Analogies and Metaphors to Explain Gödel’s Theorem

Douglas R. Hofstadter

When I was a graduate student at Berkeley in mathematics during 1966 and 1967, I found out, to my chagrin, that mathematics was too abstract for me. I had always thought that I was a pretty abstract thinker, but what I began to realize about that time in my life was that, in fact, all of my thoughts are very concrete. They all are based on images, analogies and metaphors. I really think only in concrete ideas, and I found that I couldn’t attach any concrete ideas to some of the mathematics I was learning. I could learn the formal statements and theorems, I could prove theorems formally, but I really could not go beyond them. I was just not able to get the concepts without images, so I turned away from mathematics and went on to physics at the University of Oregon. Then my career went through variegated phases, and finally I wound up in computer science and artificial intelligence, which is not exactly an accident because my greatest interest in artificial intelligence nowadays is in understanding analogies. In a way I have come back to study, through computer science (and particularly through the branch of it called artificial intelligence), what these analogies are that I think with. So there is a little loop there that I have closed, and I feel very happy doing it.

My book* illustrates particularly strongly my own predilection for thinking in metaphors, analogies, images, and so forth. Many of the images that I have produced in the book are connected with Gödel’s theorem. Some of you may know and some of you may not know exactly what Gödel’s theorem is. I will try to give you a flavor of what it is in this talk.† It is not intended just for you to learn how to explain Gödel’s theorem to other people, but possibly for you to learn some of the concepts directly right now from me. For those of you who already do know what the theorem is, I hope you will find some new variations of looking at it and how to think about it. Gödel’s theorem depends on the idea of self-reference or level crossing, which you’ll see in this paper. Its major impact upon mathematics is to show that formal systems have a certain kind of incompleteness. It really involves the concept of how self-reference, or mixing of different levels, can be brought into mathematics.

†This paper is a lightly edited transcript of the talk presented at the Fourth International Congress on Mathematical Education which was held in August of 1980 in Berkeley.
Which Ball Is The Most Different?

The obviously different one is the small white one. Two of them are the same color and two are the same size, and the only one that is in the intersection of those two sets is the small white one.

To begin with I will give you a little puzzle. It is a puzzle which delights me because it was something that happened to me as a real-life event. I was sitting at the edge of a swimming pool, and in the pool there were three balls, a small black one, a small white one, and a large white one. I was looking at them, and I thought, "How strange—the obviously different one is the small white one! Two of them are the same size and two are the same color, and the only one that is in the intersection of those two sets is the small white one. It is the most different because it is the most same!" Here we have a sort of mixing of levels. At some primitive level of observation, at the perceptual level, we have the categories of size and color, and there is no unique answer yielded at that level. But we have this more abstract view of the situation that says that size and color are two equally primitive things, so we can't choose between them. However, we do have an object that is different. We have one that differs in size, another that differs in color, and so there is one left over. So we are somehow changing in our level of description. We are moving to a more abstract level, and yet somehow we feel we really haven't cheated. I don't really know quite how to say this, but somehow there is a validity to this answer. Although it seems strange, there is some sense to this response and it is perfectly fair to move to this next abstract level and give this as our answer. This is the first example of how level mixing can appear in a simple context and can seem quite natural and amusing at the same time. Gödel's theorem deals primarily with this idea of self-reference, which is something that has fascinated people from all times and all places. Self-reference is something that occurs in many jokes: it is general and fascinates people.
What is the only place in a room that a laser cannot hit if you shine the beam? That place is indicated by the black dot in the figure on the left. The only place that laser cannot hit is its own rear end. You can shine it to any other point except that black dot.

You can actually get it to shine on that spot, depending on what you mean by “that spot.” If that spot is defined in coordinates *relative to the room*, then, of course, the laser can shine through it. If, though, it is defined *in terms of self-reference*, in other words, in terms of the laser itself, then, of course, it can’t ever shine on that particular point.

Now I would like to give you some examples of self-reference outside Gödel’s theorem. Some of them will be very simple, and some of them may be very familiar to you: they are intended to give you the flavor of it. My first example involves a question about a laser. What is the only place in a room that a laser cannot hit if you shine the beam? I have indicated it by the black dot in the figure. The only place that laser cannot hit is its own rear end. You can shine it to any other point in the room except that black dot. However, this matter is just a little bit trickier than that. You can actually get it to shine on that spot, depending on what you mean by “that spot.” If I define it in coordinates *relative to the room*, then, of course, the laser can shine through it. If, though, I define it *in terms of self-reference*, in other words, with respect to the laser itself, then, of course, it can’t ever shine on that particular point. This is reminiscent of an epigrammatic statement that was made about 200 years ago by the German man of letters Georg Lichtenberg. He said that the one place in a room that a fly can land with safety is on the handle of a flyswatter.

