DALE JACQUETTE

METAMATHEMATICAL CRITERIA FOR MINDS AND MACHINES

1.

The Turing test of machine intelligence highlights a dilemma in the philosophy of mind. The imitation game described by A. M. Turing appears easy for a logically possible finite state machine to play and win. The machine, hidden from but in peripheral communication with an interrogator, need only reply to questions addressed to it in such a way that the interrogator is unable to determine whether the answers are given by a person or machine.[1]

If the machine's sophisticated simulation of human verbal behavior is sufficient proof that the machine is intelligent, then some possible machines are persons or minds, and the mind itself is mechanical. But if the simulation is not sufficient, then there is no adequate philosophical justification for rejecting solipsism. The only evidence that there are other minds is the external behavior and especially verbal behavior of beings relevantly like ourselves. If the indistinguishably precise mechanical simulation of intelligent verbal behavior does not prove that machines have minds, then the behavior itself cannot be used to prove that any other human has a mind. In either case, the conclusion contradicts the intuitively plausible, commonly held beliefs that we are justified in accepting the existence of other minds, and that machines, however complex, cannot have the private or internal psychological experiences that uniquely characterize the mind.

This dilemma can be avoided by applying some of the formal undecidability consequences in the metatheory of standard first-order logic. The mechanical simulation of mind can always be detected in Turing's imitation game if the interrogator asks about the truth-value of appropriately formulated Gödel sentences. This provides a metamathematical criterion for the distinction between minds and machines, because finite state machines are inherently unable consistently to simulate the mind's judgment that Gödel sentences are true and their negations false. There is in principle therefore some respect in which machines can only imperfectly imitate the activity of

Erkenntnis **27** (1987) 1–16.

minds. The criterion makes it possible to avoid the dilemma by using the Turing test to distinguish between minds and machines on the basis of their distinct verbal behavioral performances in playing the imitation game.

2.

When J. R. Lucas first proposed a version of the Gödel criterion for the distinction between minds and machines, he was able to report that "almost every mathematical logician" with whom he discussed the problem agreed that Gödel's theorems imply that mechanism is false, that minds are not machines, and that no possible machine can adequately model the mind.[2]

But if the same informal poll were conducted among logicians today, the results would probably be very different. The tide of opinion has turned against the Gödel-Lucas refutation of mechanism, and few if any mathematically trained philosophers of mind would now support Lucas' metamathematical distinction between minds and machines. There have been many criticisms of Lucas' argument, including those of Judson Webb, Paul Benacerraf, Charles S. Chihara, Hilary Putnam, David Lewis, Anthony Hutton, Daniel C. Dennett, and David L. Boyer.[3] Lucas has responded capably to most of these attacks,[4] but doubt and skepticism remain. Nevertheless I am convinced that Lucas is correct in his underlying contention that Gödel's incompleteness theorems establish a real distinction between minds and machines.

I shall not try to evaluate in detail the objections which have been raised against Lucas' original treatment of the criterion. Instead I will offer a somewhat different formulation which I believe avoids these criticisms. Then I will sketch an alternative explanation of why the Gödel criterion works. Here I sharply disagree with Lucas, who maintains that the criterion distinguishes between minds and machines because of an analogy between the diagonalized self-reference of Gödel sentences and the reflective self-consciousness of thought.[5] The lack of an adequate account of why the Gödel criterion works has hitherto led many critics to suppose that the incompleteness theorems could not possibly provide the basis for a distinction between minds and machines. But when a correct explanation is given, the criterion becomes much more plausible.

3.

Gödel sentences are formally undecidable sentences of standard first-order predicate logic powerful enough to represent the axioms of Dedekind-Peano arithmetic. The sentences of the logic are arithmetically coded to avoid type-theoretical restrictions on paradoxical syntax combinations (like the application of a predicate to another predicate in ZZ, where $Z = \lambda x[\sim(xx)]$). An open sentence with a free variable is constructed, which says in effect that the object to be designated by a substitution for the free variable is not provable. The open sentence has a certain Gödel number in the arithmetical coding of the logic, which is then substituted for the free variable. This completes the diagonalization. The Gödel sentence which results from this substitution says of itself that it is not provable. The mapping of logical syntax into Gödel-numbered space circumvents type-theoretical restrictions, because the Gödel sentence that obtains does not involve the application of a predicate to another predicate, but to an object-designating term, a constant or numeral, which stands for the sentence's own Gödel number.

