(2 + 2^{x_8})^{x_8} + (1 + 2^{x_8})^{x_6}(1 + 2x_{30} + (1 + 2^{x_8})x_{31}) + x_{32})^2 + (1 - (1 + 2^{x_8})^{x_6} + x_{32} + x_{33})^2 +$$
$$(1 - 2^{x_8} + 2x_{30} + x_{34})^2 + (-(2 + 2^{x_8})^{x_8} + (1 + 2^{x_8})^{2x_6}(1 + 2x_{35} + (1 + 2^{x_8})x_{36}) + x_{37})^2 + (1 - (1 + 2^{x_8})^{2x_6} + x_{37} + x_{38})^2 +$$
$$(1 - 2^{x_8} + 2x_{35} + x_{39})^2 + (-(2 + 2^{x_6})^{x_6} + (1 + 2^{x_6})^{x_9}(1 + 2x_{40} + (1 + 2^{x_6})x_{41}) + x_{42})^2 + (1 - (1 + 2^{x_6})^{x_9} + x_{42} + x_{43})^2 +$$
$$(1 - 2^{x_6} + 2x_{40} + x_{44})^2 + (-(2 + 4^{x_7})^{2x_7} + (1 + 4^{x_7})^{x_9}(1 + 2x_{45} + (1 + 4^{x_7})x_{46}) + x_{47})^2 + (1 - (1 + 4^{x_7})^{x_9} + x_{47} + x_{48})^2 +$$
$$(1 - 4^{x_7} + 2x_{45} + x_{49})^2 + (-(2 + 2^{x_{19}})^{x_{19}} + (1 + 2^{x_{19}})^{x_6}(1 + 2x_{50} + (1 + 2^{x_{19}})x_{51}) + x_{52})^2 + (1 - (1 + 2^{x_{19}})^{x_6} + x_{52} + x_{53})^2 +$$
$$(1 - 2^{x_{19}} + 2x_{50} + x_{54})^2 + (-(2 + 2^{x_{19}})^{x_{19}} + (1 + 2^{x_{19}})^{2x_7}(1 + 2x_{55} + (1 + 2^{x_{19}})x_{56}) + x_{57})^2 + (1 - (1 + 2^{x_{19}})^{2x_7} + x_{57} + x_{58})^2 +$$
$$(1 - 2^{x_{19}} + 2x_{55} + x_{59})^2 + (-(2 + 2^{x_9})^{x_9} + (1 + 2^{x_9})^{x_{10}}(1 + 2x_{60} + (1 + 2^{x_9})x_{61}) + x_{62})^2 + (1 - (1 + 2^{x_9})^{x_{10}} + x_{62} + x_{63})^2 + (1 - 2^{x_9} + 2x_{60} + x_{64})^2 +$$
$$(-(2 + 4^{x_8})^{2x_8} + (1 + 4^{x_8})^{x_{10}}(1 + 2x_{65} + (1 + 4^{x_8})x_{66}) + x_{67})^2 + (1 - (1 + 4^{x_8})^{x_{10}} + x_{67} + x_{68})^2 + (1 - 4^{x_8} + 2x_{65} + x_{69})^2 +$$
$$(-(2 + 2^{x_{11}})^{x_{11}} + (1 + 2^{x_{11}})^{x_9}(1 + 2x_{70} + (1 + 2^{x_{11}})x_{71}) + x_{72})^2 + (1 - (1 + 2^{x_{11}})^{x_9} + x_{72} + x_{73})^2 + (1 - 2^{x_{11}} + 2x_{70} + x_{74})^2 +$$
$$(-(2 + 2^{x_{11}})^{x_{11}} + (1 + 2^{x_{11}})^{2x_8}(1 + 2x_{75} + (1 + 2^{x_{11}})x_{76}) + x_{77})^2 + (1 - (1 + 2^{x_{11}})^{2x_8} + x_{77} + x_{78})^2 + (1 - 2^{x_{11}} + 2x_{75} + x_{79})^2 = 0$$

x_1 (initial width)	1
x_2 (initial state)	1
x_3 (steps)	1
x_4 (evolution)	22
x_5	1
x_6	2
x_7	0
x_8	6
x_9	0
x_{10}	0
x_{11}	12
x_{12}	41
x_{13}	0
x_{14}	0
x_{15}	1
⋮	

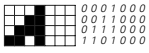
$x_4 = 22 =$

x_1 (initial width)	1
x_2 (initial state)	1
x_3 (steps)	2
x_4 (evolution)	4508
x_5	1
x_6	140
x_7	8
x_8	412
x_9	0
x_{10}	0
x_{11}	824
x_{12}	28259
x_{13}	0
x_{14}	0
x_{15}	3
⋮	

$x_4 = 4508 =$

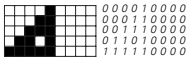
x_1 (initial width)	1
x_2 (initial state)	1
x_3 (steps)	3
x_4 (evolution)	17177704
x_5	1
x_6	134200
x_7	2096
x_8	400504
x_9	32
x_{10}	32
x_{11}	801008
x_{12}	251257751
x_{13}	0
x_{14}	0
x_{15}	6
⋮	

$x_4 = 17177704 =$



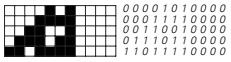
x_1 (initial width)	1
x_2 (initial state)	1
x_3 (steps)	4
x_4 (evolution)	1105983545840
x_5	1
x_6	2160124112
x_7	8437888
x_8	6471934448
x_9	32768
x_{10}	32768
x_{11}	12943868896
x_{12}	34078388542991
x_{13}	0
x_{14}	0
x_{15}	15
⋮	

$x_4 = 1105983545840 =$



x_1 (initial width)	3
x_2 (initial state)	5
x_3 (steps)	4
x_4 (evolution)	1409438147512048
x_5	7
x_6	688202220464
x_7	940049184
x_8	2063666612208
x_9	805306880
x_{10}	805306880
x_{11}	4127333224416
x_{12}	34619358871451919
x_{13}	2
x_{14}	0
x_{15}	13
⋮	

$x_4 = 1409438147512048 =$



Universality in arithmetic, illustrated by an integer equation whose solutions in effect emulate the rule 110 universal cellular automaton from Chapter 11. The equation has many solutions, but all of them satisfy the constraint that the variables x_1 through x_4 must encode possible initial conditions and evolution histories for rule 110. If one fills in fixed values for x_1, x_2 and x_3, then only one value for x_4 is ever possible—corresponding to the evolution history of rule 110 for x_3 steps starting from a width x_1 initial condition given by the digit sequence of x_2. In general any statement about the possible behavior of rule 110 can be encoded as a statement in arithmetic about solutions to the equation. So for example if one fills in values for x_1, x_2 and x_4, but not x_3, then the statement that the equation has no solution for any x_3 corresponds to a statement that rule 110 can never exhibit certain behavior, even after any number of steps. But the universality of rule 110 implies that such statements must in general be undecidable. So from this it follows that in at least some instances the axioms of arithmetic can never be used to give a finite proof of whether or not the statement is true. The construction shown here can be viewed as providing a simple proof of Gödel's Theorem on the existence of unprovable statements in arithmetic. Note that the equation shown is a so-called exponential Diophantine one, in which some variables appear in exponents. At the cost of considerably more complication—and using for example 2154 variables—it is possible to avoid this. The equation above can however already be viewed as capturing the essence of what is needed to demonstrate the general unsolvability of Diophantine equations and Hilbert's Tenth Problem.

Over the past seventy years a few simpler examples have been constructed—mostly with no obviously self-referential character.

But usually these examples have involved rather sophisticated and obscure mathematical constructs—most often functions that are somehow set up to grow extremely rapidly. Yet at least in principle there should be examples that can be constructed based just on statements that no solutions exist to particular integer equations.

If an integer equation such as $x^2 = y^3 + 12$ has a definite solution such as $x = 47,\ y = 13$ in terms of particular finite integers then this fact can certainly be proved using the axioms of arithmetic. For it takes only a finite calculation to check the solution, and this very calculation can always in effect be thought of as a proof.

But what if the equation has no solutions? To test this explicitly one would have to look at an infinite number of possible integers. But the point is that even so, there can still potentially be a finite mathematical proof that none of these integers will work.

And sometimes the proof may be straightforward—say being based on showing that one side of the equation is always odd while the other is always even. In other cases the proof may be more difficult—say being based on establishing some large maximum size for a solution, then checking all integers up to that size.

And the point is that in general there may in fact be absolutely no proof that can be given in terms of the normal axioms of arithmetic.

So how can one see this?

The picture on the facing page shows that one can construct an integer equation whose solutions represent the behavior of a system like a cellular automaton. And the way this works is that for example one variable in the equation gives the number of steps of evolution, while another gives the outcome after that number of steps.

So with this setup, one can specify the number of steps, then solve for the outcome after that number of steps. But what if for example one instead specifies an outcome, then tries to find a solution for the number of steps at which this outcome occurs?

If in general one was able to tell whether such a solution exists then it would mean that one could always answer the question of

whether, say, a particular pattern would ever die out in the evolution of a given cellular automaton. But from the discussion of the previous section we know that this in general is undecidable.

So it follows that it must be undecidable whether a given integer equation of some particular general form has a solution. And from the arguments above this in turn implies that there must be specific integer equations that have no solutions but where this fact cannot be proved from the normal axioms of arithmetic.

So how ultimately can this happen?

At some level it is a consequence of the involvement of infinity. For at least in a universal system like arithmetic any question that is entirely finite can in the end always be answered by a finite procedure.

But what about questions that somehow ask, say, about infinite numbers of possible integers? To have a finite way to address questions like these is often in the end the main justification for setting up typical mathematical axiom systems in the first place.

For the point is that instead of handling objects like integers directly, axiom systems can just give abstract rules for manipulating statements about them. And within such statements one can refer, say, to infinite sets of integers just by a symbol like s.

And particularly over the past century there have been many successes in mathematics that can be attributed to this basic kind of approach. But the remarkable fact that follows from Gödel's Theorem is that whatever one does there will always be cases where the approach must ultimately fail. And it turns out that the reason for this is essentially the phenomenon of computational irreducibility.

For while simple infinite quantities like $1/0$ or the total number of integers can readily be summarized in finite ways—often just by using symbols like ∞ and $\aleph_0$—the same is not in general true of all infinite processes. And in particular if an infinite process is computationally irreducible then there cannot in general be any useful finite summary of what it does—since the existence of such a summary would imply computational reducibility.

So among other things this means that there will inevitably be questions that finite proofs based on axioms that operate within ordinary computational systems will never in general be able to answer.

And indeed with integer equations, as soon as one has a general equation that is universal, it typically follows that there will be specific instances in which the absence of solutions—or at least of solutions of some particular kind—can never be proved on the basis of the normal axioms of arithmetic.

For several decades it has been known that universal integer equations exist. But the examples that have actually been constructed are quite complicated—like the one on page 786—with the simplest involving 9 variables and an immense number of terms.

Yet from the discoveries in this book I am quite certain that there are vastly simpler examples that exist—so that in fact there are in the end rather simple integer equations for which the absence of solutions can never be proved from the normal axioms of arithmetic.

If one just starts looking at sequences of integer equations—as on the next page—then in the very simplest cases it is usually fairly easy to tell whether a particular equation will have any solutions. But this rapidly becomes very much more difficult. For there is often no obvious pattern to which equations ultimately have solutions and which do not. And even when equations do have solutions, the integers involved can be quite large. So, for example, the smallest solution to $x^2 = 61\, y^2 + 1$ is $x = 1766319049$, $y = 226153980$, while the smallest solution to $x^3 + y^3 = z^3 + 2$ is $x = 1214928$, $y = 3480205$, $z = 3528875$.

Integer equations such as $a\, x + b\, y + c\, z = d$ that have only linear dependence on any variable were largely understood even in antiquity. Quadratic equations in two variables such as $x^2 = a\, y^2 + b$ were understood by the 1800s. But even equations such as $x^2 = a\, y^3 + b$ were not properly understood until the 1980s. And with equations that have higher powers or more variables questions of whether solutions exist quickly end up being unsolved problems of number theory.

It has certainly been known for centuries that there are questions about integer equations and other aspects of number theory that are easy to state, yet seem very hard to answer. But in practice it has almost

$2x+3y=1$ □	$x^2=y^3-20$ $x=14$ $y=6$	$x^3=y^4-20xy-1$ $x=10$ $y=7$	$x^3+y^3=z^2+1$ $x=1$ $y=1$ $z=1$
$2x+3y=2$ □	$x^2=y^3-19$ $x=18$ $y=7$	$x^3=y^4-19xy-1$ $x=3$ $y=4$	$x^3+y^3=z^2+2$ $x=107$ $y=232$ $z=3703$
$2x+3y=3$ □	$x^2=y^3-18$ $x=3$ $y=3$	$x^3=y^4-18xy-1$ $x=75$ $y=26$	$x^3+y^3=z^2+3$ $x=1$ $y=3$ $z=5$
$2x+3y=4$ □	$x^2=y^3-17$ □	$x^3=y^4-17xy-1$	$x^3+y^3=z^2+4$ $x=5$ $y=12$ $z=43$
$2x+3y=5$ $x=1$ $y=1$	$x^2=y^3-16$ □	$x^3=y^4-16xy-1$	$x^3+y^3=z^2+5$ $x=1$ $y=2$ $z=2$
$2x+3y=6$ □	$x^2=y^3-15$ $x=7$ $y=4$	$x^3=y^4-15xy-1$ $x=624$ $y=125$	$x^3+y^3=z^2+6$ $x=7$ $y=24$ $z=119$
$2x+3y=7$ $x=2$ $y=1$	$x^2=y^3-14$ □	$x^3=y^4-14xy-1$	$x^3+y^3=z^2+7$ $x=2$ $y=2$ $z=3$
$2x+3y=8$ $x=1$ $y=2$	$x^2=y^3-13$ $x=70$ $y=17$	$x^3=y^4-13xy-1$	$x^3+y^3=z^2+8$ $x=1$ $y=2$ $z=1$
$2x+3y=9$ $x=3$ $y=1$	$x^2=y^3-12$ □	$x^3=y^4-12xy-1$ $x=3$ $y=2$	$x^3+y^3=z^2+9$ $x=3$ $y=7$ $z=19$
$2x+3y=10$ $x=2$ $y=2$	$x^2=y^3-11$ $x=4$ $y=3$	$x^3=y^4-11xy-1$	$x^3+y^3=z^2+10$ $x=2$ $y=3$ $z=5$
$2x+3y=11$ $x=1$ $y=3$	$x^2=y^3-10$ □	$x^3=y^4-10xy-1$	
$2x+3y=12$ $x=3$ $y=2$	$x^2=y^3-9$ □	$x^3=y^4-9xy-1$ $x=80$ $y=27$	$x^3+y^3=z^3-20$ $x=107$ $y=137$ $z=156$
$2x+3y=13$ $x=2$ $y=3$	$x^2=y^3-8$ □	$x^3=y^4-8xy-1$ $x=12$ $y=7$	$x^3+y^3=z^3-19$ $x=14$ $y=16$ $z=19$
$2x+3y=14$ $x=1$ $y=4$	$x^2=y^3-7$ $x=1$ $y=2$	$x^3=y^4-7xy-1$ $x=1$ $y=2$	$x^3+y^3=z^3-18$ $x=1$ $y=2$ $z=3$
$2x+3y=15$ $x=3$ $y=3$	$x^2=y^3-6$ □	$x^3=y^4-6xy-1$ $x=15$ $y=8$	$x^3+y^3=z^3-17$ $x=103$ $y=111$ $z=135$
	$x^2=y^3-5$ □	$x^3=y^4-5xy-1$	$x^3+y^3=z^3-16$ $x=10$ $y=12$ $z=14$
$x^2=y^2+1$ □	$x^2=y^3-4$ $x=2$ $y=2$	$x^3=y^4-4xy-1$ $x=30$ $y=13$	$x^3+y^3=z^3-15$ $x=262$ $y=265$ $z=332$
$x^2=y^2+2$ □	$x^2=y^3-3$ □	$x^3=y^4-3xy-1$	$x^3+y^3=z^3-14$ □
$x^2=y^2+3$ $x=2$ $y=1$	$x^2=y^3-2$ $x=5$ $y=3$	$x^3=y^4-2xy-1$	$x^3+y^3=z^3-13$ □
$x^2=y^2+4$ □	$x^2=y^3-1$ □	$x^3=y^4-xy-1$	$x^3+y^3=z^3-12$ $x=5725013$ $y=9019406$ $z=9730705$
$x^2=y^2+5$ $x=3$ $y=2$	$x^2=y^3$ $x=1$ $y=1$	$x^3=y^4-1$ □	$x^3+y^3=z^3-11$ $x=2$ $y=2$ $z=3$
$x^2=y^2+6$ □	$x^2=y^3+1$ $x=3$ $y=2$	$x^3=y^4+xy-1$ $x=1$ $y=1$	$x^3+y^3=z^3-10$ $x=3$ $y=3$ $z=4$
$x^2=y^2+7$ $x=4$ $y=3$	$x^2=y^3+2$ □	$x^3=y^4+2xy-1$ $x=3$ $y=2$	$x^3+y^3=z^3-9$ $x=52$ $y=216$ $z=217$
$x^2=y^2+8$ $x=3$ $y=1$	$x^2=y^3+3$ $x=2$ $y=1$	$x^3=y^4+3xy-1$ $x=5$ $y=3$	$x^3+y^3=z^3-8$ $x=16$ $y=12$ $z=18$
$x^2=y^2+9$ $x=5$ $y=4$	$x^2=y^3+4$ □	$x^3=y^4+4xy-1$ $x=2$ $y=1$	$x^3+y^3=z^3-7$ $x=605809$ $y=680316$ $z=812918$
$x^2=y^2+10$ □	$x^2=y^3+5$ □	$x^3=y^4+5xy-1$	$x^3+y^3=z^3-6$ $x=1$ $y=1$ $z=2$
$x^2=y^2+11$ $x=6$ $y=5$	$x^2=y^3+6$ □	$x^3=y^4+6xy-1$	$x^3+y^3=z^3-5$ □
$x^2=y^2+12$ $x=4$ $y=2$	$x^2=y^3+7$ □	$x^3=y^4+7xy-1$	$x^3+y^3=z^3-4$ □
$x^2=y^2+13$ $x=7$ $y=6$	$x^2=y^3+8$ $x=3$ $y=1$	$x^3=y^4+8xy-1$ $x=20$ $y=9$	$x^3+y^3=z^3-3$
$x^2=y^2+14$ □	$x^2=y^3+9$ $x=6$ $y=3$	$x^3=y^4+9xy-1$ $x=3$ $y=1$	$x^3+y^3=z^3-2$ $x=5$ $y=6$ $z=7$
$x^2=y^2+15$ $x=4$ $y=1$	$x^2=y^3+10$ □	$x^3=y^4+10xy-1$	$x^3+y^3=z^3-1$ $x=6$ $y=8$ $z=9$
$x^2=y^2+16$ $x=5$ $y=3$	$x^2=y^3+11$ □	$x^3=y^4+11xy-1$ $x=5$ $y=2$	$x^3+y^3=z^3$ □
	$x^2=y^3+12$ $x=47$ $y=13$	$x^3=y^4+12xy-1$	$x^3+y^3=z^3+1$ $x=1$ $y=2$ $z=2$
$x^2=y^2+1$ □	$x^2=y^3+13$ □	$x^3=y^4+13xy-1$	$x^3+y^3=z^3+2$ $x=1214928$ $y=3480205$ $z=3528875$
$x^2=2y^2+1$ $x=3$ $y=2$	$x^2=y^3+14$ □	$x^3=y^4+14xy-1$	$x^3+y^3=z^3+3$ $x=4$ $y=4$ $z=5$
$x^2=3y^2+1$ $x=2$ $y=1$	$x^2=y^3+15$ $x=4$ $y=1$	$x^3=y^4+15xy-1$	$x^3+y^3=z^3+4$ □
$x^2=4y^2+1$ □	$x^2=y^3+16$ □	$x^3=y^4+16xy-1$ $x=4$ $y=1$	$x^3+y^3=z^3+5$ □
$x^2=5y^2+1$ $x=9$ $y=4$	$x^2=y^3+17$ $x=5$ $y=2$	$x^3=y^4+17xy-1$	$x^3+y^3=z^3+6$ $x=10529$ $y=60248$ $z=60355$
$x^2=6y^2+1$ $x=5$ $y=2$	$x^2=y^3+18$ $x=19$ $y=7$	$x^3=y^4+18xy-1$ $x=8$ $y=3$	$x^3+y^3=z^3+7$ $x=32$ $y=104$ $z=105$
$x^2=7y^2+1$ $x=8$ $y=3$	$x^2=y^3+19$ $x=12$ $y=5$	$x^3=y^4+19xy-1$	$x^3+y^3=z^3+8$ $x=1$ $y=2$ $z=1$
$x^2=8y^2+1$ $x=3$ $y=1$	$x^2=y^3+20$ □	$x^3=y^4+20xy-1$	$x^3+y^3=z^3+9$ $x=2097$ $y=11305$ $z=11329$
$x^2=9y^2+1$ □			$x^3+y^3=z^3+10$ $x=130$ $y=141$ $z=171$
$x^2=10y^2+1$ $x=19$ $y=6$	$x^2=y^3+1$ $x=3$ $y=2$	$x^2=y^5+3$ $x=2$ $y=1$	$x^3+y^3=z^3+11$ $x=297$ $y=619$ $z=641$
$x^2=11y^2+1$ $x=10$ $y=3$	$x^2=2y^3+1$ □	$x^2=y^5+y+3$ $x=2537$ $y=23$	$x^3+y^3=z^3+12$ $x=7$ $y=10$ $z=11$
$x^2=12y^2+1$ $x=7$ $y=2$	$x^2=3y^3+1$ $x=2$ $y=1$	$x^2=y^5+2y+3$	$x^3+y^3=z^3+13$ □
$x^2=13y^2+1$ $x=649$ $y=180$	$x^2=4y^3+1$ □	$x^2=y^5+3y+3$	$x^3+y^3=z^3+14$ □
$x^2=14y^2+1$ $x=15$ $y=4$	$x^2=5y^3+1$ □	$x^2=y^5+4y+3$	$x^3+y^3=z^3+15$ $x=2$ $y=2$ $z=1$
$x^2=15y^2+1$ $x=4$ $y=1$	$x^2=6y^3+1$ $x=7$ $y=2$	$x^2=y^5+5y+3$ $x=3$ $y=1$	$x^3+y^3=z^3+16$ $x=2429856$ $y=6960410$ $z=7057750$
$x^2=16y^2+1$ □	$x^2=7y^3+1$ □	$x^2=y^5+6y+3$	$x^3+y^3=z^3+17$ $x=25$ $y=50$ $z=52$
$x^2=17y^2+1$ $x=33$ $y=8$	$x^2=8y^3+1$ $x=3$ $y=1$	$x^2=y^5+7y+3$ $x=7$ $y=2$	$x^3+y^3=z^3+18$ $x=94$ $y=101$ $z=123$
$x^2=18y^2+1$ $x=17$ $y=4$	$x^2=9y^3+1$ □	$x^2=y^5+8y+3$	$x^3+y^3=z^3+19$ $x=26$ $y=76$ $z=77$
$x^2=19y^2+1$ $x=170$ $y=39$			
$x^2=20y^2+1$ $x=9$ $y=2$	$x^2=10y^3+1$ $x=9$ $y=2$	$x^2=y^5+9y+3$	$x^3+y^3=z^3+20$ $x=1$ $y=3$ $z=2$

