edX

Course › Godel's... › Introdu... › Introdu...

Introduction

Video: Introduction to Lecture 10

1:35 / 1:35 1.50x

One reason the theorem is philosophically important

is that it can be used to argue that we can never

be absolutely certain about the truth

of our mathematical theories.

It's not obvious why this is so, but I'll say more about it

later and you'll see it's an astonishing result.

End of transcript. Skip to the start.

Video

Download video file

Transcripts

Download SubRip (.srt) file

Download Text (.txt) file

Exercise

1/1 point (ungraded)

Show that the following formulations of Gödel's Theorem are equivalent:

> **Gödel's Incompleteness Theorem**
> Let L be a (rich enough) arithmetical language. Then no Turing Machine M is such that when run on an empty input: M runs forever, outputting sentences of L, every true sentence of L is eventually output by M, and no false sentence of L is ever output by M.
>
> **Gödel's Incompleteness Theorem**
> Let L be a (rich enough) arithmetical language. Then no Turing Machine M' is such that whenever it is given a sentence of L as input it outputs "1" if the sentence is true and "0" if the sentence is false. (In other words: *no Turing machine can decide the truth-value of the sentences of* L.)

(Hint: you may assume that there is a Turing Machine that outputs every sentence of L and nothing else, and that every sentence in L has a negation that is also in L.)

Done

Explanation

Here is a characterization of M', assuming M exists. To run M' with ϕ as input, run M as a subroutine and wait until it outputs either ϕ or its negation. We know that it will eventually output one or the other because we know that M eventually outputs every true sentence of L (and because we know that one or the other is true). We also know that whichever of ϕ or its negation is output by M must be true, since M never outputs any falsehood. So M' can use this information to decide whether to output "0" or "1".

Here is a characterization of M, assuming M' exists. We know that there is a Turing Machine M^S, which outputs every sentence in L (and nothing else). To run M on an empty input, run M^S as a subroutine. Each time M^S outputs a sentence, use M' to check whether the sentence is true or false. If it's true, output that sentence; if not, keep going.

Submit

Answers are displayed within the problem

Discussion

Topic: Week 10 / Introduction

Hide Discussion

Add a Post

Show all posts by recent activity

There are no posts in this topic yet.

✖