Every well-formed sentence of standard bivalent logic is either true or false. If the Gödel sentence is true, then there are true sentences of the logic which are unprovable, which means that the logic is formally incomplete. But if the Gödel sentence is false, then it is provable, since it says of itself that it is not provable. This implies that the Gödel sentence, though false, is forthcoming as a theorem. The logic must then be formally inconsistent, because false sentences cannot be validly deduced from a formally consistent logic if its axioms are true. The logic is therefore either inconsistent or incomplete.[6] In order to avoid inconsistency and the consequent triviality of deductive inference, it is generally admitted that first-order logic with arithmetic is formally incomplete. The Gödel sentence is judged true, and its negation false.

The Turing test can be used to distinguish between minds and machines if the interrogator asks about the truth-values of Gödel sentences and their negations. For this it is important to limit consideration to what may be called normal Turing test participants. The interesting problem posed by the Turing test is not whether the mind can deceptively mimic the programmed responses of a machine, but whether any machine can adequately simulate the intelligent responses

of a mind, and whether or not the simulation could be detected by a Turing test interrogator. In the uninteresting case, both minds and machines give identical canned replies to the interrogator's questions from such an impoverished vocabulary that no interrogator could possibly distinguish them. The criterion must therefore be limited to Turing tests in which normal participant minds and machines 'sincerely' try to convince the interrogator that they are minds rather than machines. The interrogator inquires of normal Turing test participants whether certain formulations of Gödel sentences and their negations are true or false.

There are several ways in which finite state machines can be programmed to simulate judgments about the truth-values of sentences. Machines can: (1) produce truth-value pronouncements at random by means of a random number generator; (2) make truth-value pronouncements in response to questions about particular sentences stored in their memories; (3) deduce the truth or falsehood of sentences from a data base of information in accord with the rules of deductive inference. But none of these possibilities enable a finite state machine adequately to simulate the mind's ability to judge of every Gödel sentence that it is true and its negation false.

(1) When asked about the truth-value of a Gödel sentence or its negation, a machine may print 'True" if a number randomly generated on the occasion comes up even, and 'False' if the number is odd (or vice versa). But if a machine's answers are given entirely at random, then it should not be difficult for an interrogator to discover that he is communicating with a machine rather than a mind. The interrogator would be justified in drawing this conclusion after receiving different answers to the same or similar questions, a result reasonably expected of a randomizer, but not of a normal participant mind.

(2) A finite state machine can be supplied with a list of sentences to be stored in its memory, together with instructions to print either 'True' or 'False' when asked about their truth-values. Gödel sentences might be added to such a list, along with the command to print 'True' when asked about their truth-value, and 'False' when asked about the truth-value of their negations. But a finite state machine has only finite memory, and since there are infinitely many natural numbers and combinations of natural numbers to use in the Gödel-coding of logical syntax, there are infinitely many different Gödel sentences. No matter how finitely large a list of Gödel sentences a finite state machine is

programmed to pronounce true, an interrogator in principle can stump the machine with a selection from an infinite inventory of other Gödel sentences not in its memory.

(3) Calculating devices can be described as 'unprepared' for particular conclusions they are asked to deduce if the conclusions or their immediate equivalents are not already stored in an accessible database from which the machines can simply retrieve them. Deductive proofs given by finite state machines are formally and philosophically interesting only if the machines are unprepared for their conclusions, but must derive them from other available information by means of deductively valid algorithms or inference procedures. Prepared machines that 'deduce' conclusions previously given to them are the mechanical embodiments of circular reasoning and question-begging argument. Although finite state machines can validly deduce the truth-values of many kinds of sentences, unprepared logically consistent finite state machines cannot deduce the truth-value of any Gödel sentence. Gödel sentences say of themselves that they are unprovable. If a Gödel sentence is true, it is thereby unprovable; so no unprepared machine (and indeed no mind) can deductively prove that Gödel sentences are true.[7]

4.

There may conceivably be yet another method by which a finite state machine might simulate the mind's ability to recognize any formulation of a Gödel sentence, distinguish it from its negation, and then pronounce it true and its negation false. The mind undoubtedly has procedures of this kind. But to suppose that any effective procedure must be programmable is automatically to suppose that any effective procedure is necessarily mechanical, which is viciously circular in the defense of mechanism against the Gödel sentence Turing test criterion.