universally been assumed that with the continued development of mathematics any of these questions could in the end be answered.

However, what Gödel's Theorem shows is that there must always exist some questions that cannot ever be answered using the normal axioms of arithmetic. Yet the fact that the few known explicit examples have been extremely complicated has made this seem somehow fundamentally irrelevant for the actual practice of mathematics.

But from the discoveries in this book it now seems quite certain that vastly simpler examples also exist. And it is my strong suspicion that in fact of all the current unsolved problems seriously studied in number theory a fair fraction will in the end turn out to be questions that cannot ever be answered using the normal axioms of arithmetic.

If one looks at recent work in number theory, most of it tends to be based on rather sophisticated methods that do not obviously depend only on the normal axioms of arithmetic. And for example the elaborate proof of Fermat's Last Theorem that has been developed may make at least some use of axioms that come from fields like set theory and go beyond the normal ones for arithmetic.

But so long as one stays within, say, the standard axiom systems of mathematics on pages 773 and 774, and does not in effect just end up implicitly adding as an axiom whatever result one is trying to prove, my strong suspicion is that one will ultimately never be able to go much further than one can purely with the normal axioms of arithmetic.

And indeed from the Principle of Computational Equivalence I strongly believe that in general undecidability and unprovability will start to occur in practically any area of mathematics almost as soon as one goes beyond the level of questions that are always easy to answer.

But if this is so, why then has mathematics managed to get as far as it has? Certainly there are problems in mathematics that have remained unsolved for long periods of time. And I suspect that many of these will in fact in the end turn out to involve undecidability and

◀ Smallest solutions for various sequences of integer (or so-called Diophantine) equations. □ indicates that it can be proved that no solution exists. A blank indicates that I know only that no solution exists below a billion. Methods for resolving some of the equations in the first column were known in antiquity; all had been resolved by the 1800s. Practical methods for resolving the so-called elliptic curve equations in the second column were developed only in the 1980s. No general methods are yet known for most of the other equations given—and some classes of them may in fact show undecidability.

unprovability. But the issue remains why such phenomena have not been much more obvious in everyday work in mathematics.

At some level I suspect the reason is quite straightforward: it is that like most other fields of human inquiry mathematics has tended to define itself to be concerned with just those questions that its methods can successfully address. And since the main methods traditionally used in mathematics have revolved around doing proofs, questions that involve undecidability and unprovability have inevitably been avoided.

But can this really be right? For at least in the past century mathematics has consistently given the impression that it is concerned with questions that are somehow as arbitrary and general as possible.

But one of the important conclusions from what I have done in this book is that this is far from correct. And indeed for example traditional mathematics has for the most part never even considered most of the kinds of systems that I discuss in this book—even though they are based on some of the very simplest rules possible.

So how has this happened? The main point, I believe, is that in both the systems it studies and the questions it asks mathematics is much more a product of its history than is usually realized.

And in fact particularly compared to what I do in this book the vast majority of mathematics practiced today still seems to follow remarkably closely the traditions of arithmetic and geometry that already existed even in Babylonian times.

It is a fairly recent notion that mathematics should even try to address arbitrary or general systems. For until not much more than a century ago mathematics viewed itself essentially just as providing a precise formulation of certain aspects of everyday experience—mainly those related to number and space.

But in the 1800s, with developments such as non-Euclidean geometry, quaternions, group theory and transfinite numbers it began to be assumed that the discipline of mathematics could successfully be applied to any abstract system, however arbitrary or general.

Yet if one looks at the types of systems that are actually studied in mathematics they continue even to this day to be far from as general as possible. Indeed at some level most of them can be viewed as having

been arrived at by the single rather specific approach of starting from some known set of theorems, then trying to find systems that are progressively more general, yet still manage to satisfy these theorems.

And given this approach, it tends to be the case that the questions that are considered interesting are ones that revolve around whatever theorems a system was set up to satisfy—making it rather likely that these questions can themselves be addressed by similar theorems, without any confrontation with undecidability or unprovability.

But what if one looks at other kinds of systems?

One of the main things I have done in this book is in a sense to introduce a new approach to generalization in which one considers systems that have simple but completely arbitrary rules—and that are not set up with any constraint about what theorems they should satisfy.

But if one has such a system, how does one decide what questions are interesting to ask about it? Without the guidance of known theorems, the obvious thing to do is just to look explicitly at how the system behaves—perhaps by making some kind of picture.

And if one does this, then what I have found is that one is usually immediately led to ask questions that run into phenomena like undecidability. Indeed, from my experiments it seems that almost as soon as one leaves behind the constraints of mathematical tradition undecidability and unprovability become rather common.

As the picture on the next page indicates, it is quite straightforward to set up an axiom system that deals with logical statements about a system like a cellular automaton. And within such an axiom system one can ask questions such as whether the cellular automaton will ever behave in a particular way after any number of steps.

But as we saw in the previous section, such questions are in general undecidable. And what this means is that there will inevitably be cases of them for which no proof of a particular answer can ever be given within whatever axiom system one is using.

So from this one might conclude that as soon as one looks at cellular automata or other kinds of systems beyond those normally studied in mathematics it must immediately become effectively impossible to make progress using traditional mathematical methods.

$\langle a, b \diamond c, d\rangle == \langle a, b, c\rangle \diamond \langle b, c, d\rangle$
$\langle a \diamond b, c, d\rangle == \langle b, c, d\rangle$
$\langle a, b, c \diamond d\rangle == \langle a, b, c\rangle$
$a \diamond (b \diamond c) == (a \diamond b) \diamond c$

basic axioms

$\langle \blacksquare, \blacksquare, \blacksquare\rangle == \square$
$\langle \blacksquare, \blacksquare, \square\rangle == \blacksquare$
$\langle \blacksquare, \square, \blacksquare\rangle == \blacksquare$
$\langle \blacksquare, \square, \square\rangle == \square$
$\langle \square, \blacksquare, \blacksquare\rangle == \blacksquare$
$\langle \square, \blacksquare, \square\rangle == \blacksquare$
$\langle \square, \square, \blacksquare\rangle == \blacksquare$
$\langle \square, \square, \square\rangle == \square$

rule 110 axioms

$(\downarrow a) \longrightarrow \langle \square, (\square \diamond a) \diamond \square, \square\rangle$

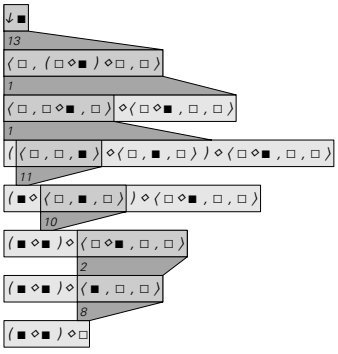
definition

$O\,a \Leftrightarrow O\,((\square \diamond a) \diamond \square)$
$O\,a \Rightarrow O \downarrow a$

advanced axioms

$\blacksquare$	*black cell*
$\square$	*white cell*
$\diamond$	*concatenation*
$\langle\rangle$	*update*
$\downarrow$	*evolution step*
O	*state occurs*

typical interpretations



$\downarrow \blacksquare == (\blacksquare \diamond \blacksquare) \diamond \square$
$\downarrow \downarrow \blacksquare == \downarrow ((\blacksquare \diamond \blacksquare) \diamond \square)$
$\downarrow \downarrow \blacksquare == (\blacksquare \diamond ((\blacksquare \diamond \blacksquare) \diamond \square)) \diamond \square$
$O\,\blacksquare \Rightarrow O\,((\blacksquare \diamond \blacksquare) \diamond \square)$
$O\,\blacksquare \Rightarrow O\,((\blacksquare \diamond ((\blacksquare \diamond \blacksquare) \diamond \square)) \diamond \square)$
$O\,\blacksquare \Rightarrow \exists_a \exists_b O\,(a \diamond (((\blacksquare \diamond ((\blacksquare \diamond \blacksquare) \diamond \square)) \diamond \square) \diamond b))$

provable statements

$O\,\blacksquare \Rightarrow \exists_a \exists_b O\,(a \diamond (((\square \diamond ((\blacksquare \diamond \square) \diamond \blacksquare)) \diamond \square) \diamond b))$

unprovable statement

An axiom system for statements about the rule 110 cellular automaton. The top statement above makes the assertion that the outcome after one step of evolution from a single black cell has a particular form. A proof of this statement is shown to the left. All the statements in the top block above can be proved true from the axiom system. The statement at the bottom, however, cannot be proved either true or false. The axioms given are set up using predicate logic.

But in fact, in the fifteen years or so since I first emphasized the importance of cellular automata all sorts of traditional mathematical work has actually been done on them. So how has this been possible?

The basic point is that the work has tended to concentrate on particular aspects of cellular automata that are simple enough to avoid undecidability and unprovability. And typically it has achieved this in one of two ways: either by considering only very specific cases that have been observed or constructed to be simple, or by looking at things in so much generality that only rather simple properties ever survive.

So for example when presented with the 256 elementary cellular automaton patterns shown on page 55 mathematicians in my experience have two common responses: either to single out specific patterns that have a simple repetitive or perhaps nested form, or to generalize and look not at individual patterns, but rather at aggregate properties obtained say by evolving from all possible initial conditions.

And about questions that concern, for example, the structure of a pattern that looks to us complex, the almost universal reaction is that such questions can somehow not be of any real mathematical interest.

Needless to say, in the framework of the new kind of science in this book, such questions are now of great interest. And my results

suggest that if one is ever going to study many important phenomena that occur in nature one will also inevitably run into them. But to traditional mathematics they seem uninteresting and quite alien.

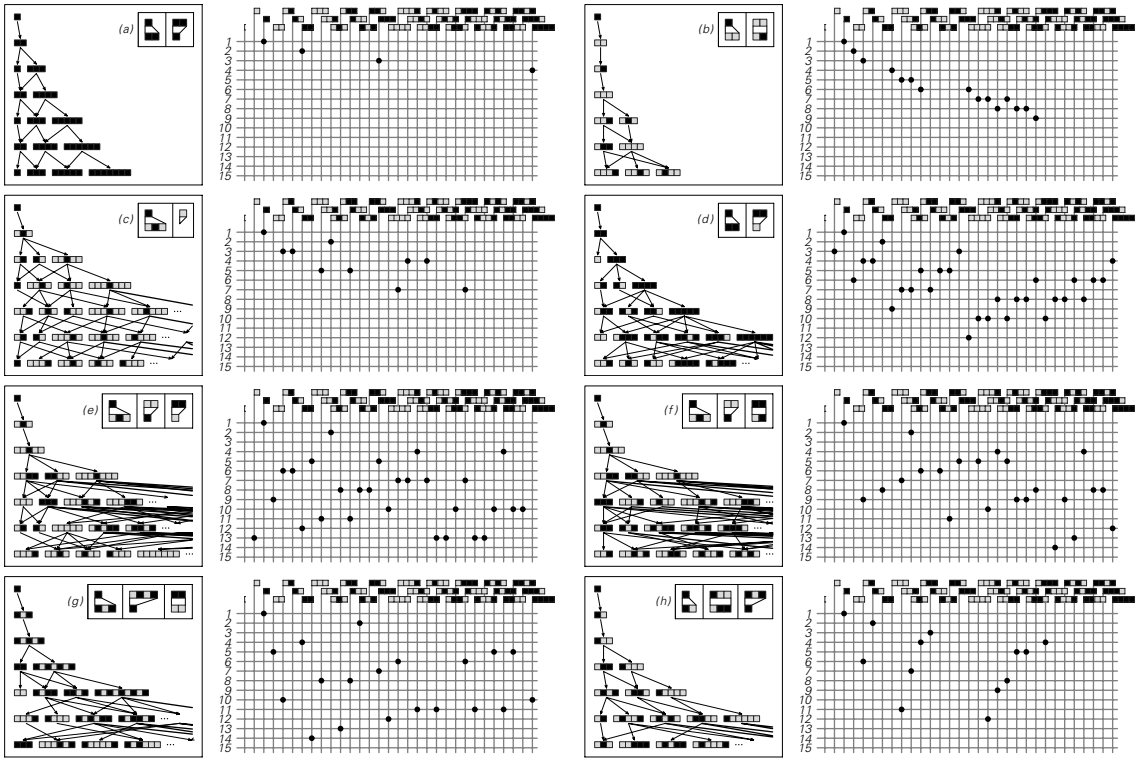
As I said above, it is at some level not surprising that questions will be considered interesting in a particular field only if the methods of that field can say something useful about them. But this I believe is ultimately why there have historically been so few signs of undecidability or unprovability in mathematics. For any kinds of questions in which such phenomena appear are usually not amenable to standard methods of mathematics based on proof, and as a result such questions have inevitably been viewed as being outside what should be considered interesting for mathematics.

So how then can one set up a reasonable idealization for mathematics as it is actually practiced? The first step—much as I discussed earlier in this section—is to think not so much about systems that might be described by mathematics as about the internal processes associated with proof that go on inside mathematics.

A proof must ultimately be based on an axiom system, and one might have imagined that over the course of time mathematics would have sampled a wide range of possible axiom systems. But in fact in its historical development mathematics has normally stuck to only rather few such systems—each one corresponding essentially to some identifiable field of mathematics, and most given on pages 773 and 774.

So what then happens if one looks at all possible simple axiom systems—much as we looked, say, at all possible simple cellular automata earlier in this book? To what extent does what one sees capture the features of mathematics? With axiom systems idealized as multiway systems the pictures on the next page show some results.

In some cases the total number of theorems that can ever be proved is limited. But often the number of theorems increases rapidly with the length of proof—and in most cases an infinite number of theorems can eventually be proved. And given experience with mathematics an obvious question to ask in such cases is to what extent the system is consistent, or complete, or both.



Plots showing which possible strings get generated in the first 15 steps of evolution in various multiway systems. Each string that is generated can be thought of as a theorem derived from the set of axioms represented by the rules of the multiway system. A dot shows at which step a given string first appears—and indicates the shortest proof of the theorem that string represents. In most cases, many strings are never produced—so that there are many possible statements that simply do not follow from the axioms given. Thus for example in first case shown only strings containing nothing but black elements are ever produced.

But to formulate such a question in a meaningful way one needs a notion of negation. In general, negation is just some operation that takes a string and yields another, giving back the original if it is applied a second time. Earlier in this section we discussed cases in which negation simply reverses the color of each element in a string. And as a generalization of this one can consider cases in which negation can be any operation that preserves lengths of strings.