Georg Lichtenberg’s invulnerable fly. These days, perhaps the invulnerable fly sits on top of a malathion-spraying helicopter!
Here are three classic examples of self-reference that often appear in little signs that you can buy in stores.

1. **THIMK**
2. **I NEVER MAKE MISTEAKS**
3. **PLAN Ahead**

The first one is the most concise version of Gödel’s theorem that I have ever seen. The second one does in fact contain a mistake, since “mistake” is misspelled. In the third one, it’s implied that the signmaker did not plan ahead (but on second thought, of course, that’s false). These are all variants of self-reference, but an interesting version of self-reference, in that the self-reference is *indirect*. In each case, it is left to the observer to perceive that what the sign is saying is related to the sign itself. It doesn’t say anything such as “This sign contains a mistake.” It doesn’t refer to itself *directly*. Yet it does have an indirect self-reference.

Here are some *directly* self-referential statements:

**Hofstadter’s Law**

It always takes longer than you expect, even when you take into account Hofstadter’s Law.

The next one is very similar to Hofstadter’s Law, and is, unfortunately, a sad fact:

**One of the lessons of history is that no one ever learns the lessons of history.**

Finally, we come to what is often regarded as the central self-referential statement, namely, the Epimenides Paradox:

**Epimenides Paradox**

This sentence is false.

Probably all of you have thought about the Epimenides Paradox, and realize that if it is true, then it is false. And if it is false, then it is true, and that gets you into a bind. Gödel’s theorem is really based upon the Epimenides Paradox. I would now like to tell you what Gödel actually did, and then further illustrate Gödel’s theorem.
In order to describe what Gödel did, I have to set the stage a bit and describe the state of the foundations of mathematics at the end of the last century. Mathematics was being axiomatized in an attempt to make very clear what did, and what did not, constitute a proof of a statement. Various axiomatic systems had been developed for Euclidean geometry, non-Euclidean geometry, projective geometry, etc. At that time, mathematicians such as Peano, Hilbert, and Frege were involved in axiomatizing mathematics. Perhaps the most ambitious attempts were made by Russell and Whitehead near the beginning of this century in their Principia Mathematica, in which they attempted to develop all of mathematics from the notions of logic and sets. The important thing about an axiomatization of any mathematical system is that it takes the set of concepts that one uses for thinking about these things and reduces them to a fixed and finite vocabulary, a finite set of symbols, and a finite set of axioms; or if you want, a finite set of axiom schemas (where schemas are like a mold for axioms) and a finite set of rules of inference. Thus, everything is collapsed down to a formal system, which involves expressing things in a fixed vocabulary according to a fixed grammar and then evolving theorems from axioms according to fixed rules. In other words, the result is a sort of typographical way of reasoning: reasoning is turned into a mechanical procedure—very similar to what computers do nowadays in manipulating symbols within themselves. This is what Principia Mathematica was all about: it tried to evolve all of mathematics in one system.

Then, Gödel, in 1930–31, twenty or so years after the publication of Principia Mathematica (1910–1913), realized that there was something going on here that could lead to some profound consequences. His idea was that the axiomatization of any branch of mathematics creates a very interesting formal object or formal structure, namely, the formal system itself. In other words, Principia Mathematica is not just a system in which mathematics is being done, but moreover it itself can become a mathematical object, in the sense that one can look upon its axioms, theorems, rules of inference and so forth as mathematical objects. The rules of inference are really things that are manipulating objects, the objects being strings of symbols. So this observation could have led Gödel to say, “Maybe I should invent a mathematics that applies to strings of symbols.” He would have then been the inventor of something resembling a modern programming language, like Lisp, Snobol, and other languages that are called “string manipulation languages.” But he didn’t do that. He thought to himself, “There is no branch of mathematics that studies the properties of strings of symbols. However, there is a branch called number theory which studies the properties of integers. I can just replace all of these symbols by integers, and that way I can turn the study of Principia Mathematica as a mathematical object into a branch of number theory. I can just replace all the strings in Principia Mathematica by numbers, and then I can describe what happens on each page as a sequence of transformations of numbers.” So it becomes a little branch of number theory, which is an irony because Principia Mathematica was supposed to be a system in which all of mathematics was developed. But here Gödel is in a way turning around and saying, “But Principia Mathematica itself, its own structure, just forms part of number theory. If you think of the book and the system of theorems, the system in which the symbols are being manipulated, as a mathematical object, then it itself has been sort of “swallowed” by number theory, which is one of the subjects that it is supposed to be studying.” That was the really tricky insight that Gödel had, the idea that one could turn a mathematical system
on itself so that it could become its own object of study—slightly indirect because it involves replacing or coding symbols by numbers that stand for them. That’s called Gödel numbering. That whole system had to be worked out in great detail to convince mathematicians that what he was doing was quite rigorous and not some made-up sleight-of-hand. Once you realize that Principia Mathematica (which includes number theory as one of its subjects) can talk about Principia Mathematica itself through this code called Gödel numbering, then you can get sentences that have two levels of interpretation rather than one. Earlier, a given sentence was only thought of as speaking about numbers. (It could say something like “641 is prime.”) But now, there emerges a second level of interpretation, because numbers represent statements. Someone could say, “Really, this statement says something about strings in Principia Mathematica”!