An unprogrammable nonmechanical procedure which the mind may use to judge the truth-value of Gödel sentences can be characterized as an intensional conditional in the imperative mood.

(P) If *S* says that *S* is unprovable, then answer (print): '*S* is true.'

Procedure (P) goes into effect when the mind determines that an

interrogator's quiz sentence says of itself that it is unprovable. The mind is able nonmechanically to implement (P) because it understands the meaning of Gödel sentences and their negations, and can use this semantic information to decide when a sentence S says of itself that it is unprovable. The interrogator need only explain that $S = \sim Thm(n)$, and that Gödel number $g(\ulcorner \sim Thm(n) \urcorner) = n$.[8] This is sufficient for S to be formally undecidable for both minds and machines, and for the unprepared finite state machine to be unable satisfactorily to simulate the mind's judgment that S is true and $\sim S$ false.

Machines cannot be programmed to mechanically implement procedure (P) because there is no suitable purely extensional or syntactical translation of the required intensional procedural condition that 'S says that S is unprovable.' It is inadequate to inform the machine that if $S = \sim Thm(n)$ and $g(\ulcorner \sim Thm(n) \urcorner) = n$, then S says that S is unprovable. This makes the unprovability predication too specific. The problem is not that there are infinitely many Gödel arithmetizations of logical syntax, since the two conditions given by the interrogator jointly guarantee diagonalization in sentence S for any Gödel number n. The problem is that the interrogator might choose predicates other than '$\sim Thm$' to represent unprovability, which would then need to be invalidly substituted in at least some referentially opaque nonextensional contexts.

The appearance of the external negation sign in '$\sim Thm$' suggests that a machine might be programmed to distinguish Gödel sentences from their negations by first translating troublesome constructions into prenex form, and then checking for occurrences of an outermost negation sign. But this will not avail. Gödel sentences can be formulated without negation by means of a primitive unprovability predicate, such as '*Nthm*' or even '*Thm*'. If negation and nonnegation Gödel sentence formulations are tried out by the Turing test interrogator, then finite state machines with the prenex subroutine will inevitably confuse Gödel sentences of one formulation with negations of Gödel sentences in the other formulation.

Suppose an interrogator decides to use '$\sim Thm$' to represent the property of being provable, reserving '*Thm*' for the now more primitive metatheoretical property of being unprovable. In that case, $S(= \sim Thm(n)$, where $g(\ulcorner \sim Thm(n) \urcorner) = n)$ is not diagonalized, but harmlessly self-referential, saying of itself that it is provable or at least not unprovable, rather than unprovable. If this is false, if S is not prov-

able, then it should not be blindly pronounced true by the mechanical implementation of (P). This would conflate judgments about the truth-values of some Gödel sentences with judgments about the truth-values of the negations of others. The mechanical implementation of the mind's procedure (P) cannot adequately simulate the mind's judgment that Gödel sentences are true and their negations false, and so has no prospect of surviving the Gödel sentence Turing test. (It similarly will not do to have the machine answer 'I do not know' or 'The sentence is neither true nor false' for both Gödel sentences and their negations. This is no better than the retarded robots in bad science fiction films woodenly spouting 'That does not compute!' The mind in principle is not limited to these waffling noncommital judgments.)

If the Turing test is to have philosophical significance, minds and machines must be fairly pitted against one another. The mind must not have access to information denied its mechanical competitors. Otherwise the interrogator might as well require unprepared machines to answer questions in Etruscan or Serbo-Croatian. But no amount of shared information can facilitate the mechanical implementation of (P) in any plausible simulation of mind. To bootstrap the machine so that it can fairly compete with the mind it is necessary to determine what information and definitions of terms the machine needs, and in what form it can use them.

The machine table of an information processing mechanism is an applied realization of standard first-order logic, and as such must be expressible in a purely extensional idiom. If a machine is equipped with a functionally intensional instruction, then by definition it will generate its own syntactical inconsistency. Intensional contexts are not truth-preserving under intersubstitution of logically equivalent sentences or extensionally codesignative terms. A machine with an irreducible functionally intensional operation eventually transforms true into false propositions by invalid semantically unauthorized substitutions.