And in this case it turns out that the criterion for whether a system is complete and consistent is simply that exactly half the

possible strings of a given length are eventually generated if one starts from the string representing "true".

For if more than half the strings are generated, then somewhere both a string and its negation would have to appear, implying that the system must be inconsistent. And similarly, if less than half the strings are generated, there must be some string for which neither that string nor its negation ever appear, implying that the system is incomplete.

The pictures on the next page show the fractions of strings of given lengths that are generated on successive steps in various multiway systems. In general one might have to wait an arbitrarily large number of steps to find out whether a given string will ever be generated. But in practice after just a few steps one already seems to get a reasonable indication of the overall fraction of strings that will ever be generated.

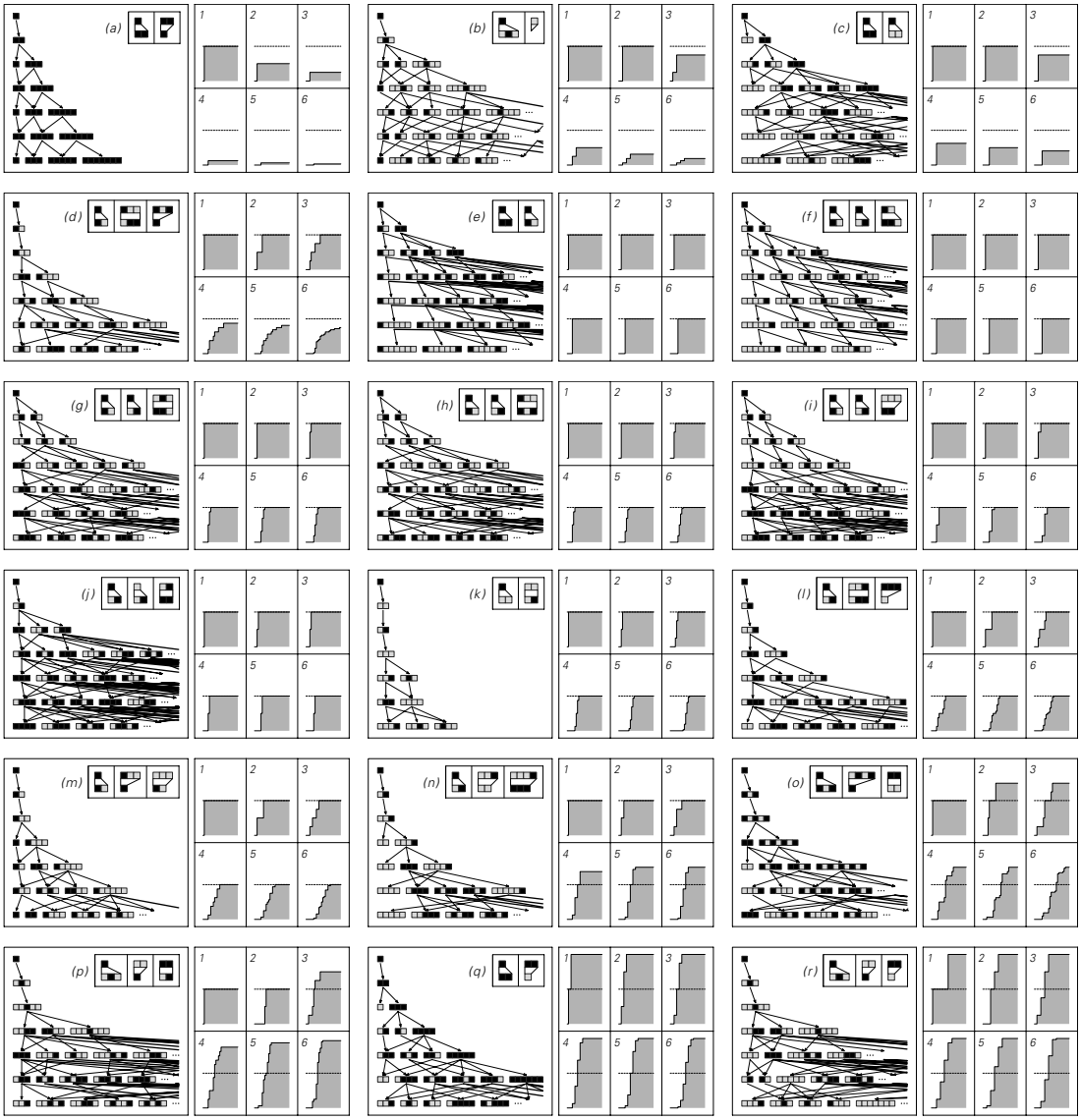
And what one sees is that there is a broad distribution: from cases in which very few strings can be generated—corresponding to a very incomplete axiom system—to cases in which all or almost all strings can be generated—corresponding to a very inconsistent axiom system.

So where in this distribution do the typical axiom systems of ordinary mathematics lie? Presumably none are inconsistent. And a few—like basic logic and real algebra—are both complete and consistent, so that in effect they lie right in the middle of the distribution. But most are known to be incomplete. And as we discussed above, this is inevitable as soon as universality is present.

But just how incomplete are they? The answer, it seems, is typically not very. For if one looks at axiom systems that are widely used in mathematics they almost all tend to be complete enough to prove at least a fair fraction of statements either true or false.

So why should this be? I suspect that it has to do with the fact that in mathematics one usually wants axiom systems that one can think of as somehow describing definite kinds of objects—about which one then expects to be able to establish all sorts of definite statements.

And certainly if one looks at the history of mathematics most basic axiom systems have been arrived at by starting with objects—such as finite integers or finite sets—then trying to find collections of axioms that somehow capture the relevant properties of these objects.



Examples of multiway systems that generate different fractions of possible strings, and in effect range from being highly incomplete to highly inconsistent. The plots show what fraction of strings of a given length have been produced by each of the first 25 steps in the evolution of each multiway system. If less than half the strings of a given length are ever produced, this means that there must be some strings where neither the string nor its negation can be proved, indicating incompleteness. But if more than half the strings are produced, there must be cases where both a string and its negation can be proved, indicating inconsistency. Rules (f) through (i), however, produce exactly half the strings of any given length, and can be considered complete and consistent.

But one feature is that normally the resulting axiom system is in a sense more general than the objects one started from. And this is why for example one can often use the axiom system to extrapolate to infinite situations. But it also means that it is not clear whether the axiom system actually describes only the objects one wants—or whether for example it also describes all sorts of other quite different objects.

One can think of an axiom system—say one of those listed on pages 773 and 774—as giving a set of constraints that any object it describes must satisfy. But as we saw in Chapter 5, it is often possible to satisfy a single set of constraints in several quite different ways.

And when this happens in an axiom system it typically indicates incompleteness. For as soon as there are just two objects that both satisfy the constraints but for which there is some statement that is true about one but false about the other it immediately follows that at least this statement cannot consistently be proved true or false, and that therefore the axiom system must be incomplete.

One might imagine that if one were to add more axioms to an axiom system one could always in the end force there to be only one kind of object that would satisfy the constraints of the system. But as we saw earlier, as soon as there is universality it is normally impossible to avoid incompleteness. And if an axiom system is incomplete there must inevitably be different kinds of objects that satisfy its constraints. For given any statement that cannot be proved from the axioms there must be distinct objects for which it is true, and for which it is false.

If an axiom system is far from complete—so that a large fraction of statements cannot be proved true or false—then there will typically be many different kinds of objects that are easy to specify and all satisfy the constraints of the system but for which there are fairly obvious properties that differ. But if an axiom system is close to complete—so that the vast majority of statements can be proved true or false—then it is almost inevitable that the different kinds of objects that satisfy its constraints must differ only in obscure ways.

And this is presumably the case in the standard axiom system for arithmetic from page 773. Originally this axiom system was intended to describe just ordinary integers. But Gödel's Theorem showed that it is

incomplete, so that there must be more than one kind of object that can satisfy its constraints. Yet it is rather close to being complete—since as we saw earlier one has to go through at least millions of statements before finding ones that it cannot prove true or false.

And this means that even though there are objects other than the ordinary integers that satisfy the standard axioms of arithmetic, they are quite obscure—in fact, so much so that none have ever yet actually been constructed with any real degree of explicitness. And this is why it has been reasonable to think of the standard axiom system of arithmetic as being basically just about ordinary integers.

But if instead of this standard axiom system one uses the reduced axiom system from page 773—in which the usual axiom for induction has been weakened—then the story is quite different. There is again incompleteness, but now there is much more of it, for even statements as simple as $x + y = y + x$ and $x + 0 = x$ cannot be proved true or false from the axioms. And while ordinary integers still satisfy all the constraints, the system is sufficiently incomplete that all sorts of other objects with quite different properties also do. So this means that the system is in a sense no longer about any very definite kind of mathematical object—and presumably that is why it is not used in practice in mathematics.

At this juncture it should perhaps be mentioned that in their raw form quite a few well-known axiom systems from mathematics are actually also far from complete. An example of this is the axiom system for group theory given on page 773. But the point is that this axiom system represents in a sense just the beginning of group theory. For it yields only those theorems that hold abstractly for any group.

Yet in doing group theory in practice one normally adds axioms that in effect constrain one to be dealing say with a specific group rather than with all possible groups. And the result of this is that once again one typically has an axiom system that is at least close to complete.

In basic arithmetic and also usually in fields like group theory the underlying objects that one imagines describing can at some level be manipulated—and understood—in fairly concrete ways. But in a field like set theory this is less true. Yet even in this case an attempt has

historically been made to get an axiom system that somehow describes definite kinds of objects. But now the main way this has been done is by progressively adding axioms so as to get closer to having a system that is complete—with only a rather vague notion of just what underlying objects one is really expecting to describe.

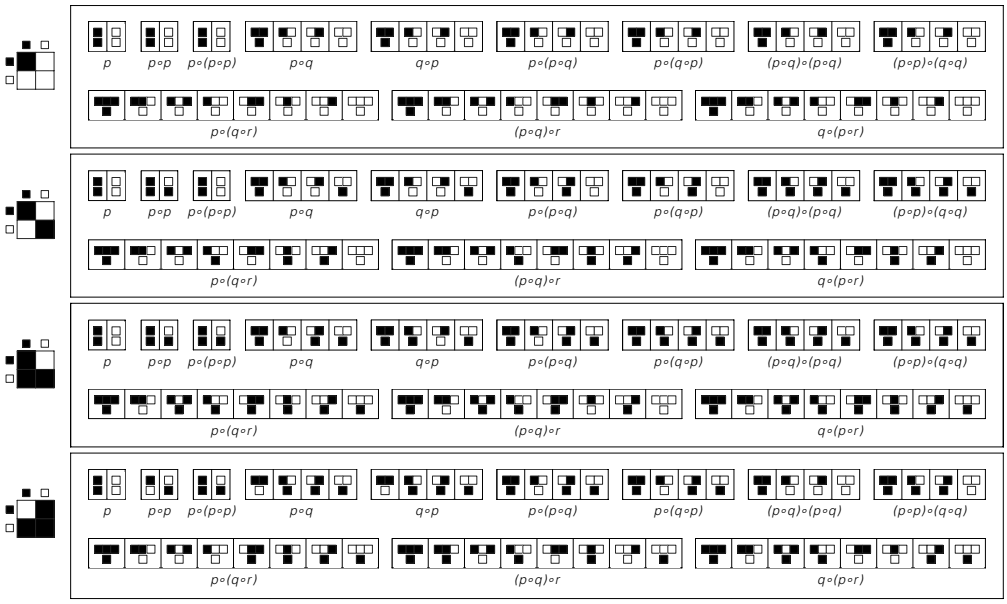
In studying basic processes of proof multiway systems seem to do well as minimal idealizations. But if one wants to study axiom systems that potentially describe definite objects it seems to be somewhat more convenient to use what I call operator systems. And indeed the version of logic used on page 775—as well as many of the axiom systems on pages 773 and 774—are already set up essentially as operator systems.

The basic idea of an operator system is to work with expressions such as $(p \circ q) \circ ((q \circ r) \circ p)$ built up using some operator $\circ$, and then to consider for example what equivalences may exist between such expressions. If one has an operator whose values are given by some finite table then it is always straightforward to determine whether expressions are equivalent. For all one need do, as in the pictures at the top of the next page, is to evaluate the expressions for all possible values of each variable, and then to see whether the patterns of results one gets are the same.

And in this way one can readily tell, for example, that the first operator shown is idempotent, so that $p \circ p = p$, while both the first two operators are associative, so that $(p \circ q) \circ r = p \circ (q \circ r)$, and all but the third operator are commutative, so that $p \circ q = q \circ p$. And in principle one can use this method to establish any equivalence that exists between any expressions with an operator of any specific form.

But the crucial idea that underlies the traditional approach to mathematical proof is that one should also be able to deduce such results just by manipulating expressions in purely symbolic form, using the rules of an axiom system, without ever having to do anything like filling in explicit values of variables.

And one advantage of this approach is that at least in principle it allows one to handle operators—like those found in many areas of mathematics—that are not based on finite tables. But even for operators given by finite tables it is often difficult to find axiom systems that can successfully reproduce all the results for a particular operator.

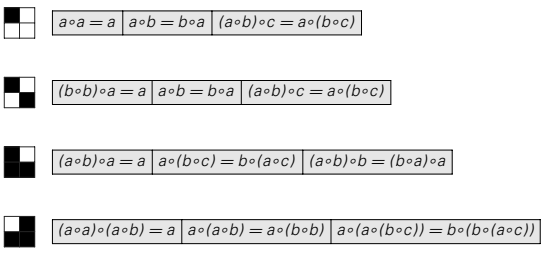


Values of expressions obtained by using operators of various forms. For each expression the sequence of values for every possible combination of values of variables is shown. Two expressions are equivalent when this sequence of values is the same. With black and white interpreted as TRUE and FALSE, the forms of operators shown here correspond respectively to AND, EQUAL, IMPLIES and NAND. (The first argument to each operator is shown on the left; the second on top.) The arrays of values generated can be thought of as being like truth tables.

With the way I have set things up, any axiom system is itself just a collection of equivalence results. So the question is then which equivalence results need to be included in the axiom system in order that all other equivalence results can be deduced just from these.

In general this can be undecidable—for there is no limit on how long even a single proof might need to be. But in some cases it turns out to be possible to establish that a particular set of axioms can successfully generate all equivalence results for a given operator—and indeed the picture at the top of the facing page shows examples of this for each of the four operators in the picture above.

So if two expressions are equivalent then by applying the rules of the appropriate axiom system it must be possible to get from one to the other—and in fact the picture on page 775 shows an example of how



Axiom systems that can be used to derive all the equivalences between expressions that involve operators with the forms shown. Each axiom can be applied in either direction—as in the picture on page 775, with each variable standing for any expression, as in a *Mathematica* pattern. The operators shown are AND, EQUAL, IMPLIES and NAND. They yield respectively junctional, equivalential, implicational and full propositional or sentential calculus (ordinary logic).

this can be done for the fourth axiom system above. But if one removes just a single axiom from any of the axiom systems above then it turns out that they no longer work, and for example they cannot establish the equivalence result stated by whichever axiom one has removed.

In general one can think of axioms for an operator system as giving constraints on the form of the operator. And if one is going to reproduce all the equivalences that hold for a particular form then these constraints must in effect be such as to force that form to occur.

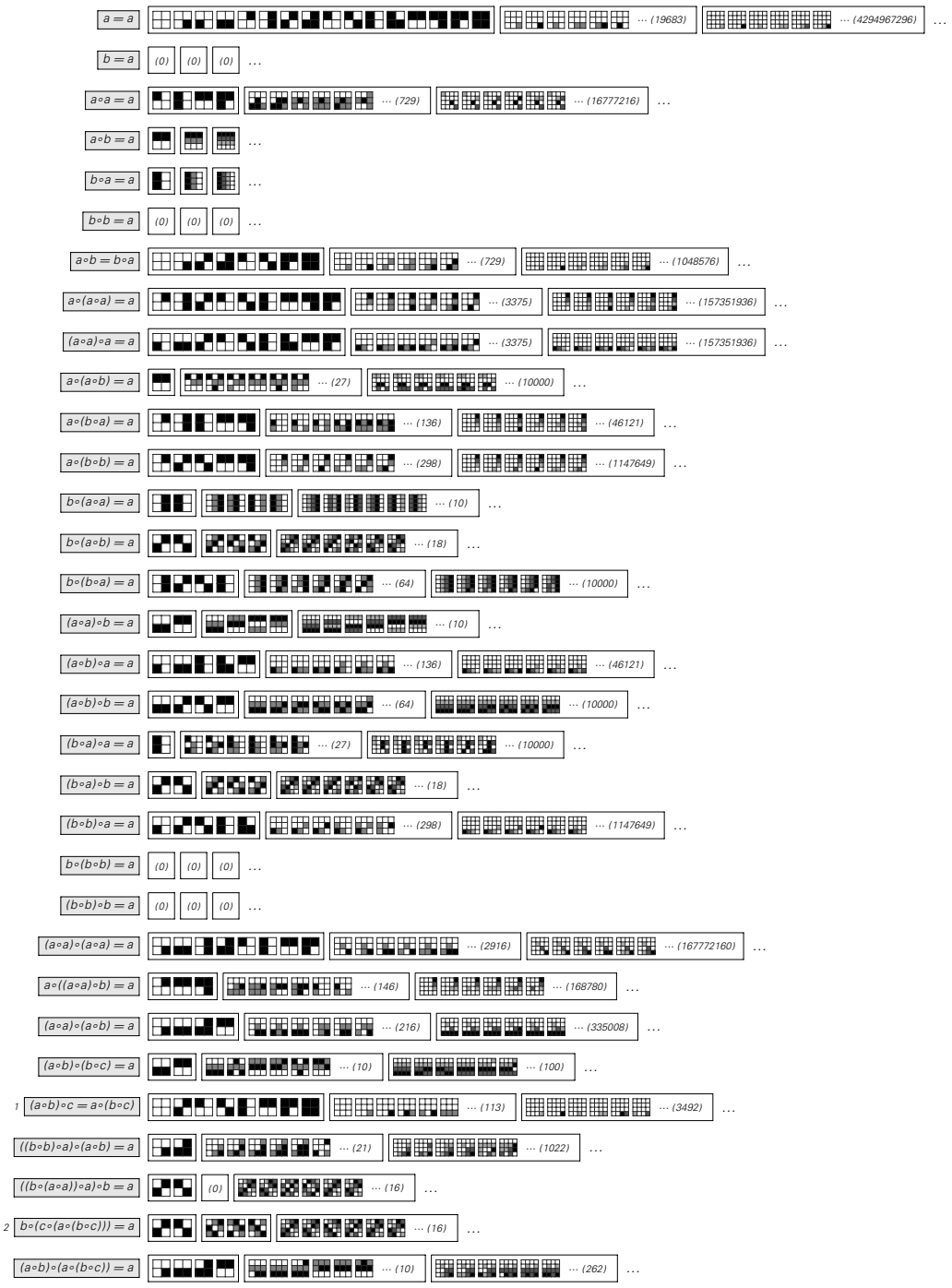
So what happens in general for arbitrary axiom systems? Do they typically force the operator to have a particular form, or not?

