

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

Source: <http://sci.tech-archive.net/Archive/sci.logic/2004-08/3033.html>

From: Marc Goodman (marc.goodman_at_comcast.net)

Date: 08/24/04

Date: Tue, 24 Aug 2004 17:11:16 GMT

Martin Shobe wrote:

> *Of course the value changed. The *input* changed.*

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?

But, both

> *functions implement exactly the same, um, function.*

You are free to call it "different input" if you want, but the fact remains that if you are talking about a TM with access to its own state transition table, the line between "input" and "process" has been blurred. To the extent that a seemingly local change to the TM can have a global change to the result produced by the TM.

Apply the **same**

> *input to both functions and you will receive the *same* output.*

Yes. Too bad the input changes when you change the code. I can keep repeating this if you find it helpful.

That

> *was my arguement (you know, the one you called naive),*

It was "Ghost in the Machine's argument that I called "naive." If you were arguing the same point, then... I stand by the above explanation of why a local change to a TM that has access to its own state transition table might have a non-local affect on the result produced.

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

which is what

> *you posted the above challenge to.*

>

> *Martin*

>