The final trick is to find a specific string that can say something about itself. What Gödel found was that it was possible to find a sentence that said this: “This sentence is not provable.” And this is where we come back to the Epimenides Paradox. The Epimenides Paradox states, “This sentence is false.” In mathematics up until that time the idea of truth was exactly equated with provability. In particular, if one took the system of Principia Mathematica (which was supposed to be all-inclusive), the idea of provability within Principia Mathematica would have been synonymous with truth. So, to say “This sentence is not provable” would have been synonymous with saying “This sentence is not true,” or, “This sentence is false.” But if that were really what Gödel’s sentence said, then Principia Mathematica would have a statement in it that was neither true nor false. This would be a statement about numbers, neither true nor false, and that seems impossible. It seems that if it’s true, then it’s false, and if it’s false, then it’s true. That just seems contradictory, and people had to think very hard to figure it out. (But Gödel perhaps didn’t have to think too hard.)

Even after Gödel had made it very clear in his paper what the consequences of this were, a lot of people debated for many years about the differences between truth and provability. What Gödel showed was that there was actually a distinction between provability within any specific system and truth. Consider Gödel’s sentence, for example:

<table>
<thead>
<tr>
<th>Gödel’s Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>“This sentence is not provable.”</td>
</tr>
<tr>
<td>(More precisely, “This sentence is not provable in formal system X.”)</td>
</tr>
</tbody>
</table>

There are two possibilities. Either it is provable, or it is not provable. Those are the two possibilities. If it is provable, then it is true, so what it says has to be true, and it says it’s not provable. That is really a contradiction, an absolute contradiction. If things were that way, then mathematics would be inconsistent, and that would be pretty much the end of mathematics. Thus that leaves us with only one alternative, namely, that the sentence is not provable (and that’s what it says about itself). So it must be true. Now, here is the distinction—here is the trick. It could be true and yet not be provable, which would mean the system was not quite as strong as one had thought. That is, Principia Mathematica actually had a defect—it was not
capable of proving all true statements. One really has to specify what one means by “provable.” You can’t just say “provable” in some ethereal sense—you have to specify the system you mean.

In passing, I should also mention that Leon Henkin invented a different kind of sentence:

Henkin’s Sentence
“This sentence is provable.”

Henkin’s sentence, by the way, is not the negation of Gödel’s sentence. Do you see why?

I now want to give you an image to hold in your mind of a formal system in which one can express and prove mathematical statements. As in my book, I call the system Typographical Number Theory (TNT). That’s appropriate because it sort of “blows up.” I’m not going to present it by any means in full because it would take too much space, and I just want to give you the flavor of it. Here are five axioms of arithmetic written in its notation just so you have a sense of the way in which one can express statements in a formal system.

**TYPOGRAPHICAL NUMBER THEORY (TNT):**
A formal system in which one can express and prove mathematical statements

**Five Axioms**

1. \( \forall a: \sim S a = 0 \)
2. \( \forall a: (a + 0) = a \)
3. \( \forall a: \forall a': (a + S a') = S(a + a') \)
4. \( \forall a: (a \cdot 0) = 0 \)
5. \( \forall a: \forall a': (a \cdot S a') = ((a \cdot a') + a) \)

and 18 rules of manipulation.