Let a machine have built into its operating instructions and lexical initializations a particular predicate principle for interpreting Gödel sentences, and a uniform substitution rule for terms of identical reference in purely extensional contexts.

(G) If $g(\ulcorner \sim Thm(n) \urcorner) = n$, then $\sim Thm(n)$ says that $\sim Thm(n)$ is $\sim Thm$.

(E) If $T_1 = T_2$ and $(\ldots T_1 \ldots)$, then $(\ldots T_2 \ldots)$.

The interrogator's input at first is that $S = \sim Thm(n)$, $g(\ulcorner \sim Thm(n) \urcorner) = n$, and $\sim Thm$ = unprovable. This makes possible the following ordered sequence of transformations in which the mind's procedure (P) is finally implemented.

(1)	$S = \sim Thm(n)$	
(2)	$g(\ulcorner \sim Thm(n) \urcorner) = n$	
(3)	$\sim Thm$ = unprovable	
(4)	$\sim Thm(n)$ says that $\sim Thm(n)$ is $\sim Thm$	(2, G)
(5)	S says that S is $\sim Thm$	(1, 4, E)
(6)	S says that S is unprovable	(3, 5, E)

The inference in (6) satisfies the procedural condition for (P). The procedure is implemented, and the machine prints the reply to the interrogator: 'S is true.'

This is the desired result. The transformations can be supplemented by a complementary procedure to simulate the mind's judgment that the negations of Gödel sentences are false.

(P*) If $\sim S$ is true by (P), then answer (print): 'S is false.'

When asked about the truth-value of the negation of a Gödel sentence, the machine first checks to see if it is a true Gödel sentence under procedure (P). If it is not, then the machine checks to see if its negation is a true Gödel sentence under (P). If it is, then the procedural condition of complementary procedure (P*) is satisfied, (P*) is implemented, the machine recognizes S as the negation of a Gödel sentence, and pronounces the sentence false.

The method seems general, and appears to enable the machine to score a perfect draw against any mind in the Turing test or imitation game, outwitting even the shrewdest, most relentless interrogator with flying colors.

Problems arise when the interrogator tries to redefine predicate '$\sim Thm$' as provability rather than unprovability. This can be done by informing the machine that $Thm = \sim Thm$, $\sim Thm$ = provable, or Thm = unprovable. These identities can be made to cancel previous definitions, such as $\sim Thm$ = unprovable in (3) of the transformations. But not all essential occurrences of the unprovability predicate can legitimately be replaced by its intended substituend, because some are

embedded in irreducibly nonextensional contexts. It is possible to retract the identification that $\sim Thm$ = unprovable, and to inform the machine that $S = Thm(n)$, $g(\ulcorner Thm(n)\urcorner) = n$, and even that $Thm = \sim Thm$. But this is not enough. To implement (P) under these re-identifications, principle (G) must also be revised. If this is not done, (G) cannot produce the required procedural condition that '$Thm(n)$ says that $Thm(n)$ is Thm'. But the revision of (G) cannot be licensed on the basis of any extensional substitution principle like (E), because (G) contains essential occurrences of the original unprovability predicate '$\sim Thm$' in irreducibly intensional contexts.

Consider the following alternative to (G):

(G′) For any (metatheoretical) property Φ, if $g(\ulcorner \Phi(n)\urcorner) = n$, then $\Phi(n)$ says that $\Phi(n)$ is Φ.

If the interrogator asks about the truth-value of sentence $S(= Thm(n)$, or $= \sim Thm(n))$, and informs the machine that $g(\ulcorner Thm(n)\urcorner) = n$ and Thm = unprovable (or $g(\ulcorner \sim Thm(n)\urcorner) = n$ and $\sim Thm$ = unprovable), then by instantiating 'Thm' ('$\sim Thm$') for Φ, the machine can execute the desired transformations, leading to the mechanical implementation of procedure (P).

(G′) need not be dismissed as higher-order if 'Φ' is interpreted as a metavariable admitting substitution instances, rather than as a bound variable for higher-order quantifications over metatheoretical properties. But the generality of (G′) is unjustifiably obtained only by permitting invalid substitutions or instantiations of unprovability predicates in the referentially opaque intensional Gödel number quotation context "$\ulcorner \Phi(n)\urcorner$".