The pictures on the next two pages show which forms of operators are allowed by various different axiom systems. The successive blocks of results in each case give the forms allowed with progressively more possible values for each variable.

Indicated by stars near the bottom of the picture are the four axiom systems from the top of this page. And for each of these only a limited number of forms are allowed—all of which ultimately turn out to be equivalent to just the single forms shown on the facing page.

But what about other axiom systems? Every axiom system must allow an operator of at least some form. But what the pictures on the next two pages show is that the vast majority of axiom systems actually allow operators with all sorts of different forms.

And what this means is that these axiom systems are in a sense not really about operators of any particular form. And so in effect they are also far from complete—for they can prove only equivalence results that hold for every single one of the various operators they allow.


a = a
… (19683)
… (4294967296)
b = a
(0)
(0)
(0)
a∘a = a
… (729)
… (16777216)
a∘b = a
b∘a = a
b∘b = a
a∘b = b∘a
… (729)
… (1048576)
a∘(a∘a) = a
… (3375)
… (157351936)
(a∘a)∘a = a
… (3375)
… (157351936)
a∘(a∘b) = a
… (27)
… (10000)
a∘(b∘a) = a
… (136)
… (46121)
a∘(b∘b) = a
… (298)
… (1147649)
b∘(a∘a) = a
… (10)
b∘(a∘b) = a
… (18)
b∘(b∘a) = a
… (64)
… (10000)
(a∘a)∘b = a
… (10)
(a∘b)∘a = a
… (136)
… (46121)
(a∘b)∘b = a
… (64)
… (10000)
(b∘a)∘a = a
… (27)
… (10000)
(b∘a)∘b = a
… (18)
(b∘b)∘a = a
… (298)
… (1147649)
b∘(b∘b) = a
(0)
(0)
(0)
(b∘b)∘b = a
(0)
(0)
(0)
(a∘a)∘(a∘a) = a
… (2916)
… (167772160)
a∘((a∘a)∘b) = a
… (146)
… (168780)
(a∘a)∘(a∘b) = a
… (216)
… (335008)
(a∘b)∘(b∘c) = a
… (10)
… (100)
1
(a∘b)∘c = a∘(b∘c)
… (113)
… (3492)
((b∘b)∘a)∘(a∘b) = a
… (21)
… (1022)
((b∘(a∘a))∘a)∘b = a
(0)
… (16)
2
b∘(c∘(a∘(b∘c))) = a
… (16)
(a∘b)∘(a∘(b∘c)) = a
… (10)
… (262)

Forms of a binary operator satisfying the constraints of a series of different axiom systems. The successive blocks of results in each case show forms of the operator allowed with 2, 3 and 4 possible elements. Note that with 3 and 4 elements, only forms inequivalent under interchange of element labels are shown. Representations of notable systems in mathematics are: (1) semigroup theory, (2) commutative group theory, (3) basic logic, (4) commutative semigroup theory, (5) squag theory, (6) group theory, (7) junctional calculus, (8) equivalential calculus and (9) implicational calculus. In each case the operator forms shown correspond to possible semigroups, commutative groups, systems of logic (Boolean algebras), etc. with 2, 3 and 4 possible elements. The operator forms shown can be thought of as giving multiplication tables. In model theory, these forms are usually called the models of an axiom system.

So if one makes a list of all possible axiom systems—say starting with the simplest—where in such a list should one expect to see axiom systems that correspond to traditional areas of mathematics?

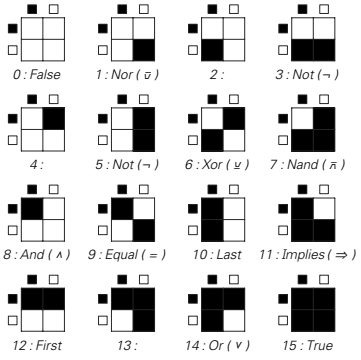
Most axiom systems as they are given in typical textbooks are sufficiently complicated that they will not show up at all early. And in fact the only immediate exception is the axiom system $\{(a\circ b)\circ c = a\circ(b\circ c)\}$ for what are known as semigroups—which ironically are usually viewed as rather advanced mathematical objects.

But just how complicated do the axiom systems for traditional areas of mathematics really need to be? Often it seems that they can be vastly simpler than their textbook forms. And so, for example, as page 773 indicates, interpreting the $\circ$ operator as division, $\{a\circ(b\circ(c\circ(a\circ b))) = c\}$ is known to be an axiom system for commutative group theory, and $\{a\circ((((a\circ a)\circ b)\circ c)\circ(((a\circ a)\circ a)\circ c)) = b\}$ for general group theory.

So what about basic logic? How complicated an axiom system does one need for this? Textbook discussions of logic mostly use axiom systems at least as complicated as the first one on page 773. And such axiom systems not only involve several axioms—they also normally involve three separate operators: AND ($\wedge$), OR ($\vee$) and NOT ($\neg$).

But is this in fact the only way to formulate logic?

As the picture below shows, there are 16 different possible operators that take two arguments and allow two values, say true and false. And of these AND, OR and NOT are certainly the most commonly used in both everyday language and most of mathematics.

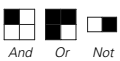


Logical functions of two arguments and their common names. Black stands for TRUE; white for FALSE. AND, OR, NOT, and IMPLIES are widely used in traditional logic. EQUAL (if and only if) is common in more mathematical settings, while XOR is widespread in discrete mathematics. NAND and NOR are mostly used only in circuit design and in a few foundational studies of logic. The first argument for each function appears on the left in the picture; the second argument on top. The functions are numbered like 2-neighbor analogs of the cellular automaton rules of page 53.

But at least at a formal level, logic can be viewed simply as a theory of functions that take on two possible values given variables with two possible values. And as we discussed on page 616, any such function can be represented as a combination of AND, OR and NOT.

But the table below demonstrates that as soon as one goes beyond the familiar traditions of language and mathematics there are other operators that can also just as well be used as primitives. And indeed it has been known since before 1900 that both NAND and NOR on their own work—a fact I already used on pages 617 and 775.

0	$\neg a \wedge a$	1	$\neg (a \vee b)$	2	$\neg a \wedge b$	3	$\neg a$
4	$\neg b \wedge a$	5	$\neg b$	6	$\neg (a \wedge b) \wedge (a \vee b)$	7	$\neg (a \wedge b)$
8	$a \wedge b$	9	$a \wedge b \vee \neg (a \vee b)$	10	b	11	$\neg a \vee b$
12	a	13	$\neg b \vee a$	14	$a \vee b$	15	$\neg a \vee a$



0	$\neg a \wedge a$	1	$\neg a \wedge \neg b$	2	$\neg a \wedge b$	3	$\neg a$
4	$\neg b \wedge a$	5	$\neg b$	6	$\neg (\neg a \wedge \neg b) \wedge \neg (a \wedge b)$	7	$\neg (a \wedge b)$
8	$a \wedge b$	9	$\neg (\neg a \wedge b) \wedge \neg (\neg b \wedge a)$	10	b	11	$\neg (\neg b \wedge a)$
12	a	13	$\neg (\neg a \wedge b)$	14	$\neg (\neg a \wedge \neg b)$	15	$\neg (\neg a \wedge a)$

0	$\neg (\neg a \vee a)$	1	$\neg (a \vee b)$	2	$\neg (\neg b \vee a)$	3	$\neg a$
4	$\neg (\neg a \vee b)$	5	$\neg b$	6	$\neg (\neg a \vee b) \vee \neg (\neg b \vee a)$	7	$\neg a \vee \neg b$
8	$\neg (\neg a \vee \neg b)$	9	$\neg (\neg a \vee \neg b) \vee \neg (a \vee b)$	10	b	11	$\neg a \vee b$
12	a	13	$\neg b \vee a$	14	$a \vee b$	15	$\neg a \vee a$

0	$\neg (a \Rightarrow a)$	1	$\neg (\neg a \Rightarrow b)$	2	$\neg (b \Rightarrow a)$	3	$\neg a$
4	$\neg (a \Rightarrow b)$	5	$\neg b$	6	$(a \Rightarrow b) \Rightarrow \neg (b \Rightarrow a)$	7	$a \Rightarrow \neg b$
8	$\neg (a \Rightarrow \neg b)$	9	$\neg ((a \Rightarrow b) \Rightarrow \neg (b \Rightarrow a))$	10	b	11	$a \Rightarrow b$
12	a	13	$b \Rightarrow a$	14	$\neg a \Rightarrow b$	15	$a \Rightarrow a$

0	$a \veebar a$	1	$(a \Rightarrow b) \veebar b$	2	$((a \Rightarrow b) \Rightarrow b) \veebar a$	3	$(a \Rightarrow a) \veebar a$
4	$((a \Rightarrow b) \Rightarrow b) \veebar b$	5	$(a \Rightarrow a) \veebar b$	6	$a \veebar b$	7	$(a \Rightarrow b) \veebar a$
8	$(((a \Rightarrow b) \Rightarrow b) \veebar a) \veebar b$	9	$((a \Rightarrow a) \veebar a) \veebar b$	10	b	11	$a \Rightarrow b$
12	a	13	$b \Rightarrow a$	14	$(a \Rightarrow b) \Rightarrow b$	15	$a \Rightarrow a$

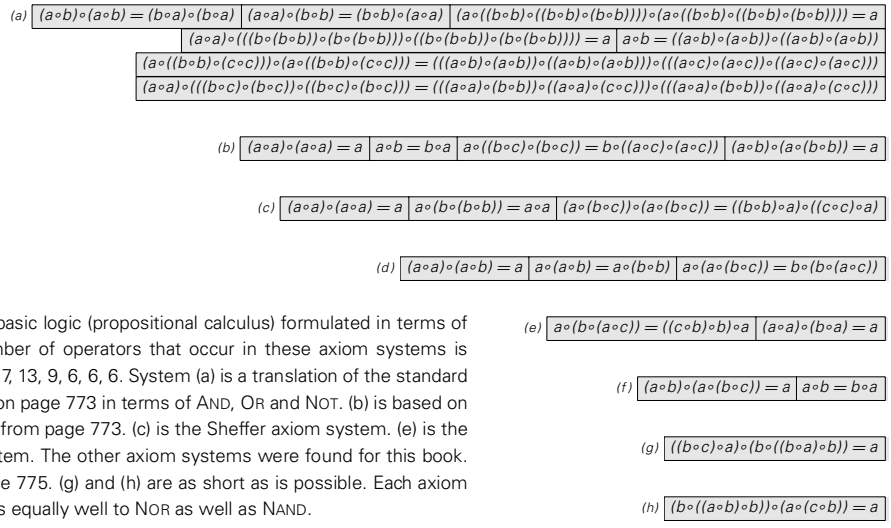
0	$a \circ a$	1	$a \circ (a \barwedge b)$	2	$a \circ b$	3	$a \circ (a \barwedge a)$
4	$b \circ a$	5	$b \circ (a \barwedge a)$	6	$a \circ b \barwedge (b \barwedge a)$	7	$a \circ b \barwedge b$
8	$(a \circ b) \circ b$	9	$(a \circ b) \circ (b \barwedge a)$	10	b	11	$b \barwedge a$
12	a	13	$a \barwedge b$	14	$a \barwedge (a \barwedge b)$	15	$a \barwedge a$

0	$((a \barwedge a) \barwedge a) \barwedge ((a \barwedge a) \barwedge a)$	1	$((a \barwedge a) \barwedge (b \barwedge b)) \barwedge ((a \barwedge a) \barwedge a)$	2	$((a \barwedge a) \barwedge a) \barwedge ((a \barwedge a) \barwedge b)$	3	$a \barwedge a$
4	$((a \barwedge a) \barwedge a) \barwedge ((a \barwedge b) \barwedge a)$	5	$b \barwedge b$	6	$((a \barwedge a) \barwedge b) \barwedge ((a \barwedge b) \barwedge a)$	7	$a \barwedge b$
8	$(a \barwedge b) \barwedge (a \barwedge b)$	9	$((a \barwedge a) \barwedge (b \barwedge b)) \barwedge (a \barwedge b)$	10	b	11	$(a \barwedge b) \barwedge a$
12	a	13	$(a \barwedge a) \barwedge b$	14	$(a \barwedge a) \barwedge (b \barwedge b)$	15	$(a \barwedge a) \barwedge a$

0	$(a \barvee a) \barvee a$	1	$a \barvee b$	2	$(a \barvee b) \barvee a$	3	$a \barvee a$
4	$(a \barvee a) \barvee b$	5	$b \barvee b$	6	$((a \barvee a) \barvee (b \barvee b)) \barvee (a \barvee b)$	7	$((a \barvee a) \barvee (b \barvee b)) \barvee ((a \barvee a) \barvee a)$
8	$(a \barvee a) \barvee (b \barvee b)$	9	$((a \barvee a) \barvee b) \barvee ((a \barvee b) \barvee a)$	10	b	11	$((a \barvee a) \barvee a) \barvee ((a \barvee a) \barvee b)$
12	a	13	$((a \barvee a) \barvee a) \barvee ((a \barvee b) \barvee a)$	14	$(a \barvee b) \barvee (a \barvee b)$	15	$((a \barvee a) \barvee a) \barvee ((a \barvee a) \barvee a)$

Functions that can be used to formulate logic. In each case the minimal combinations of primitive functions necessary to reproduce each of the 16 logical functions of two arguments is given. From these any possible logical function with any number of arguments can be obtained. Most textbook treatments of logic use AND, OR, and NOT as primitive functions. NAND and NOR are the only primitive functions that work on their own.

So this means that logic can be set up using just a single operator. But how complicated an axiom system does it then need? The first box in the picture below shows that the direct translation of the standard textbook AND, OR, NOT axiom system from page 773 is very complicated.



Axiom systems for basic logic (propositional calculus) formulated in terms of NAND ($\bar{\wedge}$). The number of operators that occur in these axiom systems is respectively 94, 17, 17, 13, 9, 6, 6, 6. System (a) is a translation of the standard textbook one given on page 773 in terms of AND, OR and NOT. (b) is based on the Robbins axioms from page 773. (c) is the Sheffer axiom system. (e) is the Meredith axiom system. The other axiom systems were found for this book. (d) was used on page 775. (g) and (h) are as short as is possible. Each axiom system given applies equally well to NOR as well as NAND.

But boxes (b) and (c) show that known alternative axiom systems for logic reduce the size of the axiom system by about a factor of ten. And some further reduction is achieved by manipulating the resulting axioms—leading to the axiom system used above and given in box (d).

But can one go still further? And what happens for example if one just tries to search simple axiom systems for ones that work?

One can potentially test axiom systems by seeing what operators satisfy their constraints, as on page 805. The first non-trivial axiom system that even allows the NAND operator is $\{(a\circ a)\circ(a\circ a) = a\}$. And the first axiom system for which NAND and NOR are the only operators allowed that involve 2 possible values is $\{((b\circ b)\circ a)\circ(a\circ b) = a\}$.

But if one now looks at operators involving 3 possible values then it turns out that this axiom system allows ones not equivalent to NAND

and NOR. And this means that it cannot successfully reproduce all the results of logic. Yet if any axiom system with just a single axiom is going to be able to do this, the axiom must be of the form $\{\ldots = a\}$.

With up to 6 NANDs and 2 variables none of the 16,896 possible axiom systems of this kind work even up to 3-value operators. But with 6 NANDs and 3 variables, 296 of the 288,684 possible axiom systems work up to 3-value operators, and 100 work up to 4-value operators.

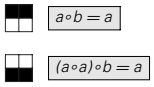
And of the 25 of these that are not trivially equivalent, it then turns out that the two given as (g) and (h) on the facing page can actually be proved as on the next two pages to be axiom systems for logic—thus showing that in the end quite remarkable simplification can be achieved relative to ordinary textbook axiom systems.

If one looks at axiom systems of the form $\{\ldots = a,\ a\circ b = b\circ a\}$ the first one that one finds that allows only NAND and NOR with 2-value operators is $\{(a\circ a)\circ(a\circ a) = a,\ a\circ b = b\circ a\}$. But as soon as one uses a total of just 6 NANDs, one suddenly finds that out of the 3402 possibilities with 3 variables 32 axiom systems equivalent to case (f) above all end up working all the way up to at least 4-value operators. And in fact it then turns out that (f) indeed works as an axiom system for logic.

So what this means is that if one were just to go through a list of the simplest few thousand axiom systems one would already be quite likely to find one that represents logic.

In human intellectual history logic has had great significance. But if one looks just at axiom systems is there anything obviously special about the ones for logic? My guess is that unless one asks about very specific details there is really not—and that standard logic is in a sense distinguished in the end only by its historical context.

One feature of logic is that its axioms effectively describe a single specific operator. But it turns out that there are all sorts of other axioms that also do this. I gave three examples on page 803, and in the picture on the right I give two more very simple examples. Indeed, given many forms of operator there are always axiom systems that can be found to describe it.



Axiom systems that reproduce equivalence results for the forms of operators shown.