The upside-down A, “\( \forall \)” is a quantifier and means “for all,” the colon is just a punctuation mark, and the little swirl, “\( \sim \),” called a tilde, means “not,” and finally the capital “\( S \)” stands for the idea of “the successor of,” or “one more than.” So the first axiom says: “For all \( a \), it is not the case that the successor of \( a \) equals 0,” and in more understandable English, it says: “No number’s successor is 0.” Well, you could say “Minus one has a successor, and it is 0.” But what this axiom is doing is telling you what the domain is, and it is saying, “Minus one is not in the domain. There is no number whose successor is 0.” So it is basically saying “0 is the lowest number.”

The second one is easier. It says: “For all \( a \), \( a \) plus 0 equals \( a \.” The next one says something about the way addition works. It says: “If you have any two numbers \( a \)
and \(a',\) then \(a\) plus the successor of \(a'\) is equal to the successor of \(a\) plus \(a'.\)" The next one says: "Multiplying by 0 gives you 0 always." Then the next one is a primitive form of the distributive law. It says: "If you multiply \(a\) with the successor of \(a',\) then you get \(a\) times \(a'\) plus \(a.\)" These axioms are to be considered as strings, as inviolate objects that are simply capable of being manipulated. You can start manipulating according to the 18 rules of manipulation (which I am not going to exhibit). They simply involve moving symbols to the right or to the left or taking symbols out and putting others in, and so forth. They are very formal, somewhat like a computer program acting on objects. Presumably those 18 rules of manipulation are well thought out, so that they never take you to a false statement if you started with true ones. That's the idea of Typographical Number Theory.

<table>
<thead>
<tr>
<th>THE GÖDEL ISOMORPHISM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 . . . . 666</td>
</tr>
<tr>
<td>S . . . . 123</td>
</tr>
<tr>
<td>= . . . . 111</td>
</tr>
<tr>
<td>+ . . . . 112</td>
</tr>
<tr>
<td>. . . . 236</td>
</tr>
<tr>
<td>( . . . . 362</td>
</tr>
<tr>
<td>) . . . . 323</td>
</tr>
<tr>
<td>&lt; . . . . 212</td>
</tr>
<tr>
<td>&gt; . . . . 213</td>
</tr>
<tr>
<td>[ . . . . 312</td>
</tr>
<tr>
<td>] . . . . 313</td>
</tr>
<tr>
<td>a . . . . 262</td>
</tr>
<tr>
<td>' . . . . 163</td>
</tr>
<tr>
<td>∨ . . . . 161</td>
</tr>
<tr>
<td>∀ . . . . 616</td>
</tr>
<tr>
<td>⊃ . . . . 633</td>
</tr>
<tr>
<td>≃ . . . . 223</td>
</tr>
<tr>
<td>∃ . . . . 333</td>
</tr>
<tr>
<td>∀ . . . . 626</td>
</tr>
<tr>
<td>: . . . . 636</td>
</tr>
</tbody>
</table>

The idea that Gödel had was that because this is such a formal process, one can see this as a mathematical operation—the idea of manipulating symbols. Now he did not invent the idea of a string manipulation language. He simply said "Let's replace all these symbols by numbers, and then we'll have a number manipulation system." The way he chose to do it is not the way I am going to show you, but the way I am going to show you is just as good. It works just as well. One has to replace each symbol by some number. In the box, I show each symbol of TNT (and there are 20 of them) corresponding to a three-digit number so that one can "translate" a string of TNT symbols into a long numeral that consists of a sequence of three-digit numerals. (I'll soon show you how.) Once you grasp this idea of replacing symbols by numbers, long strings of symbols by bigger numbers and so forth, then you have the idea that the system can speak about itself in code. Once you understand that, you can go on to hypothesize that maybe some string could be invented that speaks about itself, and says about itself that it is not provable. I am going to come back to
that question of how a statement can be constructed that speaks about itself at the end of the talk. Let's take that for granted for the moment. If you will assume that one can construct such a statement in the language of *Principia Mathematica* or in the language of TNT, or in any formal system of this sort, then you might ask this question, which is a very important question: Is Gödel's sentence really a statement about numbers, or a statement about itself? Remember it is a string in some formal system, and ostensibly that formal system contains statements about integers as did TNT.

For example, “*a times 0 is 0 for every a.*” That's a statement about integers. Or you could say something like “17 is prime,” in that language. Basically, that sounds like a statement about 17. So if you have a very long and complex statement that says something very complicated about numbers, how can it also say something about itself? Does it really say something about itself? It is a question of level of description and that's a very important idea.