Gödel arithmetization must provide a unique syntax-item-by-syntax-item numerical coding of every formal symbolization in its domain. Gödel number contexts are irreducibly intensional because $g(\ulcorner p\urcorner) \neq g(\ulcorner q\urcorner)$ where 'p' $\neq$ 'q', even if $p \equiv q$. The Gödel number of $\ulcorner p \supset p\urcorner$ is obviously not identical in the same Gödel numbering system to the Gödel number of $\ulcorner p \vee \sim p\urcorner$, despite the logical equivalence $(p \supset p) \equiv (p \vee \sim p)$.[9]

There is no adequate sufficiently general extensional method of providing the substitutions needed to mechanically trigger the mind's procedure (P). Principle (G) is too specific, and principle (G′) achieves generality only by requiring invalid intersubstitutions of extensionally codesignative unprovability predicates in referentially opaque Gödel

number quotation contexts. In a sense, the 'mechanical' implementation of procedure (P) by principle (G′) is not mechanical at all. But the mind's operations, as distinct from those of a finite state machine, are not subject to extensionality constraints. The convergence on propositional attitude contexts of intensionality in the linguistic mode and intentionality in the ontological mode suggests that the mind recognizes Gödel sentences and distinguishes them from their negations by a nonmechanical unprogrammable intensional operation. The intentionality of mind may explain why machines cannot mechanically implement the mind's procedures in its complex architecture of truth-value judgment strategies.

5.

Unlike machines, minds are able to correctly judge the truth-values of Gödel sentences. (Of course in practice not every mind has the opportunity or inclination to master the complexities of mathematical logic.) The mind's procedure is not mechanical, but intentional. It understands the meaning of Gödel sentences and their negations, and from this determines when a sentence does or does not say of itself that it is unprovable. The machine can only imperfectly simulate the mind's intentionality and understanding of a sentence's meaning. But there is no reason to suppose that every effective procedure is mechanical or programmable. If the interrogator in the Turing test or imitation game persistently asks about the truth-values of alternatively formulated Gödel sentences and their negations, then sooner or later he should always be able to unmask the imperfect simulation of mind by an imposter finite state machine.

The Turing test application of the Gödel criterion avoids an important criticism frequently raised against Lucas' argument. Gödel sentences cannot be deduced in logically consistent systems of standard first-order logic. But in inconsistent logics they are trivially deducible. If the mind is able to judge that Gödel sentences are true, it may be because the mind harbors hidden logical inconsistencies. An inconsistent mind in this respect would be indistinguishable from an inconsistent machine. The objection is therefore sometimes made that Gödel's theorems have no significance for the philosophy of mind and the distinction between minds and machines unless it can first be shown that the mind is logically consistent. But there is no straight-

forward demonstration of the mind's consistency, and Gödel's second theorem implies that if the mind is consistent, its consistency may be impossible to prove.[10]

In the Turing test these dialectical responsibilities are reversed. There the burden is on the machine to fool the interrogator, and not on the mind to prove its own consistency. If the mind is inconsistent, then an adequate mechanical model of the mind must also be inconsistent. An inconsistent finite state machine can deduce Gödel sentences and pronounce them true. But an inconsistent machine indiscriminately deduces every sentence, including Gödel sentence negations. The moral once again is that finite state machines need an extensional, mechanical, or referentially transparent syntactical method for distinguishing Gödel sentences from their negations. But no such method is available. The Turing test interrogator should easily be able to uncover the deception.

It might be thought that inhibitory filters could be programmed into the inconsistent machine to selectively screen out truth-value pronouncements, so that although every sentence is deduced, only Gödel sentences and not their negations are pronounced true. Filters of this imagined kind must be able to distinguish between Gödel sentences and their negations without substituting into intensional contexts. But previous discussion has already established that there is no effective decision procedure by which a finite state machine can satisfactorily distinguish between Gödel sentences and their negations. If there were such a filter, it would be unnecessary to make the desperate move toward the trivially valid deduction of Gödel sentences by logically inconsistent machines. The consistent machine could then simply print 'True' when asked about the truth-value of a Gödel sentence, and 'False' when asked about the truth-value of its negation. Even if the mind is inconsistent, and even if its inconsistency is somehow connected with its ability to judge that Gödel sentences are true, an inconsistent finite state machine in the Turing test or imitation game will be unable adequately to simulate the mind's judgment that Gödel sentences are true and their negations false.