L1 (a ((a a) a)) (a ((a a) a))
= [A] (((((a a) a) (a ((a a) a))) (a ((a a) a))) (((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a)))) (a ((a a) a))
= [A] (((((a a) a) (a ((a a) a))) (a ((a a) a))) (((a a) a) (a ((a a) a)))) (a ((a a) a))
= [A] (((((a a) a) (a ((a a) a))) (a ((a a) a))) a) (a ((a a) a))
= [A] ((a (a ((a a) a))) a) (a ((a a) a))
= [A] a

L2 (a a) ((a ((a a) a)) a)
= [A] (a a) ((a ((a a) a)) (((a ((a a) a)) a) (a ((a a) a))))
= [L1] (a a) ((a ((a a) a)) (((a ((a a) a)) ((a ((a a) a)) (a ((a a) a)))) (a ((a a) a))))
= [L1] (((a ((a a) a)) (a ((a a) a))) ((a ((a a) a)) (a ((a a) a)))) ((a ((a a) a)) (((a ((a a) a)) ((a ((a a) a)) (a ((a a) a)))) (a ((a a) a))))
= [A] (a ((a a) a)) (a ((a a) a))
= [L1] a

L3 (a b) ((a a) (((a a) b) (a a)))
= [L2] (((a a) ((a ((a a) a)) a)) b) ((a a) (((a a) b) (a a)))
= [A] b

L4 ((a ((a b) a)) d) (b ((b d) b))
= [A] ((a ((a b) a)) d) (b ((b d) (((a c) b) (a ((a b) a)))))
= [A] ((a ((a b) a)) d) (b ((((a c) b) (a ((a b) a))) d) (((a c) b) (a ((a b) a)))))
= [A] ((a ((a b) a)) d) ((((a c) b) (a ((a b) a))) (((((a c) b) (a ((a b) a))) d) (((a c) b) (a ((a b) a)))))
= [L3] (((((a c) b) (a ((a b) a))) ((((a c) b) ((a c) b)) (((((a c) b) ((a c) b)) (a ((a b) a))) (((a c) b) ((a c) b))))) d) ((((a c) b) (a ((a b) a))) (((((a c) b) (a ((a b) a))) d) (((a c) b) (a ((a b) a)))))
= [A] d

L5 (a ((a a) a)) ((a ((a a) a)) a)
= [A] (a ((a a) a)) ((a ((a a) a)) (((a ((a a) a)) a) (a ((a a) a))))
= [A] (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) (((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a)))) ((a ((a a) a)) (((a ((a a) a)) a) (a ((a a) a))))
= [A] (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) (((a a) a) (a ((a a) a)))) ((a ((a a) a)) (((a ((a a) a)) a) (a ((a a) a))))
= [A] (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) a) ((a ((a a) a)) (((a ((a a) a)) a) (a ((a a) a))))
= [A] (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) (((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))))) a) ((a ((a a) a)) (((a ((a a) a)) a) (a ((a a) a))))
= [L4] a

L6 (a ((a a) a)) (a (a ((a a) a)))
= [L1] (a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a)))
= [A] ((((a ((a a) a)) ((a ((a a) a)) a)) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) (((a ((a a) a)) ((a ((a a) a)) a)) ((((a ((a a) a)) ((a ((a a) a)) a)) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) ((a ((a a) a)) a))))
= [A] (a ((a a) a)) (((a ((a a) a)) ((a ((a a) a)) a)) ((((a ((a a) a)) ((a ((a a) a)) a)) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) ((a ((a a) a)) a))))
= [L5] (a ((a a) a)) (a ((((a ((a a) a)) ((a ((a a) a)) a)) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) ((a ((a a) a)) a))))
= [L5] (a ((a a) a)) (a ((a ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) ((a ((a a) a)) a))))
= [L1] (a ((a a) a)) (a ((a ((a ((a a) a)) (a (a ((a a) a))))) ((a ((a a) a)) ((a ((a a) a)) a))))
= [L1] (a ((a a) a)) (a ((a ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) ((a ((a a) a)) a))))
= [L4] (a ((a a) a)) (a (((((a ((a a) a)) a) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) ((a ((a a) a)) a))))
= [A] (a ((a a) a)) (a ((a ((a a) a)) ((a ((a a) a)) ((a ((a a) a)) a))))

L7 (a ((a a) a)) (a (a ((a a) a)))
= [L6] (a ((a a) a)) (a ((a ((a a) a)) ((a ((a a) a)) ((a ((a a) a)) a))))
= [L5] (a ((a a) a)) (a ((a ((a a) a)) a))
= [A] (a ((a a) a)) (a (((((a a) a) (a ((a a) a))) ((a a) (((a a) (a ((a a) a))) (a a)))) a))
= [A] (a ((a a) a)) (a ((a ((a a) (((a a) (a ((a a) a))) (a a)))) a))
= [L3] ((a (a ((a a) a))) ((a a) (((a a) (a ((a a) a))) (a a)))) (a ((a ((a a) (((a a) (a ((a a) a))) (a a)))) a))
= [A] (a a) (((a a) (a ((a a) a))) (a a))

L8 ((a a) (((a a) (a ((a a) a))) (a a))) ((a a) (((a a) (a ((a a) a))) (a a)))
= [L7] ((a ((a a) a)) (a (a ((a a) a)))) ((a ((a a) a)) (a (a ((a a) a))))
= [L1] ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a)))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))
= [L1] a ((a a) a)

L9 (a ((a a) a)) (a (a ((a a) a)))
= [L1] (a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a)))
= [L8] (((a ((a a) a)) (a ((a a) a))) ((((a ((a a) a)) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a))))) (((a ((a a) a)) (a ((a a) a))) ((((a ((a a) a)) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a)))))
= [L1] (a ((((a ((a a) a)) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a))))) (a ((((a ((a a) a)) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a)))))

L10 (a a) (((a a) (a ((a a) a))) (a a))
= [L7] (a ((a a) a)) (a (a ((a a) a)))
= [L9] (a ((((a ((a a) a)) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a))))) (a ((((a ((a a) a)) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a)))))
= [L1] (a ((a ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a))))) (a ((a ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a)))))

L11 (a a) (((a a) (a ((a a) a))) (a a))
= [L10] (a ((a ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a))))) (a ((a ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a)))))
= [L1] (a ((a ((a ((a a) a)) (a (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a))))) (a ((a ((a ((a a) a)) (a (a ((a a) a))))) ((a ((a a) a)) (a ((a a) a)))))
= [L7] (a ((a ((a a) (((a a) (a ((a a) a))) (a a)))) ((a ((a a) a)) (a ((a a) a))))) (a ((a ((a a) (((a a) (a ((a a) a))) (a a)))) ((a ((a a) a)) (a ((a a) a)))))

L12 (a a) (((a a) (a ((a a) a))) (a a))
= [L11] (a ((a ((a a) (((a a) (a ((a a) a))) (a a)))) ((a ((a a) a)) (a ((a a) a))))) (a ((a ((a a) (((a a) (a ((a a) a))) (a a)))) ((a ((a a) a)) (a ((a a) a)))))
= [A] (a (((((a a) a) (a ((a a) a))) ((a a) (((a a) (a ((a a) a))) (a a)))) ((a ((a a) a)) (a ((a a) a))))) (a (((((a a) a) (a ((a a) a))) ((a a) (((a a) (a ((a a) a))) (a a)))) ((a ((a a) a)) (a ((a a) a)))))
= [A] (a ((a ((a a) a)) ((a ((a a) a)) (a ((a a) a))))) (a ((a ((a a) a)) ((a ((a a) a)) (a ((a a) a)))))
= [L1] (a ((a ((a a) a)) a)) (a ((a ((a a) a)) a))

L13 (a ((a a) a)) (a (a ((a a) a)))
= [L7] (a a) (((a a) (a ((a a) a))) (a a))
= [L12] (a ((a ((a a) a)) a)) (a ((a ((a a) a)) a))

L14 (a ((a ((a a) a)) a)) (a ((a ((a a) a)) a))
= [L13] (a ((a a) a)) (a (a ((a a) a)))
= [L1] (a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a)))
= [A] ((((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) ((((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a)))))
= [A] ((((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) (((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))))) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) ((((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a)))))
= [L4] (a ((a a) a)) (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) ((((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))) ((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a)))))
= [L4] (a ((a a) a)) (((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a))) ((a ((a a) a)) ((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a)))))
= [A] (a ((a a) a)) (((((a a) a) (a ((a a) a))) (a ((a a) a))) ((a ((a a) a)) ((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a)))))
= [A] (a ((a a) a)) ((a (a ((a a) a))) ((a ((a a) a)) ((((a a) a) ((((a a) a) (a ((a a) a))) ((a a) a))) (a ((a a) a)))))
= [A] (a ((a a) a)) ((a (a ((a a) a))) ((a ((a a) a)) ((((a a) a) (a ((a a) a))) (a ((a a) a)))))

L15 (a ((a ((a a) a)) a)) (a ((a ((a a) a)) a))
= [L14] (a ((a a) a)) ((a (a ((a a) a))) ((a ((a a) a)) ((((a a) a) (a ((a a) a))) (a ((a a) a)))))
= [A] (a ((a a) a)) ((a (a ((a a) a))) ((a ((a a) a)) (a (a ((a a) a)))))
= [L13] (a ((a a) a)) ((a (a ((a a) a))) ((a ((a ((a a) a)) a)) (a ((a ((a a) a)) a))))

L16 (a ((a ((a a) a)) a)) (a ((a ((a a) a)) a))
= [L15] (a ((a a) a)) ((a (a ((a a) a))) ((a ((a ((a a) a)) a)) (a ((a ((a a) a)) a))))
= [L12] (a ((a a) a)) ((a (a ((a a) a))) ((a a) (((a a) (a ((a a) a))) (a a))))
= [L3] (a ((a a) a)) (a ((a a) a))

L17 (a ((a a) a)) (a (a ((a a) a)))
= [L13] (a ((a ((a a) a)) a)) (a ((a ((a a) a)) a))
= [L16] (a ((a a) a)) (a ((a a) a))
= [L1] a

L18 a ((a a) a)
= [L1] ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a)))) ((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a))))
= [L16] ((a ((a a) a)) (((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a)))) (a ((a a) a)))) ((a ((a a) a)) (((a ((a a) a)) (((a ((a a) a)) (a ((a a) a))) (a ((a a) a)))) (a ((a a) a))))
= [L1] ((a ((a a) a)) (((a ((a a) a)) (a (a ((a a) a)))) (a ((a a) a)))) ((a ((a a) a)) (((a ((a a) a)) (a (a ((a a) a)))) (a ((a a) a))))
= [L17] ((a ((a a) a)) (a (a ((a a) a)))) ((a ((a a) a)) (a (a ((a a) a))))
= [L17] a a

L19 (a a) (a (a a))
= [L18] (a a) (a (a ((a a) a)))
= [L18] (a ((a a) a)) (a (a ((a a) a)))
= [L17] a

L20 a (a ((a ((a a) a)) a))
= [L18] a (a ((a ((a ((a a) a)) a)) a))
= [L2] ((a a) ((a ((a a) a)) a)) (a ((a ((a ((a a) a)) a)) a))
= [A] (a ((a a) a)) a

L21 a (a a)
= [L18] a (a ((a a) a))
= [L18] a (a ((a ((a a) a)) a))
= [L20] (a ((a a) a)) a

L22 (a a) (a a)
= [L18] (a ((a a) a)) (a ((a a) a))
= [L1] a

T1 (a a) (a a)
= [L22] a

L23 (a a) a
= [L18] (a ((a a) a)) a
= [L21] a (a a)

L24 ((a b) a) (a a)
= [L18] ((a b) a) (a ((a a) a))
= [A] a

L25 a ((a ((a b) (a b))) a)
= [A] (((a b) ((a b) (a b))) (a ((a ((a b) (a b))) a))) ((a b) (((a b) (a ((a ((a b) (a b))) a))) (a b)))
= [A] ((a b) (a b)) ((a b) (((a b) (a ((a ((a b) (a b))) a))) (a b)))
= [L19] ((a b) (a b)) ((a b) (((((a b) (a b)) ((a b) ((a b) (a b)))) (a ((a ((a b) (a b))) a))) (a b)))
= [A] ((a b) (a b)) ((a b) (((((a b) (a b)) ((((a b) (a b)) (a ((a (a b)) a))) ((a b) (a b)))) (a ((a ((a b) (a b))) a))) (a b)))
= [L24] ((a b) (a b)) ((a b) (((((((a b) (a b)) ((((a b) (a b)) (a ((a (a b)) a))) ((a b) (a b)))) (a ((a (a b)) a))) (((a b) (a b)) ((((a b) (a b)) (a ((a (a b)) a))) ((a b) (a b))))) ((((a b) (a b)) ((((a b) (a b)) (a ((a (a b)) a))) ((a b) (a b)))) (((a b) (a b)) ((((a b) (a b)) (a ((a (a b)) a))) ((a b) (a b)))))) (a ((a ((a b) (a b))) a))) (a b)))
= [A] ((a b) (a b)) ((a b) ((((a ((a (a b)) a)) ((((a b) (a b)) ((((a b) (a b)) (a ((a (a b)) a))) ((a b) (a b)))) (((a b) (a b)) ((((a b) (a b)) (a ((a (a b)) a))) ((a b) (a b)))))) (a ((a ((a b) (a b))) a))) (a b)))
= [A] ((a b) (a b)) ((a b) ((((a ((a (a b)) a)) ((((a b) (a b)) ((a b) ((a b) (a b)))) (((a b) (a b)) ((a b) ((a b) (a b)))))) (a ((a ((a b) (a b))) a))) (a b)))
= [L19] ((a b) (a b)) ((a b) ((((a ((a (a b)) a)) ((a b) (a b))) (a ((a ((a b) (a b))) a))) (a b)))
= [A] ((a b) (a b)) ((a b) (((a b) (a b)) (a b)))

L26 a ((a ((a b) (a b))) a)
= [L25] ((a b) (a b)) ((a b) (((a b) (a b)) (a b)))
= [A] ((a b) (a b)) (((((a b) (a b)) (a b)) ((a b) (((a b) (a b)) (a b)))) (((a b) (a b)) ((((a b) (a b)) ((a b) (((a b) (a b)) (a b)))) ((a b) (a b)))))
= [A] ((a b) (a b)) ((a b) (((a b) (a b)) ((((a b) (a b)) ((a b) (((a b) (a b)) (a b)))) ((a b) (a b)))))
= [L12] ((a b) (a b)) ((a b) (((a b) (((a b) (((a b) (a b)) (a b))) (a b))) ((a b) (((a b) (((a b) (a b)) (a b))) (a b)))))
= [L16] ((a b) (a b)) ((a b) (((a b) (((a b) (a b)) (a b))) ((a b) (((a b) (a b)) (a b)))))
= [L1] ((a b) (a b)) ((a b) (a b))

L27 a ((a ((a b) (a b))) a)
= [L26] ((a b) (a b)) ((a b) (a b))
= [L22] a b

L28 a
= [L24] ((a b) a) (a a)
= [L27] ((a b) a) ((((a b) a) ((((a b) a) (a a)) (((a b) a) (a a)))) ((a b) a))
= [L24] ((a b) a) ((((a b) a) (a a)) ((a b) a))

L29 ((a b) a) (a ((a b) a))
= [L24] ((a b) a) ((((a b) a) (a a)) ((a b) a))
= [L28] a

L30 (a b) (a a)
= [L27] (a ((a ((a b) (a b))) a)) (a a)
= [L29] (a ((a ((a b) (a b))) a)) ((((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))) (((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))))
= [L29] (a ((a ((a b) (a b))) a)) ((((a ((a b) (a b))) a) ((((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))) ((a ((a b) (a b))) a))) (((a ((a b) (a b))) a) ((((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))) ((a ((a b) (a b))) a))))
= [A] (((((a ((a b) (a b))) a) ((((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))) ((a ((a b) (a b))) a))) (a ((a ((a b) (a b))) a))) (((a ((a b) (a b))) a) ((((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))) ((a ((a b) (a b))) a)))) ((((a ((a b) (a b))) a) ((((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))) ((a ((a b) (a b))) a))) (((a ((a b) (a b))) a) ((((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))) ((a ((a b) (a b))) a))))
= [L24] ((a ((a b) (a b))) a) ((((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))) ((a ((a b) (a b))) a))
= [L29] ((a ((a b) (a b))) a) (a ((a ((a b) (a b))) a))
= [L29] a

L31 b ((a b) (a b))
= [L3] ((a b) ((a a) (((a a) b) (a a)))) ((a b) (a b))
= [L30] a b

L32 a ((a b) (a b))
= [L30] ((a b) (a a)) ((a b) (a b))
= [L30] a b

L33 a ((a b) a)
= [L32] a ((a ((a b) (a b))) a)
= [L27] a b

L34 ((a b) c) (b c)
= [L33] ((a b) c) (b ((b c) b))
= [L33] ((a ((a b) a)) c) (b ((b c) b))
= [L4] c

L35 b a
= [L31] a ((b a) (b a))
= [L33] a ((a ((b a) (b a))) a)
= [L31] a ((b a) a)

L36 b (b (a b))
= [L33] b (b (a ((a b) a)))
= [L33] b (b ((b (a ((a b) a))) b))
= [A] (((a ((a b) a)) b) (a ((a b) a))) (b ((b (a ((a b) a))) b))
= [A] (((a ((a b) a)) (((a ((a b) a)) b) (a ((a b) a)))) (a ((a b) a))) (b ((b (a ((a b) a))) b))
= [L4] a ((a b) a)
= [L33] a b

L37 (a b) a
= [L36] a (a ((a b) a))
= [L33] a (a b)

L38 (b a) a
= [L36] a (a ((b a) a))
= [L35] a (b a)

L39 (b b) (b (a b))
= [L38] (b b) ((a b) b)
= [L33] (b b) ((a b) (((a b) b) (a b)))
= [L3] (b b) ((a b) (((a b) ((a b) ((a a) (((a a) b) (a a))))) (a b)))
= [L3] (((a b) ((a a) (((a a) b) (a a)))) ((a b) ((a a) (((a a) b) (a a))))) ((a b) (((a b) ((a b) ((a a) (((a a) b) (a a))))) (a b)))
= [A] (a b) ((a a) (((a a) b) (a a)))
= [L3] b

L40 a
= [L39] (a a) (a ((b a) a))
= [L38] (a a) (a (a (b a)))
= [L36] (a a) (b a)

L41 a
= [L39] (a a) (a ((a b) a))
= [L33] (a a) (a b)

L42 b a
= [L41] b ((a a) (a b))

= L33 b ((aa) (a ((a b) a)))
= L33 b ((aa) ((aa) (a ((a b) a))) (aa)))
= A (((a a) b) (a ((a b) a))) ((aa) (((aa) (a ((a b) a))) (aa)))
= A a ((a b) a)
= L33 a b

L43 (a c) ((a b) c)
= L42 ((a b) c) (a c)
= L33 ((a b) c) (a ((a c) a))
= A c

L44 (b c) ((a b) c)
= L42 ((a b) c) (b c)
= L34 c

L46 (b a) ((a c) b)
= L42 (a) ((a c) b)
= L43 b

L46 (b a) (a (b c))
= L42 (b a) ((b c) a)
= L43 a

L47 b c
= L46 ((a b) (b c)) ((b c) ((a b) c))
= L44 ((a b) (b c)) c
= L42 c ((a b) (b c))

L48 a b
= L45 ((a b) (c a)) (((c a) b) (a b))
= L42 ((a b) (c a)) ((a b) ((c a) b))
= L44 ((a b) (c a)) b

L49 b ((a b) (c a))
= L42 ((a b) (c a)) b
= L48 a b

L50 (a b) c
= L43 (a ((a b) c)) ((a c) ((a b) c))
= L43 (a ((a b) c)) c
= L42 c (a ((a b) c))

L51 a (b (a b))
= L42 a ((a b) b)
= L42 ((a b) b) a
= L50 a ((a b) ((a b) b) a))
= L42 a ((a b) ((a b) b) a))
= L44 a a

L52 (b a) (a b)
= L50 (a b) (b ((b a) (a b)))
= L47 (a b) (a b)

L53 (a a) ((b a) (b a))
= L44 (a a) (b a) (((a a) (b a)) (((a a) (b a)) (a a)) (b a)))))
= L50 (a a) (((a a) (b a)) (a a)) (b a))
= L42 (a a) ((b a) (((a a) (b a)) (a a)))
= L42 (a a) ((b a) ((a a) (a a) (b a))))
= L40 (a a) ((b a) (a a) a))
= L42 (a a) ((b a) (a a) a))
= L47 a (a a)