Let me now take a minute to talk about computers. Everybody speaks of computers, at bottom, as manipulating “bits.” “What you have in computers are 1's and 0's,”—that's what people say. “You move these 1's and 0's around in fancy ways, and that's really all a computer can do.” That's sort of a strange way of speaking about it because we are also very happy to say that a computer manipulates letters. We talk about “text-handling programs” all the time. Is it wrong to say that computers manipulate texts? Do they really just manipulate 1's and 0's? Are they really just doing binary arithmetic? Of course, if you say they are just doing binary arithmetic, why should you stop there? You can say, “There are no 1's and 0's in a computer. All that's there is current going off and on. And so really all that a computer can do is manipulate electrons. It can't manipulate 1's and 0's. It can't do any arithmetic at all! It can just manipulate electrons!” There is a point at which you want to stop going down in the hierarchy of description. You might want to say that a computer can manipulate 1's and 0's. Or if it is convenient, you might want to stop above that level. You might want to say, “A computer can really manipulate letters.”

You might also want to stop for a minute and change gears to consider another image. You might ask: “What is a novel? Is a novel really a set of letters? Or is it a set of words? Or is it a set of sentences? Or is it a set of ideas? Or is it a set of events? What is a novel?” It would certainly be wrong to say a novel is a set of letters or words, because we say the novel could be translated into another language. Ultimately, when we remember a novel, we don't remember any of the letters, we don't remember any of the words (or only a few of them). We remember the events. But then—doesn't style count? It is very hard to determine the right answer to the question “What is a novel?” In some sense, all of the different answers are right. I am trying to make certain you understand my point of view. I'm not saying anything is wrong with saying Gödel's sentence is a statement about numbers, but by the same token one shouldn't feel bad about saying it is a statement about itself. It's both of those things! It is both of these things simultaneously. Neither view of it is more right or more wrong than the other.

To give you a humorous version of Gödel's theorem, consider the “Wild Dance” version of Gödel's theorem. This is a real-life example which happened at a party. At the party, certain records could not be played. Actually, most of them couldn't be played, because people would dance to them. As soon as they danced, the floor would start to vibrate, the record player would shake, the needle would bounce all
Gödel's Theorem (Wild Dance Version)
“This record cannot be played at this party”

→ Grooves → Needle → Electricity → Loudspeaker → Air → Eardrums
← Phonograph ← Table ← Floor ← Muscles ← Nerves ← Brain ←

Playing a record at a party can create a “strange loop”—a sort of generalized, level-crossing feedback—which unexpectedly prevents the record from being played. This is a real-life analogue to Gödel’s construction of a self-referential (and self-undermining sentence) in a formal mathematical system. Both involve the fact that symbolic activity (musical vibrations or proving of theorems) has simultaneously several levels of interpretation, and these levels of interpretation, far from being just “different ways of looking at one thing,” can actually interfere with each other and cause near-paradox (Gödel) or wreak havoc (at the party).

over the place, so it wouldn’t be reproducing the music any more. It was a self-destroying dance! This gives us a question to ask. We have a set of “levels of translation” of the grooves. The question is: “Is the playing of a record really: a set of vibrations in a loudspeaker, or waves in the air, or shakings of the floor, or motions of people’s feet, or quiverings of the table that the record player is on?” It’s all of these things simultaneously. Whenever one of these things happens, the next one is set in motion. If you have dancers there, people will start dancing. When they start dancing, the floor vibrates. When the floor vibrates, the table vibrates. When the table vibrates, the phonograph will vibrate. When the phonograph vibrates, the record will vibrate. So the loop will close. All of these are valid descriptions of what a record, when played, is doing. This particular example of the record is a very interesting one because one can turn it into a kind of paraphrase of Gödel’s theorem by saying: “For every record player, there are records that it can’t play.” One doesn’t even require the dancers. After all, the sounds all by themselves will vibrate the record player itself. So one can say there is automatically some sort of self-destroying record. For every record player, there is a record that will make it vibrate and fall apart. After all, every record player has a special resonant frequency at which it will start vibrating more and more, and eventually it will break. You might say, “Oh, no, it doesn’t. It might just be a very well made one.”
But certainly if you make a loud enough noise, anything will break. If you reproduce the sound of an atomic bomb in perfect fidelity, it will destroy the record player. Given any record player, there is some bomb that will destroy it. You simply require that the record have the noise of that bomb in its grooves and then the record player will not be able to play that record in full fidelity. This is another image of how to think of Gōdel's proof.

