

Re: Can returning a value change the value itself (in the Halting Problem)

Source: <http://sci.tech-archive.net/Archive/sci.logic/2004-08/3040.html>

From: Marc Goodman (marc.goodman_at_comcast.net)

Date: 08/24/04

Date: Tue, 24 Aug 2004 18:14:09 GMT

peter_douglass wrote:

> "Marc Goodman" <marc.goodman@comcast.net> wrote in message
> news:UEKWc.36354\$9d6.14638@attbi_s54...

>

>

>>The context is a TM with access to its own state transition
>>table that can calculate the check sum of its own state
>>transition table. Do you finally see that even making
>>a "trivial" change to such a TM can cause the result produced
>>by such a TM to change? Or do we have to go through this
>>line of argument three more times?

>

>

> Could you explain what you mean by a TM with access to its
> own state transition table? What does access mean?

You'd have to ask Peter exactly what he means, it's his extension. I'm just arguing that if you supply a TM with an extension that allows it to introspect, then you are no longer guaranteed that the "code" of the TM has locality. So, making the claim that a TM with such an extension can be trivially modified to return its results without changing the results it is computing is far from obvious.

> A TM as defined by Turing makes state transitions based upon
> its current state and the value read by the "head". Does this
> new machine make transitions based upon somehow "reading"
> the transition table? It would seem that if so, it would need
> a "bigger" transition table that takes as inputs what is read
> by the "head" and what is read by whatever reads the state
> transition table. Or perhaps the transition table is present
> on a tape to begin with, and there is some magical property
> that guarantees that what is on the tape matches the
> transition table that is actually used. Please explain.

>

> What I would like is a formalization this machine with access

sci.logic: Re: Can returning a value change the value itself (in the Halting Problem)

- > *to its own state transition table. (It shouldn't be called a*
- > *TM, because it is not, and no TM as defined by Turing*
- > *behaves that way")*
- >
- > --PeterD
- >
- >