A sentence S is valid or logically true in formal system L if and only if, for every truth-valuation or interpretation V in L, $V(S) = T(\text{true})$. Positive validity proofs are finite effective mechanical decision procedures whereby all valid sentences of a logic are determined to be valid. Negative validity proofs are finite effective mechanical decision

procedures whereby all sentences of a logic that are not valid are determined not to be valid. A system of logic is not formally decidable unless both positive and negative validity proofs exist for all its sentences. Standard first-order logic with arithmetic is formally undecidable because although there are positive validity proofs for all its valid sentences, there are no negative validity proofs for others that are invalid. In traditional metatheoretical terminology, the truths of standard first-order logic with identity, addition, and multiplication are said to be recursively enumerable (by closure on axioms, definitions, and inference rules), but not recursive. This asymmetry is perfectly mirrored in the failure of mechanical simulations of mind. The negations of some Gödel sentences and some harmlessly self-referential sentences that assert their own provability or deny their own unprovability are inevitably confused by machines, leaving a grey area of problematic sentences for which there are no effective mechanical negative validity proofs.

6.

The Gödel criterion distinguishes between minds and machines. But what makes it work? Why should the Gödel sentence, of all unlikely constructions, provide a basis for the distinction between minds and machines?

Lucas maintains that his version of the criterion marks a distinction between minds and machines because of an analogy between the diagonalized self-reference of the Gödel sentence and the reflective self-consciousness of thought. The fact that minds can and finite state machines cannot consistently judge that Gödel sentences are true and their negations false is supposed to indicate that machines are incapable of self-consciousness. Lucas writes:

> We can see how we might almost have expected Gödel's theorem to distinguish self-conscious beings from inanimate objects. The essence of the Gödelian formula is that it is self-referring .... When carried over to a machine ... [t]he machine is being asked a question about its own processes. We are asking it to be self-conscious, and say what things it can and cannot do. Such questions notoriously lead to paradox.[11]

Lucas' explanation has been criticized on a number of grounds.[12] Suffice it to say that there are indeterminately many harmlessly self-referential sentences (including diagnostic sentences about a machine's own processes) that pose no difficulty at all for an ap-

propriately programmed finite state machine, and for which the machine can triumphantly indistinguishably simulate the mind's responses. The issue here is not just self-reference, but diagonalization. Yet the criterion need not depend on any dubious analogy between self-reference and self-consciousness.

The Turing test version of the criterion which is proposed does not require that the Gödel numbering of diagonalized sentences used by the interrogator encode a machine's own logical operations. There is no mention of the machine's own particular Gödel sentences or Gödel numbers. The numbering of logical syntax in the construction of an undecidable Gödel sentence *S* might encode negation and the material conditional as the only primitive propositional connectives of a formal system. In the Gödel sentence Turing test, the mechanical simulation of mind by a finite state machine *M* might be foiled by sentence *S*. But machine *M* as described by its software or finite machine table might operate entirely by and-gates and no-gates, or by or-gates and no-gates, with no irreducible functional equivalent of the material conditional as a primitive propositional connective. In that case, the discrepancy entails that Gödel sentence *S* could not possibly be any of the uniquely corresponding Gödel sentences of machine *M*.

This makes the problem of self-consciousness irrelevant, since it is possible for a machine's attempt to simulate the mind to be thwarted by an undecidable Gödel sentence that is not the Gödel sentence of that particular machine. The approach also cuts short efforts to get around the criterion by invoking nonlinear extensions or time-sharing modes of Gödel-encoded machines that can monitor the operations of subordinate machines and pronounce judgment on the truth-values of their uniquely corresponding Gödel sentences.[13] The Gödel sentences that trip up the simulation of mind by a finite state machine presuppose that finite state machines are possible applied realizations of formal systems of standard first-order logic. But such Gödel sentences do not need uniquely to represent the logical operations of those particular machines. The correspondence is required in some formulations of the 'halting problem' for finite state machines,[14] with which the Gödel criterion is sometimes confused. But it is not essential in applying Gödel's first theorem in the Turing test to provide a metamathematical criterion for the distinction between minds and machines. The interrogator cannot challenge machines by confronting them with their uniquely corresponding Gödel sentences, because at

first the interrogator does not even know whether he is questioning a machine or nonmachine, let alone what particular machine is behind the curtain or what machine table it has. Without this information the machine's uniquely corresponding Gödel sentences cannot be constructed.