I have one more image of Gōdel's proof that I would like to give you. The idea here involves likening symbol manipulation to the moving of railroad cars in a shunting yard. We have an engineer in a locomotive, and this engineer has been told to follow certain instructions in the shunting yard. He is blindly following these instructions. The instructions involve the numbers on the sides of the boxcars. He doesn't know what is inside the boxcars. All he knows is that he must follow certain rules, and he can move cars around in one way or another in the yard, and he tries to create certain trains according to the rules. For example, there is one rule that says you can detach any three cars that have the pattern, \[626, \_, \_, 636,\] where the middle one can be anything (see figure). That's one rule that he can obey. He can then detach them and they go away. He is left with the set of cars as shown in the figure.

Another rule that he can obey is this one: "Replace 262 by 666." Then he can replace that one as shown. The engineer is an intelligent guy, and as he is shunting these cars around, he realizes that, in fact, he can think of just the numbers on the sides. He doesn't have to actually do the physical manipulation of the cars.
(c) Another rule says that a pattern such as \[ \begin{array}{cccc}
123 & 123 & \ldots & 123 & 666
\end{array} \] can be substituted for \[ 262. \] So in the five-car train, the engineer performs the simplest of all such substitutions: namely, just \[ 666 \] for \[ 262. \] Thus we have mathematically converted one very large number, in two steps, into a somewhat smaller, but still huge, number. Or is it that we rearranged some freight, in a railroad shunting yard, into a different order? And what kind of freight might it be?

He can just think of these numbers on the side of the train and do it all in his head! Rather than talk about "assemblable trains" (because some trains are assemblable according to these rules, and others are not), he can just as well talk about "producible numbers." What I mean by a "producible number" is the long number that stands for any whole assemblable train. In this case (see figure), the number would be \( 223,123,666,111,666. \) It would be a producible number because he started with the train in one position that was assemblable and did certain things according to the rules, and he came up with another train (i.e., number). Thus he could start thinking about the numbers that he was producing—these long, gigantic numbers—and ask if they had some number-theoretical property that distinguishes them from the non-producible numbers. Of course, the answer would have to be "yes", because if you think about it you will see all that he is doing are formal operations. But any formal operation on these numbers is really a mathematical operation. It is an operation that belongs within the discipline or the framework of number theory. It is a number-theoretical operation. The set of numbers attainable through these manipulations constitutes a set of numbers definable in number theory. Thus, he can become a number theorist instead of an engineer, and decide which trains are assemblable and which ones are not. This is something he finds quite fascinating. So he starts investigating the properties of these huge producible numbers, and, flipping the coin, of the non-producible numbers.

(d) View from a helicopter of the train in (a). Here is our answer as to what kind of freight is involved. Each car is carrying a symbol of the formal system TNT! Shunting cars in the railroad yard is simultaneously an act of TNT symbol-manipulation (seen from above), and numerical computation (seen from the side). Theorem-proving in a formal system, as seen by Gödel (whose numbers adorn the sides of the cars), is equivalent to number-theoretical calculation according to certain computational rules.
Quite coincidentally, he also happens to have studied TNT (Typographical Number Theory), and he is very interested in that, too, but that is just another idea to him. One day he happens to be taking a helicopter ride, and he gets a top view of what his trains look like. He gets to see what's inside the boxcars (see figure). This one in particular has one of the five axioms of TNT in it: “For all $a$, it is not the case that the successor of $a$ equals 0.” Now remember that one of the things he was allowed to do was remove these three cars so that it left him with a shorter string, in which he then replaced the 262 by a 666. Well, the 666 cars all have 0's inside (see figure). In fact, the correspondence between a car's side number and the symbol it is carrying is exactly the one that I exhibited earlier—the one that had all the symbols of TNT with their three-digit codes, so that all the 666 cars carry 0's inside. So, in fact, what he had unknowingly started out with was an axiom-carrying train; and by manipulating a little bit, he came up with this new theorem-carrying train, which seen from above says, “It is not the case that 1 equals 0”—a true sentence that follows from the other one because the other one says: “No number's successor is 0.” This one says: “It is not the case that the successor of 0 is 0,” and I just call the successor of 0 "one". This new train says “It is not the case that 1 equals 0.” This is a revelation because now he sees that although he had thought that all he was doing was constructing producible numbers, it turns out that from another (literally higher-level) viewpoint, he was actually manipulating strings of TNT. In fact, he was producing statements of number theory when seen from above! From the side, he was just producing numbers; but from above he was producing statements of number theory (and they were all true). Thus, he could conceive of the idea that some train, some very long train, might be talking about its own number, its own code number, and it might be saying something about the producibility of that number. Remember that producibility (like, say, primality) is a number-theoretical notion, and if our formal system is at all powerful, then it can certainly reason about this notion of producible numbers. Thus, some very long train might be talking about whether or not it itself is producible and might even say, “This train is not producible; this train is not assemblable.” Now that is a very strange idea. I would like to suggest to you how that could happen.