L54 ((a b) (a b)) ((a b) (a b))
= L52 ((b a) (a b)) ((b a) (a b))
= L52 ((a b) (b a)) ((b a) (a b))
= L52 ((a b) (b a)) ((b a) (a b))

L55 a b
= L22 ((a b) (a b)) ((a b) (a b))
= L54 ((a b) (b a)) ((a b) (a b))
= L52 ((b a) (b a)) ((a b) (a b))

L56 a (b (b b))
= L53 a ((b b) ((a b) (a b)))
= L42 a (((a b) (a b)) (b b))
= L40 a (((a b) (a b)) (((b b) (a b)) ((b b) (a b))))
= L53 a ((a b) ((a b) (a b)))
= L42 a (((a b) (a b)) (a b))
= L32 a (((a b) (a b)) (a ((a b) (a b))))
= L61 a a

T2 a (b (b b))
= L56 a a

L57 ((a a) (((a b) (a b)) c)) ((a a) (((a b) (a b)) c))
= L56 ((a a) (((a b) (d (d d))) c)) ((a a) (((a b) (d (d d))) c))
= L56 ((a (d (d d))) (((a b) (d (d d))) c)) ((a (d (d d))) (((a b) (d (d d))) c))
= L56 ((a (d (d d))) (((a b) (d (d d))) c)) (d (d d)))
= L42 ((d (d d)) ((a (d (d d))) (((a b) (d (d d))) c)))
= L42 ((d (d d)) (((a b) (d (d d))) c)) (a (d (d d))))
= L42 (((a b) (d (d d))) c) (a (d (d d)))) (d (d d)))
= L46 (((a b) (d (d d))) c) (a (d (d d)))) (((a b) (d (d d))) ((d (d d)) ((a b) (d (d d))))) ((d (d d)) ((a b) (d (d d)))))
= L33 ((((a b) (d (d d))) c) (a ((a (d (d d))) a))) (((a b) (d (d d))) ((d (d d)) ((a b) (d (d d)))))

= A ((((a b) (d (d d))) c) (a ((a (d (d d))) a))) (((a b) (d (d d))) ((((a b) (d (d d))) (a ((a (d (d d))) a))) ((a b) (d (d d))))))
= A a ((a (d (d d))) a)
= L33 a (d (d d))

L58 (b b) (((b c) (b c)) d)
= L22 ((((b b) (((b c) (b c)) d)) ((b b) (((b c) (b c)) d))) (((b b) (((b c) (b c)) d)) ((b b) (((b c) (b c)) d)))
= L57 (b (a (a a))) (b (a (a a)))
= L56 (b b) (b b)

L59 (a a) (((a b) (a b)) c)
= L58 (a a) (a a)
= L22 a

L60 a
= L59 (a a) (((a ((b a) (c b))) (a ((b a) (c b)))) ((d ((a ((b a) (c b))) (a ((b a) (c b)))) (e d)))
= L49 (a a) (d ((a ((b a) (c b))) (a ((b a) (c b)))))
= L49 (a a) (d ((b a) (b a)))

L61 c ((a c) (((a b) c) ((a b) c)))
= L42 c ((((a b) c) ((a b) c)) (a c))
= L42 ((((a b) c) ((a b) c)) (a c)) c
= L46 ((((a b) c) ((a b) c)) (a c)) (((a b) c) (c ((a b) c)))
= L33 ((((a b) c) ((a b) c)) (a ((a c) a))) (((a b) c) (c ((a b) c)))
= A ((((a b) c) ((a b) c)) (a ((a c) a))) (((a b) c) ((((a b) c) (a ((a c) a))) ((a b) c)))
= A (a (a c) a)
= L33 a c

L62 (a a) b
= L61 b (((a a) b) ((((a a) ((a c) (a c)) d)) b) (((a a) ((a c) (a c)) d)) b)))
= L59 b (((a a) b) ((a b) (a b)))
= L50 b ((((a b) (a b)) ((a a) (((a a) b) ((a b) (a b))))))
= L45 b ((((a b) (a b)) ((((a b) (b c a)) ((a b) (b c a))) (((a a) b) ((a b) (a b))))))
= L59 b (a b)

L63 a (a b)
= L42 a (a b a)
= L62 (b b) a

L64 a (b c)
= L45 (a (b c)) c) ((c a) (a (b c)))
= L42 ((a (b c)) c) ((c a) ((b c) a))
= L44 ((a (b c)) c) a

L65 a (b c)
= L64 ((a (b c)) c) a
= L42 a (a ((b c) c))
= L42 a (c (a (b c)))

L66 a c
= L59 ((a c) (a c)) ((((a c) (c a)) ((a c) (c a))) b)
= L52 ((a c) (a c)) ((((c a) (c a)) ((c a) (c a))) b)
= L22 ((a c) (a c)) (((c a) b)

L67 (a b) (a b)
= L59 (((a b) (a b)) ((a b) (a b))) ((((((a b) (a b)) ((b a) (b a))) (((a b) (a b)) ((b a) (b a)))) c)
= L55 (((a b) (a b)) ((a b) (a b))) (((b a) (b a)) c)
= L22 ((a b) (((b a) (b a)) c)

L68 a (((b c) (b a)) ((b c) (b a)))
= L42 (((b c) (b a)) ((b c) (b a))) a
= L63 a (a ((b c) (b a)))
= L33 a (a ((b c) (b ((b a) b))))
= L33 a (a ((b c) (((b c) (b ((b a) b))) (b c))))
= A (((((b c) a) (b ((b a) b))) ((b c) (((b c) (b ((b a) b))) (b c)))) (((b c) a) (b ((b a) b))))
= A a (b ((b a) b))
= L33 a (b a)
= L62 (b b) a

L69 (b c) a
= L22 (((b c) (b c)) ((b c) (b c))) a
= L68 a ((((b c) (b c)) ((c b) (c b))) (((b c) (b c)) a) ((((b c) (b c)) ((c b) (c b))) (((b c) (b c)) a)))
= L55 a (((c b) (((b c) (b c)) a)) ((c b) (((b c) (b c)) a)))

L70 (b c) a
= L69 a (((c b) (((b c) (b c)) a)) ((c b) (((b c) (b c)) a)))
= L67 a (((c b) (c b)) ((c b) (c b)))
= L22 a (c b)

L71 ((b c) (b c)) a
= L68 a ((((b c) (c b)) ((b c) a)) (((b c) (c b)) ((b c) a)))
= L52 a ((((c b) (c b)) ((b c) a)) (((c b) (c b)) ((b c) a)))
= L66 a (c b) (c b))

L72 (b a) (((b c) a) (b c) a))
= L3 (a ((b a) (((b c) a) ((b c) a)))) ((a a) (((a a) ((b a) (((b c) a) ((b c) a)))) (a a)))
= L39 (a ((b a) (((b c) a) ((b c) a)))) ((a a) ((b a) (((b c) a) ((b c) a))))
= L61 (b a) ((a a) ((b a) (((b c) a) ((b c) a))))

L73 (b a) (((b c) a) ((b c) a))
= L72 (b a) ((a a) ((b a) (((b c) a) ((b c) a))))
= L60 (b a) a
= L70 a (a b)
= L63 (b b) a

L74 (a a) c
= L73 (a c) (((a b) c) ((a b) c))
= L50 (((a b) c) ((a b) c)) (a ((a c) (((a b) c) ((a b) c))))
= L73 (((a b) c) ((a b) c)) (a ((a a) c))
= L22 (((a b) c) ((a b) c)) (((a a) (a a)) ((a a) c))
= L41 (((a b) c) ((a b) c)) (a a)

L75 (a a) ((c (a b)) (c (a b)))
= L71 (((a b) c) ((a b) c)) (a a)
= L74 (a a) c

L76 (b (a c)) (a a)
= L22 (((b (a c)) (b (a c))) ((b (a c)) (b (a c)))) (a a))
= L63 (a a) ((a a) ((b (a c)) (b (a c))))
= L75 (a a) ((a a) b)

L77 ((a b) (a b)) (c a)
= L75 ((a b) (a b)) (((c a) ((a b) c)) ((c a) ((a b) c)))
= L45 ((a b) (a b)) (c c)

L78 ((b c) (b c)) a
= L45 ((b c) (b c)) ((a b) ((b c) a))
= L70 (((b c) a) (a b)) ((b c) (b c))
= L22 (((((b c) a) (a b)) (((b c) a) (a b))) ((((b c) a) (a b)) (((b c) a) (a b)))) ((b c) (b c))
= L63 ((b c) (b c)) (((b c) (b c)) ((((b c) a) (a b)) (((b c) a) (a b))))
= L76 (((((b c) a) (a b)) (((b c) a) (a b))) ((b c) (a b))) ((b c) (b c)))
= L70 ((b c) (b c)) (((b c) (a b)) ((((b c) a) (a b)) (((b c) a) (a b)))))
= L73 ((b c) (b c)) (((b c) (b c)) (a b)))
= L63 ((a b) (a b)) ((b c) (b c)))
= L47 ((a b) (a b)) ((c ((a b) (b c))) (c ((a b) (b c))))
= L75 ((a b) (a b)) c
= L71 c ((b a) (b a))

L79 a (c ((a b) (a b)))
= L42 a (c ((b a) (b a)))
= L78 a (((b c) (b c)) a)
= L62 (((b c) (b c)) ((b c) (b c))) a
= L22 (b c) a

L80 a ((b a) c)
= L70 (c (b a)) a
= L79 a ((b a) ((a c) (a c)))
= L42 a (((a c) (a c)) (b a))
= L77 a (((a c) (a c)) (b b))
= L78 a (c ((a (b b)) (a (b b))))
= L79 ((b b) c) a

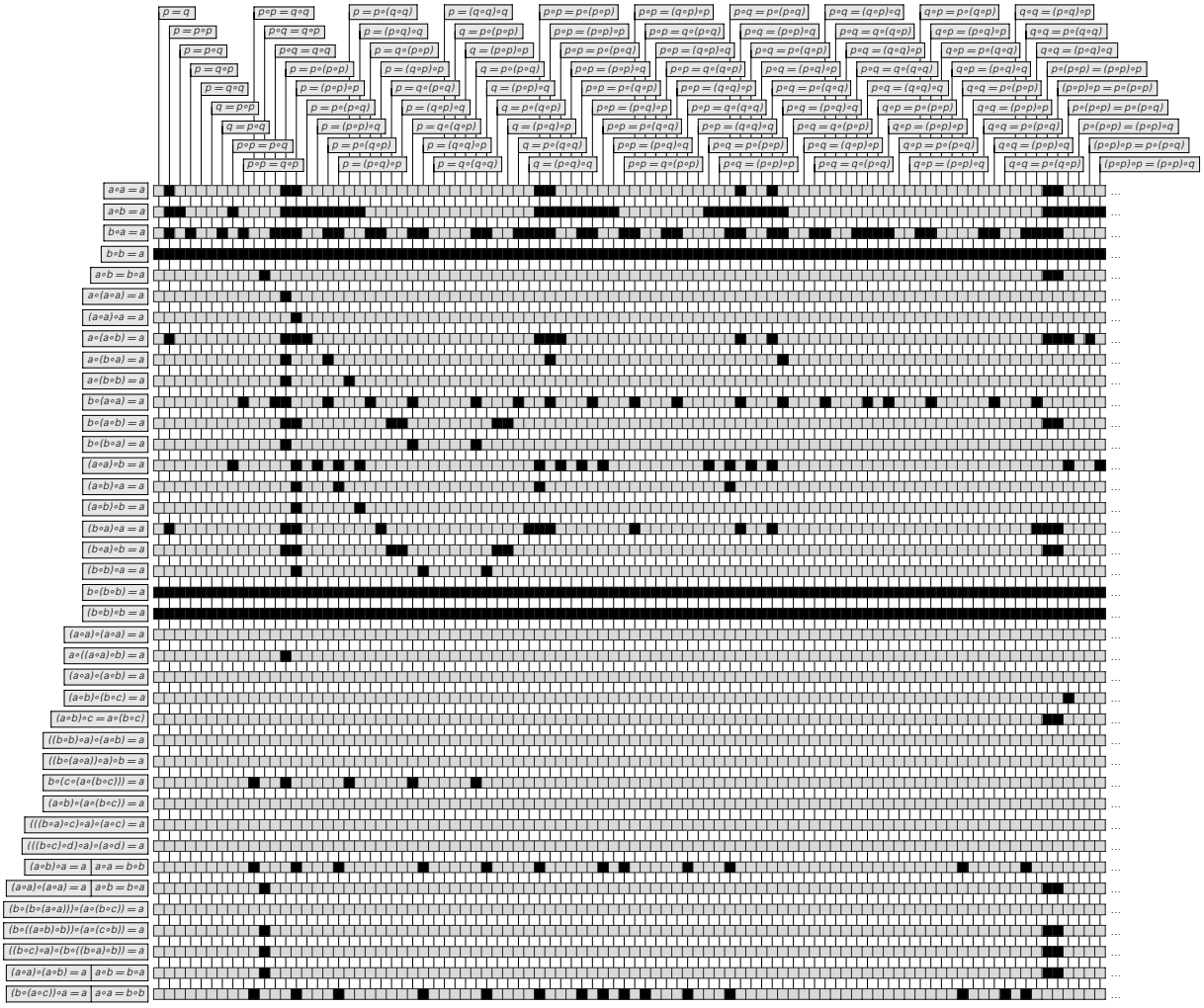
L81 ((c a) (a b)) ((c a) (a b))
= L40 ((((c a) (a b)) ((c a) (a b))) (((c a) (a b)) ((c a) (a b)))) ((a a) (((c a) (a b)) ((c a) (a b))))
= L75 ((((c a) (a b)) ((c a) (a b))) (((c a) (a b)) ((c a) (a b)))) ((a a) (c a)))
= L22 (((c a) (a b)) ((a a) (c a)))
= L70 (((a a) (c a)) ((a b) (c a)))
= L40 a ((a b) (c a))
= L70 (((c a) (a b)) a)
= L79 a ((a b) ((a c a)) (a (c a))))
= L70 a (((a (c a)) (a (c a))) (b a))
= L77 a (((a (c a)) (a (c a))) (b b))
= L78 a ((c a) ((a b b)) (a b b))))
= L79 (((b b) (c a)) a)

T3 ((b b) a) ((c c) a)
= L42 ((b b) a) (a (c c))
= L42 (a (c c)) ((b b) a)
= L22 (((a a) (a a)) (c c)) ((b b) a))
= L80 ((b b) a) (((a a) ((b b) a)) (c c))
= L70 (((c c) ((a a) ((b b) a))) ((b b) a))
= L81 (((a a) ((b b) a)) (((b b) a c)) (((a a) ((b b) a)) (((b b) a c))
= L40 (a (((b b) a c)) (a (((b b) a c))
= L80 ((((b b) (b b)) c) a) ((((b b) (b b)) c) a)
= L42 ((c ((b b) (b b))) a) ((c ((b b) (b b))) a)
= L40 ((((c ((b b) (b b))) (c ((b b) (b b)))) ((b b) (c ((b b) (b b))))) a) ((((c ((b b) (b b))) (c ((b b) (b b)))) ((b b) (c ((b b) (b b))))) a)
= L60 ((((c ((b b) (b b))) (c ((b b) (b b)))) b) a) ((((c ((b b) (b b))) (c ((b b) (b b)))) b) a)
= L42 (((b ((c ((b b) (b b))) (c ((b b) (b b))))) a) ((b ((c ((b b) (b b))) (c ((b b) (b b))))) a)
= L78 ((((c b) (c b)) ((b b) (b b))) a) ((((c b) (c b)) ((b b) (b b))) a)
= L40 ((((c b) (c b)) ((b ((b b) (c b))) (b ((b b) (c b))))) a) ((((c b) (c b)) ((b ((b b) (c b))) (b ((b b) (c b))))) a)
= L65 ((((c b) (c b)) ((b ((c b) (b ((b b) (c b))))) (b ((c b) (b ((b b) (c b))))))) a) ((((c b) (c b)) ((b ((c b) (b ((b b) (c b))))) (b ((c b) (b ((b b) (c b))))))) a)
= L75 ((((c b) (c b)) b) a) ((((c b) (c b)) b) a)
= L78 (((b ((c b) (c b))) a) ((b ((c b) (c b))) a)
= L31 ((c b) a) ((c b) a)
= L70 (a (b c)) (a (b c))

A proof that the axiom system $\{((b\circ c)\circ a)\circ(b\circ((b\circ a)\circ b)) = a\}$ given as example (g) on page 808 can reproduce the Sheffer axiom system (c), and is thus a complete axiom system for logic. The proof involves taking the original axiom A and using it to establish a sequence of lemmas Ln, from which it is eventually possible to prove the three Sheffer axioms Tn. In each part of the proof each line can be obtained from the previous one just as on page 775 by applying the axiom or lemma indicated. Explicit π operators have been omitted to allow expressions to be printed more compactly. The proof shown takes a total of 343 steps, and involves intermediate expressions with as many as 128 NANDs. It is quite possible that the proof could be considerably shortened. Note that any proof can always be recast without lemmas, but will usually then be much longer.