7.

I now want to propose a different explanation of the Gödel criterion. To understand what a sentence means, or, in Frege's terminology, to grasp its thought,[15] it is necessary to understand at least part of what it does not mean. For this it is minimally required to be able to distinguish the sentence from its negation. A finite state machine can simulate the mind's understanding or grasping of the sense of a sentence in ordinary cases because it can use deductive methods to distinguish extensionally, mechanically, or syntactically between sentences and their negations, linking each to distinct appropriate verbal or other behavioral responses (move an electronic arm; get an umbrella; print 'True'; print 'False'). But the diagonalized construction of the Gödel sentence prevents any first-order finite state machine from mechanically distinguishing between all Gödel sentences and their negations. Here the deductive apparatus and finite memory of a finite state machine are inescapably inadequate.

This may help to explain Lucas' puzzling statements about the mind's ability to 'see' that Gödel sentences are true. Lucas writes: "... the machine cannot produce the corresponding [Gödel] formula as being true. But *we* can see that the Gödelian formula is true ..."[16] If the rationale which I have offered is correct, then what Lucas means by 'seeing' that the Gödel sentence is true is just another way of describing the mind's ability to understand or grasp the meaning of the sentence, and thereby judge that it is true. It is the same intentionality by means of which the mind is able to understand whether or not a sentence says of itself that it is unprovable, in order nonmechanically to implement its intentional truth-value judging procedures.

Minds are different from machines because minds can understand or grasp the meaning of sentences like Gödel sentences and their negations. Machines can only imperfectly simulate the understanding or grasping of the meaning of sentences, as critics of mechanism have long maintained.[17] The mind is capable of understanding; it grasps the

meaning or significance of its pronouncements. But machines at their logical optimum are only rule-governed manipulators of syntax and robotic behavioral simulacra (in computer lingo they are mere 'symbol-crunchers'). Mechanism is therefore false; the mind is not a machine. There is no satisfactory mechanical model of the mind, and no logically possible finite state machine can have the same intentional capacities as the mind.

NOTES

[1] A. M. Turing: 1950, 'Computing Machinery and Intelligence', *Mind* **59**, 433–435.

[2] J. R. Lucas: 1961, 'Minds, Machines, and Gödel', *Philosophy* **36**; rpt., *Minds and Machines*, edited by Alan R. Anderson. Englewood Cliffs, Prentice-Hall, Inc., 1964, p. 43.

[3] Judson Webb: 1980, *Mechanism, Mentalism, and Metamathematics: An Essay on Finitism*, D. Reidel Publ. Co.; Dordrecht-Boston; Judson Webb: 1968, 'Metamathematics and the Philosophy of Mind', *Philosophy of Science* **35**, 156–178; Paul Benacerraf: 1967, 'God, the Devil, and Gödel', *The Monist* **51**, 9–32; Charles S. Chihara: 1972, 'On Alleged Refutations of Mechanism Using Gödel's Incompleteness Results', *The Journal of Philosophy* **6**, 507–526; Hilary Putnam: 1960, 'Minds and Machines', *Dimensions of Mind*, edited by Sidney Hook, New York University Press, New York, pp. 152–153; Daniel C. Dennett: 1978, 'The Abilities of Men and Machines', *Brainstorms: Philosophical Essays on Mind and Psychology*, Bradford Books, Montgomery, pp. 256–266; Dennett: 1972, 'Review of J. R. Lucas, *The Freedom of the Will*', *The Journal of Philosophy* **69**, 527–531; see also: J. J. C. Smart: 1961, 'Gödel's Theorem, Church's Theorem, and Mechanism', *Synthese* **13**, 105–110; William H. Hanson; 1971, 'Mechanism and Gödel's Theorems', *Brit. J. Phil. Sci.* **22**, 9–16; Peter Slezak: 1982, 'Gödel's Theorem and the Mind', *Brit. J. Phil. Sci.* **33**, 41–52. And below, notes 7 and 10.

[4] J. R. Lucas: 1970, *The Freedom of the Will*, Clarendon Press, Oxford, pp. 134–145; Lucas, 1971, 'Metamathematics and the Philosophy of Mind: A Rejoinder', *Philosophy of Science* **38**, 310–313 (reply to Webb); Lucas: 1968, 'Satan Stultified: A Rejoinder to Paul Benacerraf', *The Monist* **52**, 145–158.