(e) Helicopter view of the train in (c). This train expresses a theorem of TNT that is a direct consequence of the axiom seen in (d). The theorem says: “One is not zero.” The engineer created this theorem not by thinking about how to deduce logical consequences of axioms, but by following rules governing numbers on the sides of the cars. Yet from an outsider's point of view, they are equivalent. No one could tell which way the engineer was thinking about his activity! This equivalence between doing reasoning about mathematics and doing mere calculation within mathematics yields a startling level-crossing loop, allowing a mathematical system to talk about itself. It's the gist of Gödel's insight that self-referential, paradoxical statements can be translated into formal mathematical systems, by means of codes where numbers stand for symbols.
This is very hard to figure out; it is the last trick that Gödel used. “How do you get a string of some formal system not just to speak about other strings in the formal system, but to speak exactly about itself? If you try naively to make something that speaks about itself, you might try to quote a sentence within itself. (I am speaking now of English.) If I try to make a sentence that talks about itself, there are two ways I might use. One is to say something like, “This sentence is false.” Now how does that work? It is based on the convention that the phrase “this sentence” refers to the sentence it is contained in. Or I could say something like, “I am lying.” That is using the convention that “I” stands for the speaker. We are using in a certain sense a sleight of hand. I don’t know what you want to call it, but it is a kind of convention. There is another way of achieving self-reference, and that is more like the one that said “THIMK,” where the self-reference is more indirect, where the thing that is expressing something has the same form as the thing it is talking about.

If you want to have a sentence that talks about its own form without referring to itself directly, you have to have some sort of way of quoting it within itself, and you come to something like this example (see figure). We have here an attempt to create a sentence that speaks about itself, and it works. It talks about a quoted sentence which is infinitely long. So the sentence itself has to be infinitely long, and lo and behold! The quoted sentence is the sentence itself! It is a sequence of nested sentences, each of which is quoted within the outer one, and since it is an infinite structure we have the outermost one identical to the one that is one level in, and it’s identical to the sentence one level further in, and so forth. But this is not going to satisfy us if we are trying to produce a finite sentence.

So how do you do it in a finite sentence? It might seem impossible. Gödel found a way to do it, and basically it involves the idea of diagonalization as in Cantor’s argument that shows the real numbers are uncountable.

\[
\begin{align*}
\text{The sentence} & \\
\text{"The sentence} & \\
\text{"The sentence} & \\
\text{"} & \\
\text{"} & \\
\text{is infinitely long."} & \\
\text{is infinitely long."} & \\
\text{is infinitely long.} &
\end{align*}
\]

I am going to show you a way of doing it in English. Instead of showing you via Cantor, the mathematician, I’ll show you through Quine, the philosopher. Quine invented the following method. The idea is of self-reference via form not via convention such as “I” or “this sentence.” “Quining” is what I called it in my book. (He certainly didn’t call it that!) Quining is an operation that I define on any string of English. Take the string, put it in quotes, then take another copy and put it right after the quoted one.
So you have two copies of a string, and the first one is in quotes—that’s all. Here is an example of a quined phrase:


“is a sentence with no subject”
is a sentence with no subject.

It is a perfectly true sentence. Neither half is a sentence, but the full thing is a sentence. (And it is a sentence with a subject, incidentally!) The next one, though, is a little trickier and is one where you get real self-reference.


“yields falsehood when preceded by its quotation”
yields falsehood when preceded by its quotation.

What is this really saying? The second part of the sentence is active: it’s the thing that is speaking. It is referring to some phrase in quotes. What is it saying? It is saying that something, when you precede it by its quotation, yields a false statement. Well, what is that thing? That thing is the very thing that is speaking. And when you precede it by its quotation, you more or less “accidentally” have reconstructed the sentence itself. So, in fact, this sentence is talking about itself by means of form, not by convention.