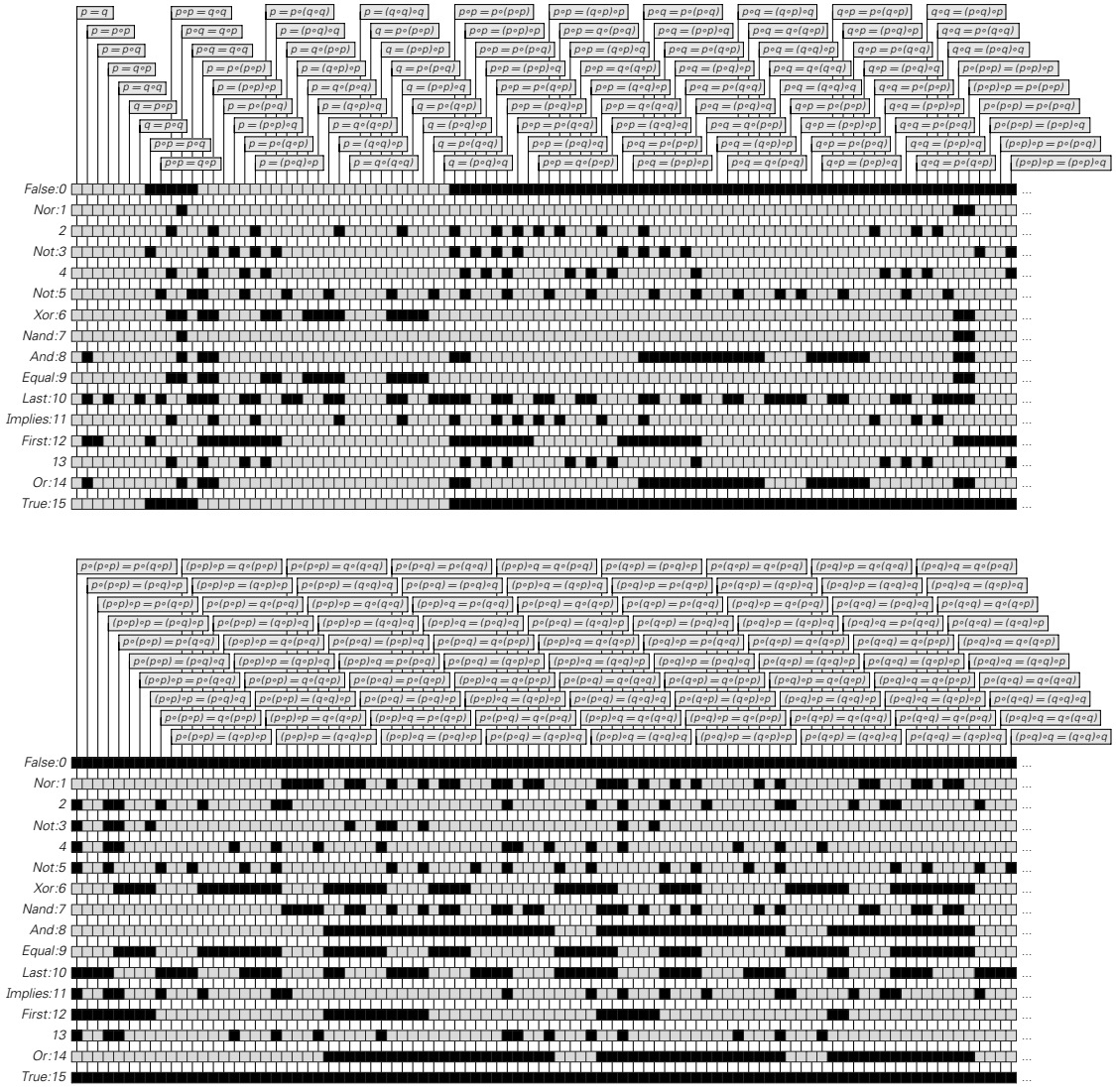
So what about patterns of theorems? Does logic somehow stand out when one looks at these? The picture below shows which possible simple equivalence theorems hold in systems from page 805.

And comparing with page 805 one sees that typically the more forms of operator are allowed by the constraints of an axiom system, the fewer equivalence results hold in that axiom system.



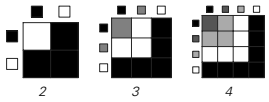
Theorems that can be proved on the basis of simple axiom systems from page 805. A black square indicates that a particular theorem holds in a particular axiom system. In general the question of whether a given theorem holds is undecidable, but the particular theorems given here happen to be simple enough that results for them can with some effort be established with certainty.

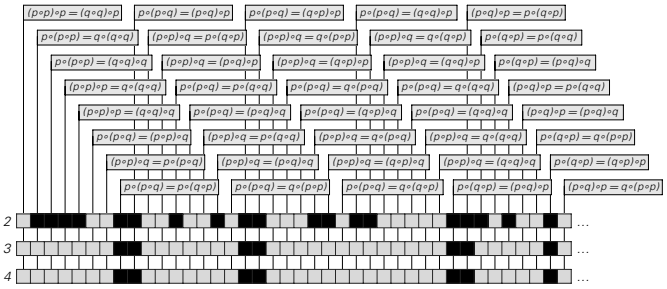
So what happens if essentially just a single form of operator is allowed? The pictures below show results for the 16 forms from page 806, and among these one sees that logic yields the fewest theorems.



Theorems that hold with operators of each of the forms shown on page 806. NAND and NOR yield the smallest number of theorems.

But if one considers for example analogs of logic for variables with more than two possible values, the picture below shows that one immediately gets systems with still fewer theorems.





Theorems in analogs of logic that allow different numbers of truth values. Statements like $p = \neg \neg p$ do not hold in general with more than 2 truth values.

So what about proofs? Is there something about these that is somehow special in the case of ordinary logic?

In the axiom systems on page 803 the typical lengths of proofs seem to increase from one system to the next, so that they end up being longest for the last axiom system, which corresponds to logic.

But if one picks a different axiom system for logic—say one of the others on page 808—then the length of a particular proof will usually change. But since one can always just start by proving the new axioms, the change can only be by a fixed amount. And as it turns out, even the simplest axiom system (f) given on page 808 seems to allow fairly short proofs of at least most short theorems.

But as one tries to prove progressively longer theorems it appears that whatever axiom system one uses for logic the lengths of proofs can increase as fast as exponentially. A crucial point, however, is that for theorems of a given length there is always a definite upper limit on the length of proof needed. Yet once again this is not something unique to logic. Indeed, it turns out that this must always be the case for any axiom system—like those on page 803—that ends up allowing essentially only operators of a single form.

So what about other axiom systems?

The very simplest ones on pages 805 and 812 seem to yield proofs that are always comparatively short. But when one looks at axiom systems that are even slightly more complicated the proofs of anything

but the simplest results can get much longer—making it in practice often difficult to tell whether a given result can actually even be proved at all.

And this is in a sense just another example of the same basic phenomenon that we already saw early in this section in multiway systems, and that often seems to occur in real mathematics: that even if a theorem is short to state, its proof can be arbitrarily long.

And this I believe is ultimately a reflection of the Principle of Computational Equivalence. For the principle suggests that most axiom systems whose consequences are not obviously simple will tend to be universal. And this means that they will exhibit computational irreducibility and undecidability—and will allow no general upper limit to be placed on how long a proof could be needed for any given result.

As I discussed earlier, most of the common axiom systems in traditional mathematics are known to be universal—basic logic being one of the few exceptions. But one might have assumed that to achieve their universality these axiom systems would have to be specially set up with all sorts of specific sophisticated features.

Yet from the results of this book—as embodied in the Principle of Computational Equivalence—we now know that this is not the case, and that in fact universality should already be rather common even among very simple axiom systems, like those on page 805.

And indeed, while operator systems and multiway systems have many superficial differences, I suspect that when it comes to universality they work very much the same. So in either idealization, one should not have to go far to get axiom systems that exhibit universality—just like most of the ones in traditional mathematics.

But once one has reached an axiom system that is universal, why should one in a sense ever have to go further? After all, what it means for an axiom system to be universal is that by setting up a suitable encoding it must in principle be possible to make that axiom system reproduce any other possible axiom system.

But the point is that the kinds of encodings that are normally used in mathematics are in practice rather limited. For while it is common, say, to take a problem in geometry and reformulate it as a problem in algebra, this is almost always done just by setting up a direct

translation between the objects one is describing—usually in effect just by renaming the operators used to manipulate them.

Yet to take full advantage of universality one must consider not only translations between objects but also translations between complete proofs. And if one does this it is indeed perfectly possible, say, to program arithmetic to reproduce any proof in set theory. In fact, all one need do is to encode the axioms of set theory in something like the arithmetic equation system of page 786.

But with the notable exception of Gödel's Theorem these kinds of encodings are not normally used in mathematics. So this means that even when universality is present realistic idealizations of mathematics must still distinguish different axiom systems.

So in the end what is it that determines which axiom systems are actually used in mathematics? In the course of this section I have discussed a few criteria. But in the end history seems to be the only real determining factor. For given almost any general property that one can pick out in axiom systems like those on pages 773 and 774 there typically seem to be all sorts of operator and multiway systems—often including some rather simple ones—that share the exact same property.

So this leads to the conclusion that there is in a sense nothing fundamentally special about the particular axiom systems that have traditionally been used in mathematics—and that in fact there are all sorts of other axiom systems that could perfectly well be used as foundations for what are in effect new fields of mathematics—just as rich as the traditional ones, but without the historical connections.

So what about existing fields of mathematics? As I mentioned earlier in this section, I strongly believe that even within these there are fundamental limitations that have implicitly been imposed on what has actually been studied. And most often what has happened is that there are only certain kinds of questions or statements that have been considered of real mathematical interest.

The picture on the facing page shows a rather straightforward version of this. It lists in order a large number of theorems from basic logic, highlighting just those few that are considered interesting enough by typical textbooks of logic to be given explicit names.

1 $a = a \wedge a$	2 $a = a \vee a$	$a \wedge a = a \vee a$	3 $a \wedge b = b \wedge a$	4 $a \vee b = b \vee a$	5 $a = \neg \neg a$
$a \wedge a = \neg \neg a$	$a \vee a = \neg \neg a$	$\neg a = \neg (a \wedge a)$	$\neg a = \neg (a \vee a)$	$a = (a \wedge a) \wedge a$	$a = (a \vee a) \wedge a$
$a = a \wedge (a \wedge a)$	$a = a \wedge (a \vee a)$	$a = a \wedge a \vee a$	$a = (a \vee a) \vee a$	$a = a \vee a \wedge a$	$a = a \vee (a \vee a)$
6 $a = a \wedge (a \vee b)$	7 $a = a \vee a \wedge b$	$a = (a \vee b) \wedge a$	$a = a \wedge (b \vee a)$	$a = a \wedge b \vee a$	$a = a \vee b \wedge a$
$a = (b \vee a) \wedge a$	$a = b \wedge a \vee a$	$\neg a = \neg a \wedge \neg a$	$\neg a = \neg a \vee \neg a$	$\neg a \wedge a = a \wedge \neg a$	$\neg a \vee a = a \vee \neg a$
$\neg (a \wedge a) = \neg (a \vee a)$	$\neg \neg a = (a \wedge a) \wedge a$	$\neg \neg a = (a \vee a) \wedge a$	$\neg \neg a = a \wedge (a \wedge a)$	$\neg \neg a = a \wedge (a \vee a)$	$\neg \neg a = a \wedge a \vee a$
$\neg \neg a = (a \vee a) \vee a$	$\neg \neg a = a \vee a \wedge a$	$\neg \neg a = a \vee (a \vee a)$	$\neg \neg a = a \wedge (a \vee b)$	$\neg \neg a = a \vee a \wedge b$	$\neg \neg a = (a \vee b) \wedge a$
$\neg \neg a = a \wedge (b \vee a)$	$\neg \neg a = a \wedge b \vee a$	$\neg \neg a = a \vee b \wedge a$	8 $\neg a \wedge a = \neg b \wedge b$	$a \wedge \neg a = \neg b \wedge b$	$\neg a \wedge a = b \wedge \neg b$
$a \wedge \neg a = b \wedge \neg b$	9 $\neg a \vee a = \neg b \vee b$	$a \vee \neg a = \neg b \vee b$	$\neg a \vee a = b \vee \neg b$	$a \vee \neg a = b \vee \neg b$	$\neg \neg a = (b \vee a) \wedge a$
$\neg \neg a = b \wedge a \vee a$	$a \wedge \neg b = \neg b \wedge a$	$\neg a \wedge b = b \wedge \neg a$	$a \vee \neg b = \neg b \vee a$	$\neg a \vee b = b \vee \neg a$	$\neg (a \wedge b) = \neg (b \wedge a)$
$\neg (a \vee b) = \neg (b \vee a)$	$a \wedge a = (a \wedge a) \wedge a$	$a \vee a = (a \wedge a) \wedge a$	$a \wedge a = (a \vee a) \wedge a$	$a \vee a = (a \vee a) \wedge a$	$a \wedge a = a \wedge (a \wedge a)$
$a \vee a = a \wedge (a \wedge a)$	$a \wedge a = a \wedge (a \vee a)$	$a \vee a = a \wedge (a \vee a)$	$a \wedge a = a \wedge a \vee a$	$a \vee a = a \wedge a \vee a$	$a \wedge a = (a \vee a) \vee a$
$a \vee a = (a \vee a) \vee a$	$a \wedge a = a \vee a \wedge a$	$a \vee a = a \vee a \wedge a$	$a \wedge a = a \vee (a \vee a)$	$a \vee a = a \vee (a \vee a)$	$a \wedge a = a \wedge (a \vee b)$
$a \vee a = a \wedge (a \vee b)$	$a \wedge a = a \vee a \wedge b$	$a \vee a = a \vee a \wedge b$	$a \wedge a = (a \vee b) \wedge a$	$a \vee a = (a \vee b) \wedge a$	$a \wedge a = a \wedge (b \vee a)$
$a \vee a = a \wedge (b \vee a)$	$a \wedge a = a \wedge b \vee a$	$a \vee a = a \wedge b \vee a$	$a \wedge a = a \vee b \wedge a$	$a \vee a = a \vee b \wedge a$	$a \wedge a = (b \vee a) \wedge a$
$a \vee a = (b \vee a) \wedge a$	$a \wedge a = b \wedge a \vee a$	$a \vee a = b \wedge a \vee a$	$a \wedge b = (a \wedge a) \wedge b$	$a \wedge b = (a \vee a) \wedge b$	$a \wedge b = a \wedge (a \wedge b)$
$a \vee b = a \wedge a \vee b$	$a \vee b = (a \vee a) \vee b$	$a \vee b = a \vee (a \vee b)$	$a \wedge b = (a \wedge b) \wedge a$	$a \wedge b = a \wedge (b \wedge a)$	$a \vee b = (a \vee b) \vee a$
$a \vee b = a \vee (b \vee a)$	$a \wedge b = (a \wedge b) \wedge b$	$a \wedge b = a \wedge (b \wedge b)$	$a \wedge b = a \wedge (b \vee b)$	$a \vee b = (a \vee b) \vee b$	$a \vee b = a \vee b \wedge b$
$a \vee b = a \vee (b \vee b)$	$a \wedge b = (b \wedge a) \wedge a$	$a \wedge b = b \wedge (a \wedge a)$	$a \wedge b = b \wedge (a \vee a)$	$a \vee b = (b \vee a) \vee a$	$a \vee b = b \vee a \wedge a$
$a \vee b = b \vee (a \vee a)$	$a \wedge b = (b \wedge a) \wedge b$	$a \wedge b = b \wedge (a \wedge b)$	$a \vee b = (b \vee a) \vee b$	$a \vee b = b \vee (a \vee b)$	$a \wedge b = (b \wedge b) \wedge a$
$a \wedge b = (b \vee b) \wedge a$	$a \wedge b = b \wedge (b \wedge a)$	$a \vee b = b \wedge b \vee a$	$a \vee b = (b \vee b) \vee a$	$a \vee b = b \vee (b \vee a)$	$\neg (a \wedge a) = \neg a \wedge \neg a$
$\neg (a \vee a) = \neg a \wedge \neg a$	$\neg (a \wedge a) = \neg a \vee \neg a$	$\neg (a \vee a) = \neg a \vee \neg a$	10 $\neg (a \vee b) = \neg a \wedge \neg b$	$\neg (a \wedge b) = \neg a \vee \neg b$	$\neg (a \vee b) = \neg b \wedge \neg a$
$\neg (a \wedge b) = \neg b \vee \neg a$	$\neg a \wedge \neg a = \neg a \vee \neg a$	$\neg a \wedge \neg b = \neg b \wedge \neg a$	$\neg a \vee \neg b = \neg b \vee \neg a$	$(a \wedge a) \wedge a = (a \vee a) \wedge a$	$(a \wedge a) \wedge a = a \wedge (a \wedge a)$
$(a \vee a) \wedge a = a \wedge (a \wedge a)$	$(a \wedge a) \wedge a = a \wedge (a \vee a)$	$(a \vee a) \wedge a = a \wedge (a \vee a)$	$a \wedge (a \wedge a) = a \wedge (a \vee a)$	$(a \wedge a) \wedge a = a \wedge a \vee a$	$(a \vee a) \wedge a = a \wedge a \vee a$
$a \wedge (a \wedge a) = a \wedge a \vee a$	$a \wedge (a \vee a) = a \wedge a \vee a$	$(a \wedge a) \wedge a = (a \vee a) \vee a$	$(a \vee a) \wedge a = (a \vee a) \vee a$	$a \wedge (a \wedge a) = (a \vee a) \vee a$	$a \wedge (a \vee a) = (a \vee a) \vee a$
$a \wedge a \vee a = (a \vee a) \vee a$	$(a \wedge a) \wedge a = a \vee a \wedge a$	$(a \vee a) \wedge a = a \vee a \wedge a$	$a \wedge (a \wedge a) = a \vee a \wedge a$	$a \wedge (a \vee a) = a \vee a \wedge a$	$a \wedge a \vee a = a \vee a \wedge a$
$(a \vee a) \vee a = a \vee a \wedge a$	$(a \wedge a) \wedge a = a \vee (a \vee a)$	$(a \vee a) \wedge a = a \vee (a \vee a)$	$a \wedge (a \wedge a) = a \vee (a \vee a)$	$a \wedge (a \vee a) = a \vee (a \vee a)$	$a \wedge a \vee a = a \vee (a \vee a)$
$(a \vee a) \vee a = a \vee (a \vee a)$	$a \vee a \wedge a = a \vee (a \vee a)$	$(a \wedge a) \wedge a = a \wedge (a \vee b)$	$(a \vee a) \wedge a = a \wedge (a \vee b)$	$a \wedge (a \wedge a) = a \wedge (a \vee b)$	$a \wedge (a \vee a) = a \wedge (a \vee b)$
$a \wedge a \vee a = a \wedge (a \vee b)$	$(a \vee a) \vee a = a \wedge (a \vee b)$	$a \vee a \wedge a = a \wedge (a \vee b)$	$a \vee (a \vee a) = a \wedge (a \vee b)$	$(a \wedge a) \wedge a = a \vee a \wedge b$	$(a \vee a) \wedge a = a \vee a \wedge b$
$a \wedge (a \wedge a) = a \vee a \wedge b$	$a \wedge (a \vee a) = a \vee a \wedge b$	$a \wedge a \vee a = a \vee a \wedge b$	$(a \vee a) \vee a = a \vee a \wedge b$	$a \vee a \wedge a = a \vee a \wedge b$	$a \vee (a \vee a) = a \vee a \wedge b$