[5] Lucas: 'Minds, Machines, and Gödel', pp. 56–57.

[6] Kurt Gödel: 1931, 'Über formal unentscheidbare Sätze der Principia mathematica und verwandter Systeme I', Monatshefte für Mathematik und Physik, **38**, 173–198.

[7] Some commentators have misinterpreted Lucas on the Gödel limitations of minds and machines. See David L. Boyer: 1983, 'J. R. Lucas, Kurt Gödel, and Fred Astaire', *The Philosophical Quarterly* **33**, 147: "... supposing that [formal system] S has a Gödel sentence G, Lucas (goaded by the mechanist's claim) could take the trouble to figure out G given a specification of P [a program purported to simulate Lucas] or S. He [Lucas] could then realize the truth of G, and this would be a proof of G. Since proving G is something he could do that P couldn't, it would follow that P doesn't simulate him after all." This is not only a mistaken exposition of Lucas, but reflects a fundamental

misunderstanding of Gödel's first theorem and the nature of mathematical proof. To 'realize' that something is true cannot constitute a *proof* of its truth. The Gödel sentence cannot be proved by mind or machine, since if the sentence is true, it is *un*provable. To claim that the mind can correctly but noncomputationally judge the truth-value of Gödel sentences and their negations does not contradict Church's thesis that every calculable function is recursive; Alonzo Church: 1935, 'An Unsolvable Problem of Elementary Number Theory', *Bulletin of the American Mathematical Society*, **41**, 332–333. See Benacerraf: 'God, the Devil, and Gödel', pp. 19–21; Lucas: 1984, 'Lucas, Gödel, and Astaire: A Rejoinder', *The Philosophical Quarterly* **34**, 507–508. Lucas maintains that the mind can 'see' that the Gödel sentence is true (not that the mind can prove it), and that no finite state machine can adequately simulate this ability. (Lucas' somewhat vague metaphorical term 'see' is explained in section 7.)

[8] This notation derives in part from Joseph R. Shoenfield: 1967, *Mathematical Logic*, Addison-Wesley, Reading, pp. 123–132. Gödel's original formalization uses 'Bew' for '*beweisbar*' (provable).

[9] See David R. Auerbach: 1985, 'Intensionality and the Gödel Theorems', *Philosophical Studies* **48**, 337–351.

[10] Similar arguments are at the heart of objections considered by Putnam, Benacerraf, Chihara, and others. See Anthony Hutton: 1976, 'This Gödel is Killing Me', *Philosophia*, **6**, 135–144; David Lewis: 'Lucas Against Mechanism', *Philosophy* **44**, 231–237.

[11] Lucas: 'Minds, Machines, and Gödel', pp. 56–57.

[12] Webb: 'Metamathematics and the Philosophy of Mind', pp. 161–169.

[13] Dennett: *Brainstorms*, pp. 264–266.

[14] A. M. Turing: 1937, 'On Computable Numbers, With An Application to the Entscheidungsproblem', *Proceedings of the London Mathematical Society*, **42**, 230–265. See also, George Boolos and Richard Jeffrey: 1974, *Computability and Logic*, Cambridge University Press, London, pp. 43–50.

[15] Gottlob Frege: 1918, 'Thoughts' ['Der Gedanke'], *Beiträge zur Philosophie des deutschen Idealismus*; rpt., *Logical Investigations*, edited by P. T. Geach, translated by P. T. Geach and R. H. Stoothoff, Basil Blackwell, Oxford, 1977, pp. 5–8, 24–26, 28–30.

[16] Lucas: 'Minds, Machines, and Gödel', p. 47. See also pp. 44, 48, 49, 52.

[17] Michael Scriven: 1953, 'The Mechanical Concept of Mind', *Mind* **62**, 230–234, 237–240. Paul Ziff: 1959, 'The Feelings of Robots', *Analysis* **19**; rpt., *Minds and Machines*, edited by Anderson, pp. 99–102. Wallace I. Matson: 1982, *Sentience*, University of California Press, Berkeley; Keith Gunderson: 1985, *Mentality and Machines*, 2nd ed., University of Minnesota Press, Minneapolis.

Manuscript received 18 December 1985

Department of Philosophy
The Pennsylvania State University
246 Sparks Building
University Park, PA 16802
U.S.A.