Gödel figured out a way of doing that in mathematical systems. The way is very simple, because quoting something is very much like taking its Gödel number. Converting a string into a quoted string is very much like converting a string to its Gödel number. Gödel realized that one could parallel this construction precisely within a formal system and create self-reference this way. Actually, of course, it happened the other way around. Gödel didn’t see this sentence by Quine. Quine invented it after Gödel’s work. It was just meant to explicate what is going on in Gödel’s proof.

My last picture shows a parallel phenomenon that happens inside life forms. The way self-reference happens in mathematics or formal mathematical systems is very similar to the way self-replication happens in living organisms. How does a fish reproduce itself along with its DNA inside? To give you a glimpse of what I am going to say, the fish is sort of like the unquoted string, and its DNA is like the quoted string. So first take the DNA out by itself (see figure). Now let the DNA develop according to certain chemical processes. (This is the analogue to the typographical process of quining.) The DNA develops, and we see it start to develop here, develop further, further, and you see what results? I will just say this is a sort of an answer to an age-old question. It is a very sexist question. It assumes that female ova play no role in reproduction of the species at all, that they are just there for food, and that really the sperms contain all the hereditary material. Just make this simplifying assumption: that a man contains sperms, and since his sperms are going to turn into people, they too must contain subsperms, and they must contain subssubsperms, etc., etc., ad infinitum. The medieval puzzler is: “Does a man contain all of his future progeny for all generations in some sort of infinite regress?” The answer is, no, not really, not even if reproduction went that way—no more than a fish’s DNA is actually a fish.
DNA not only codes for the structure of the fish, but also programs its own replication. The method by which this is accomplished is analogous to the way in which self-reference can be achieved in a formal mathematical system.


He was born in New York City in 1945, but spent most of his childhood years at Stanford, California. He was always interested in mathematics and physics as a child. At age 12, he began learning French and became fascinated by languages. In college he studied several other languages, as well as computer science. As an undergraduate, he used computers to experiment with several ideas about language structure, including forays into artificial intelligence research. He also used the computer to do what he calls “experimental math,” particularly number theory “experiments.” The result was a general fascination with recursion and logic.

He began his graduate studies at the University of California, Berkeley, in mathematics, but later switched to physics at the University of Oregon.

Music has been equally dominant in his life and includes piano playing and composition.
His many interests—logic, computers, languages, music, and the mind—
recombined in 1972 and set off what he describes as a "mental fire" that burned for
several years. The outcome was "Gödel, Escher, Bach," and, after finishing his Ph.D.
in physics, a return to his old interest: artificial intelligence.

He recently talked about his work. "My current research in artificial intelligence
involves attempts to get a machine to see patterns in simple structures and to
generalize those patterns in "natural" or "artistic" ways. You could call this an
attempt to program "artificial intuition." My passion is to gain insight into the intuitive
processes of thought."

---

Telegraphic Reviews

Edited by
Peter A. Lindstrom

Telegraphic Reviews are designed to give the TYCMJ readers short summaries of the contents of
recently published mathematics texts, supplementary materials, mathematics laboratory materials, and
professional readings. Such information can be helpful in deciding whether or not to order an
examination copy and whether or not the text, supplementary material, or professional reading material
might be purchased for a mathematics laboratory, college library, or personal library. Publishers are
denoted by standard abbreviations; complete addresses may be found in Books in Print.

Codes Used in Telegraphic Reviews

- T is Textbook
- S is Supplementary Material
- L is Library Purchase
- ML is Mathematics Lab Material
- P is Professional Reading
- * is Special Positive Emphasis
- ? is Special Negative Emphasis
- 1, 2, 3, 4 is Number of Semesters
- for Use of Textbook

APPLICATIONS (S**, 1): Tools for Teaching, Compiled by the UMAP Central Staff.

This book contains twenty-four UMAP (Undergraduate Mathematics and Its Applications
Project) modules. Each module is a self-contained, lesson-length, instructional unit from
which undergraduate students can learn professional applications of mathematics and
statistics to such fields as biomedical sciences, economics, American politics, earth science,
and the social sciences. This is one book that will NOT collect dust on your bookshelf as you
will use it often. PAL

ALGEBRA (COLLEGE) (T+, 1): College Algebra, by Jimmie Gilbert, James Spencer and

Includes real numbers, polynomials, exponents and radicals, linear and quadratic equa-
tions and functions, conic sections, systems of equations and inequalities, determinants and
matrices, exponents and logs. Informal approach with effective use of discussion and
illustration. Numerous and well-explained examples. Practice tests follow each chapter.
Definitions and key points are emphasized and carefully stated. LMC

[Continued on p. 163]