⋮ *50 lines*

$a \wedge b \vee b = b \wedge (b \vee a)$	$(a \vee b) \vee b = b \wedge b \vee a$	$a \vee b \wedge b = b \wedge b \vee a$	$a \vee (b \vee b) = b \wedge b \vee a$	$(a \vee b) \vee b = (b \vee b) \vee a$	$a \vee b \wedge b = (b \vee b) \vee a$
$a \vee (b \vee b) = (b \vee b) \vee a$	$(a \vee b) \wedge b = b \vee b \wedge a$	$a \wedge b \vee b = b \vee b \wedge a$	$(a \vee b) \vee b = b \vee (b \vee a)$	$a \vee b \wedge b = b \vee (b \vee a)$	$a \vee (b \vee b) = b \vee (b \vee a)$
$(a \vee b) \wedge b = (b \wedge b) \wedge b$	$a \wedge b \vee b = (b \wedge b) \wedge b$	$(a \vee b) \wedge b = (b \vee b) \wedge b$	$a \wedge b \vee b = (b \vee b) \wedge b$	$(a \vee b) \wedge b = b \wedge (b \wedge b)$	$a \wedge b \vee b = b \wedge (b \wedge b)$
$(a \vee b) \wedge b = b \wedge (b \vee b)$	$a \wedge b \vee b = b \wedge (b \vee b)$	$(a \vee b) \wedge b = b \wedge b \vee b$	$a \wedge b \vee b = b \wedge b \vee b$	$(a \vee b) \wedge b = (b \vee b) \vee b$	$a \wedge b \vee b = (b \vee b) \vee b$
$(a \vee b) \wedge b = b \vee b \wedge b$	$a \wedge b \vee b = b \vee b \wedge b$	$(a \vee b) \wedge b = b \vee (b \vee b)$	$a \wedge b \vee b = b \vee (b \vee b)$	$(a \vee b) \wedge b = b \wedge (b \vee c)$	$a \wedge b \vee b = b \wedge (b \vee c)$
$(a \vee b) \wedge b = b \vee b \wedge c$	$a \wedge b \vee b = b \vee b \wedge c$	$(a \vee b) \wedge b = (b \vee c) \wedge b$	$a \wedge b \vee b = (b \vee c) \wedge b$	$(a \vee b) \wedge b = b \wedge (c \vee b)$	$a \wedge b \vee b = b \wedge (c \vee b)$
$(a \vee b) \wedge b = b \wedge c \vee b$	$a \wedge b \vee b = b \wedge c \vee b$	$(a \vee b) \wedge b = b \vee c \wedge b$	$a \wedge b \vee b = b \vee c \wedge b$	$(a \vee b) \wedge b = (c \vee b) \wedge b$	$a \wedge b \vee b = (c \vee b) \wedge b$
$(a \vee b) \wedge b = c \wedge b \vee b$	$a \wedge b \vee b = c \wedge b \vee b$	11 $(a \wedge b) \wedge c = a \wedge (b \wedge c)$	12 $(a \vee b) \vee c = a \vee (b \vee c)$	$(a \wedge b) \wedge c = (a \wedge c) \wedge b$	$a \wedge (b \wedge c) = (a \wedge c) \wedge b$
$(a \wedge b) \wedge c = a \wedge (c \wedge b)$	$a \wedge (b \wedge c) = a \wedge (c \wedge b)$	$a \wedge (b \vee c) = a \wedge (c \vee b)$	$(a \vee b) \vee c = (a \vee c) \vee b$	$a \vee (b \vee c) = (a \vee c) \vee b$	$a \vee b \wedge c = a \vee c \wedge b$
$(a \vee b) \vee c = a \vee (c \vee b)$	$a \vee (b \vee c) = a \vee (c \vee b)$	$(a \wedge b) \wedge c = (b \wedge a) \wedge c$	$a \wedge (b \wedge c) = (b \wedge a) \wedge c$	$(a \vee b) \wedge c = (b \vee a) \wedge c$	$(a \wedge b) \wedge c = b \wedge (a \wedge c)$

⋮ *392 lines*

$a \wedge (b \vee c) = (a \wedge a) \wedge (c \vee b)$	$a \wedge (b \vee c) = (a \vee a) \wedge (c \vee b)$	$a \vee b \wedge c = a \wedge a \vee c \wedge b$	$a \vee b \wedge c = (a \vee a) \vee c \wedge b$	$(a \vee b) \vee c = a \wedge a \vee (c \vee b)$
$(a \vee b) \vee c = (a \vee a) \vee (c \vee b)$	$a \vee (b \vee c) = a \wedge a \vee (c \vee b)$	$a \vee (b \vee c) = (a \vee a) \vee (c \vee b)$	$(a \wedge b) \wedge c = (a \wedge b) \wedge (a \wedge c)$	$a \wedge (b \wedge c) = (a \wedge b) \wedge (a \wedge c)$
13 $a \wedge (b \vee c) = a \wedge b \vee a \wedge c$	14 $a \vee b \wedge c = (a \vee b) \wedge (a \vee c)$	$(a \vee b) \vee c = (a \vee b) \vee (a \vee c)$	$a \vee (b \vee c) = (a \vee b) \vee (a \vee c)$	$(a \wedge b) \wedge c = (a \wedge b) \wedge (b \wedge c)$
$a \wedge (b \wedge c) = (a \wedge b) \wedge (b \wedge c)$	$(a \vee b) \vee c = (a \vee b) \vee (b \vee c)$	$a \vee (b \vee c) = (a \vee b) \vee (b \vee c)$	$(a \wedge b) \wedge c = (a \wedge b) \wedge (c \wedge a)$	$a \wedge (b \wedge c) = (a \wedge b) \wedge (c \wedge a)$

⋮

The theorems of basic logic written out in order of increasing complexity. Those considered interesting enough to name in typical textbooks are highlighted. The theorems are respectively: (1), (2) idempotence (laws of tautology) of AND and OR, (3), (4) commutativity of AND and OR, (5) law of double negation, (6), (7) absorption (redundancy) laws, (8) law of noncontradiction (definition of FALSE), (9) law of excluded middle (definition of TRUE), (10) de Morgan's law, (11), (12) associativity of AND and OR, (13), (14) distributive laws. With the exception of the second distributive law, it turns out that the highlighted theorems are exactly the ones that cannot be derived from preceding theorems in the list. The distributive laws appear at positions 2813 and 2814 in the list; it takes a long proof to obtain the second one from preceding theorems.

But what determines which theorems these will be? One might have thought that it would be purely a matter of history. But actually looking at the list of theorems it always seems that the interesting ones are in a sense those that show the least unnecessary complication.

And indeed if one starts from the beginning of the list one finds that most of the theorems can readily be derived from simpler ones earlier in the list. But there are a few that cannot—and that therefore provide in a sense the simplest statements of genuinely new information. And remarkably enough what I have found is that these theorems are almost exactly the ones highlighted on the previous page that have traditionally been identified as interesting.

So what happens if one applies the same criterion in other settings? The picture below shows as an example theorems from the formulation of logic discussed above based on NAND.

$a \barwedge b = b \barwedge a$	$a = (a \barwedge a) \barwedge (a \barwedge a)$	$a = (a \barwedge a) \barwedge (a \barwedge b)$	$a = (a \barwedge a) \barwedge (b \barwedge a)$
$a = (a \barwedge b) \barwedge (a \barwedge a)$	$a = (b \barwedge a) \barwedge (a \barwedge a)$	$(a \barwedge a) \barwedge a = a \barwedge (a \barwedge a)$	$(a \barwedge a) \barwedge a = (b \barwedge b) \barwedge b$
$a \barwedge (a \barwedge a) = (b \barwedge b) \barwedge b$	$(a \barwedge a) \barwedge a = b \barwedge (b \barwedge b)$	$a \barwedge (a \barwedge a) = b \barwedge (b \barwedge b)$	$a \barwedge (a \barwedge b) = (a \barwedge b) \barwedge a$
$a \barwedge (a \barwedge b) = a \barwedge (b \barwedge a)$	$(a \barwedge a) \barwedge b = (a \barwedge b) \barwedge b$	$a \barwedge (a \barwedge b) = a \barwedge (b \barwedge b)$	$a \barwedge (a \barwedge b) = (b \barwedge a) \barwedge a$
$(a \barwedge a) \barwedge b = b \barwedge (a \barwedge a)$	$(a \barwedge a) \barwedge b = (b \barwedge a) \barwedge b$	$(a \barwedge a) \barwedge b = b \barwedge (a \barwedge b)$	$a \barwedge (a \barwedge b) = (b \barwedge b) \barwedge a$
$(a \barwedge a) \barwedge b = b \barwedge (b \barwedge a)$	$(a \barwedge b) \barwedge a = a \barwedge (b \barwedge a)$	$(a \barwedge b) \barwedge a = a \barwedge (b \barwedge b)$	$a \barwedge (b \barwedge a) = a \barwedge (b \barwedge b)$
$(a \barwedge b) \barwedge a = (b \barwedge a) \barwedge a$	$a \barwedge (b \barwedge a) = (b \barwedge a) \barwedge a$	$(a \barwedge b) \barwedge a = (b \barwedge b) \barwedge a$	$a \barwedge (b \barwedge a) = (b \barwedge b) \barwedge a$
$a \barwedge (b \barwedge b) = (b \barwedge a) \barwedge a$	$(a \barwedge b) \barwedge b = b \barwedge (a \barwedge a)$	$(a \barwedge b) \barwedge b = (b \barwedge a) \barwedge b$	$(a \barwedge b) \barwedge b = b \barwedge (a \barwedge b)$
$a \barwedge (b \barwedge b) = (b \barwedge b) \barwedge a$	$(a \barwedge b) \barwedge b = b \barwedge (b \barwedge a)$	$a \barwedge (b \barwedge c) = a \barwedge (c \barwedge b)$	$(a \barwedge b) \barwedge c = b \barwedge (a \barwedge c)$
$a \barwedge (b \barwedge c) = (b \barwedge c) \barwedge a$	$(a \barwedge b) \barwedge c = c \barwedge (a \barwedge b)$	$a \barwedge (b \barwedge c) = (c \barwedge b) \barwedge a$	$(a \barwedge b) \barwedge c = c \barwedge (b \barwedge a)$
$(a \barwedge a) \barwedge (a \barwedge a) = (a \barwedge a) \barwedge (a \barwedge b)$	$(a \barwedge a) \barwedge (a \barwedge a) = (a \barwedge a) \barwedge (b \barwedge a)$	$(a \barwedge a) \barwedge (a \barwedge a) = (a \barwedge b) \barwedge (a \barwedge a)$	$(a \barwedge a) \barwedge (a \barwedge a) = (b \barwedge a) \barwedge (a \barwedge a)$
$(a \barwedge a) \barwedge (a \barwedge b) = (a \barwedge a) \barwedge (a \barwedge c)$	$(a \barwedge a) \barwedge (a \barwedge b) = (a \barwedge a) \barwedge (b \barwedge a)$	$(a \barwedge a) \barwedge (a \barwedge b) = (a \barwedge a) \barwedge (c \barwedge a)$	$(a \barwedge a) \barwedge (a \barwedge b) = (a \barwedge b) \barwedge (a \barwedge a)$

⋮ *118 lines*

$a \barwedge ((a \barwedge b) \barwedge b) = (c \barwedge (a \barwedge a)) \barwedge a$	$a \barwedge ((a \barwedge b) \barwedge b) = ((c \barwedge a) \barwedge c) \barwedge a$	$a \barwedge ((a \barwedge b) \barwedge b) = (c \barwedge (a \barwedge c)) \barwedge a$	$(a \barwedge (a \barwedge b)) \barwedge b = (c \barwedge (b \barwedge b)) \barwedge b$
$(a \barwedge (a \barwedge b)) \barwedge b = ((c \barwedge b) \barwedge c) \barwedge b$	$(a \barwedge (a \barwedge b)) \barwedge b = (c \barwedge (b \barwedge c)) \barwedge b$	$a \barwedge ((a \barwedge b) \barwedge b) = (c \barwedge (c \barwedge a)) \barwedge a$	$(a \barwedge (a \barwedge b)) \barwedge b = (c \barwedge (c \barwedge b)) \barwedge b$
$a \barwedge ((a \barwedge b) \barwedge b) = ((c \barwedge c) \barwedge c) \barwedge a$	$a \barwedge ((a \barwedge b) \barwedge b) = (c \barwedge (c \barwedge c)) \barwedge a$	$(a \barwedge (a \barwedge b)) \barwedge b = ((c \barwedge c) \barwedge c) \barwedge b$	$(a \barwedge (a \barwedge b)) \barwedge b = (c \barwedge (c \barwedge c)) \barwedge b$
$a \barwedge (a \barwedge (b \barwedge c)) = a \barwedge (a \barwedge (c \barwedge b))$	$(a \barwedge (a \barwedge b)) \barwedge c = ((a \barwedge b) \barwedge a) \barwedge c$	$(a \barwedge (a \barwedge b)) \barwedge c = (a \barwedge (b \barwedge a)) \barwedge c$	$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge ((b \barwedge a) \barwedge c)$
$((a \barwedge a) \barwedge b) \barwedge c = ((a \barwedge b) \barwedge b) \barwedge c$	$(a \barwedge (a \barwedge b)) \barwedge c = (a \barwedge (b \barwedge b)) \barwedge c$	$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge ((b \barwedge b) \barwedge c)$	$a \barwedge ((a \barwedge b) \barwedge c) = ((a \barwedge b) \barwedge c) \barwedge a$
$a \barwedge (a \barwedge (b \barwedge c)) = (a \barwedge (b \barwedge c)) \barwedge a$	$a \barwedge (a \barwedge (b \barwedge c)) = a \barwedge ((b \barwedge c) \barwedge a)$	$a \barwedge (a \barwedge (b \barwedge c)) = ((a \barwedge b) \barwedge c) \barwedge c$	$(a \barwedge (a \barwedge b)) \barwedge c = (a \barwedge (b \barwedge c)) \barwedge c$
$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge ((b \barwedge c) \barwedge c)$	$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge (c \barwedge (a \barwedge b))$	$a \barwedge (a \barwedge (b \barwedge c)) = (a \barwedge (c \barwedge b)) \barwedge a$	$a \barwedge (a \barwedge (b \barwedge c)) = a \barwedge ((c \barwedge b) \barwedge a)$
$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge (c \barwedge (b \barwedge a))$	$a \barwedge (a \barwedge (b \barwedge c)) = ((a \barwedge c) \barwedge b) \barwedge b$	$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge (c \barwedge (b \barwedge b))$	$((a \barwedge a) \barwedge b) \barwedge c = (a \barwedge c) \barwedge (b \barwedge c)$
$(a \barwedge (a \barwedge b)) \barwedge c = (a \barwedge (c \barwedge b)) \barwedge c$	$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge ((c \barwedge b) \barwedge c)$	$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge (c \barwedge (b \barwedge c))$	$a \barwedge ((a \barwedge b) \barwedge c) = a \barwedge (c \barwedge (c \barwedge b))$

⋮

The theorems of logic formulated in terms of NAND. Theorems which cannot be derived from ones earlier in the list are highlighted. The last highlighted theorem is 539th in the list. No later theorems would be highlighted since the ones shown form a complete axiom system from which any theorem of logic can be derived. The last highlighted theorem is however an example of one that follows from the axioms, but is hard to prove.

Now there is no particular historical tradition to rely on. But the criterion nevertheless still seems to agree rather well with judgements a human might make. And much as in the picture on page 817, what one sees is that right at the beginning of the list there are several theorems that are identified as interesting. But after these one has to go a long way before one finds other ones.

So if one were to go still further, would one eventually find yet more? It turns out that with the criterion we have used one would not. And the reason is that just the six theorems highlighted already happen to form an axiom system from which any possible theorem about NANDs can ultimately be derived.

And indeed, whenever one is dealing with theorems that can be derived from a finite axiom system the criterion implies that only a finite number of theorems should ever be considered interesting—ending as soon as one has in a sense got enough theorems to be able to reproduce some formulation of the axiom system.

But this is essentially like saying that once one knows the rules for a system nothing else about it should ever be considered interesting. Yet most of this book is concerned precisely with all the interesting behavior that can emerge even if one knows the rules for a system.

And the point is that if computational irreducibility is present, then there is in a sense all sorts of information about the behavior of a system that can only be found from its rules by doing an irreducibly large amount of computational work. And the analog of this in an axiom system is that there are theorems that can be reached only by proofs that are somehow irreducibly long.

So what this suggests is that a theorem might be considered interesting not only if it cannot be derived at all from simpler theorems but also if it cannot be derived from them except by some long proof. And indeed in basic logic the last theorem identified as interesting on page 817—the distributivity of OR—is an example of one that can in principle be derived from earlier theorems, but only by a proof that seems to be much longer than other theorems of comparable size.

In logic, however, all proofs are in effect ultimately of limited length. But in any axiom system where there is universality—and thus

undecidability—this is no longer the case, and as I discussed above I suspect that it will actually be quite common for there to be all sorts of short theorems that have only extremely long proofs.

No doubt many such theorems are much too difficult ever to prove in practice. But even if they could be proved, would they be considered interesting? Certainly they would provide what is in essence new information, but my strong suspicion is that in mathematics as it is currently practiced they would only rarely be considered interesting.

And most often the stated reason for this would be that they do not seem to fit into any general framework of mathematical results, but instead just seem like isolated random mathematical facts.

In doing mathematics, it is common to use terms like difficult, powerful, surprising and deep to describe theorems. But what do these really mean? As I mentioned above, any field of mathematics can at some level be viewed as a giant network of statements in which the connections correspond to theorems. And my suspicion is that our intuitive characterizations of theorems are in effect just reflections of our perception of various features of the structure of this network.

And indeed I suspect that by looking at issues such as how easy a given theorem makes it to get from one part of a network to another it will be possible to formalize many intuitive notions about the practice of mathematics—much as earlier in this book we were able to formalize notions of everyday experience such as complexity and randomness.

Different fields of mathematics may well have networks with characteristically different features. And so, for example, what are usually viewed as more successful areas of pure mathematics may have more compact networks, while areas that seem to involve all sorts of isolated facts—like elementary number theory or theory of specific cellular automata—may have sparser networks with more tendrils.

And such differences will be reflected in proofs that can be given. For example, in a sparser network the proof of a particular theorem may not contain many pieces that can be used in proving other theorems. But in a more compact network there may be intermediate definitions and concepts that can be used in a whole range of different theorems.

Indeed, in an extreme case it might even be possible to do the analog of what has been done, say, in the computation of symbolic integrals, and to set up some kind of uniform procedure for finding a proof of essentially any short theorem.

And in general whenever there are enough repeated elements within a single proof or between different proofs this indicates the presence of computational reducibility. Yet while this means that there is in effect less new information in each theorem that is proved, it turns out that in most areas of mathematics these theorems are usually the ones that are considered interesting.

The presence of universality implies that there must at some level be computational irreducibility—and thus that there must be theorems that cannot be reached by any short procedure. But the point is that mathematics has tended to ignore these, and instead to concentrate just on what are in effect limited patches of computational reducibility in the network of all possible theorems.

Yet in a sense this is no different from what has happened, say, in physics, where the phenomena that have traditionally been studied are mostly just those ones that show enough computational reducibility to allow analysis by traditional methods of theoretical physics.

But whereas in physics one has only to look at the natural world to see that other more complex phenomena exist, the usual approaches to mathematics provide almost no hint of anything analogous.

Yet with the new approach based on explicit experimentation used in this book it now becomes quite clear that phenomena such as computational irreducibility occur in abstract mathematical systems.

And indeed the Principle of Computational Equivalence implies that such phenomena should be close at hand in almost every direction: it is merely that—despite its reputation for generality—mathematics has in the past implicitly tended to define itself to avoid them.

So what this means is that in the future, when the ideas and methods of this book have successfully been absorbed, the field of mathematics as it exists today will come to be seen as a small and surprisingly uncharacteristic sample of what is actually possible.