

Re: What is the Result from Invoking this Halt Function?

Source: <http://sci.tech-archive.net/Archive/sci.logic/2004-08/0961.html>

From: Simon G Best (*s.g.best_at_btopenworld.com*)

Date: 08/07/04

Date: Sat, 7 Aug 2004 12:14:52 +0000 (UTC)

Peter Olcott wrote:

> "David C. Ullrich" <ullrich@math.okstate.edu> wrote in message
news:f7v6h0dmtD1strl0fildjh6njr8oonvuns@4ax.com...

>

>>it's amazing how you seem unable to follow the simple proof that this
>>is not possible, regardless of how many times it's explained. one
>>more time:

>>

>>suppose that *P* is a program that does what you say – a correct halt
>>analyzer that refrains from returning the result under some
>>circumstances. now let *Q* be a program that does exactly what *P*
>>does, except that *Q* always does return 1 or 0. the proof shows
>>that *Q* cannot exist. hence *P* cannot exist. *qed.*

>

> Slight little error here that makes all the difference.
> You say that the conclusion that *Q* can't exist entails that *P* can't exist.
> Yet it is not that *Q* can't exist. Merely that *Q* sometimes reports results
> that are not correct.

Q always returns what *P* would return when *P* is invoked under those
circumstances in which *P* would return a result. Therefore, if *Q*
"sometimes reports results that are not correct", then it must be that *P*
"sometimes reports results that are not correct", and therefore *P* is not
a halt analyzer.

> This is a perfect example of a (otherwise) very well formed refutation,
> that just happens to be incorrect. I think that this is the basic form of
> all of the official and published refutations.

>

> This is either the key most crucial point that I am missing, or the key
> crucial point that everyone else (except me) has missed for sixty eight
> years.

The most crucial point that you are deliberately missing is that a
Turing Machine doesn't know who or what gave it its input tape. What
David Ullrich is doing is pointing out that even if we overlook this

sci.logic: Re: What is the Result from Invoking this Halt Function?

fact, we can still prove that P cannot exist.

If it comes down to it, we can have Q invoke P, and lie to P by pretending that it's something else that's invoking P. Turing Machines only 'know' what we tell them, and we can lie to them (very easily, because they only have themselves and their tapes to go on, and we have complete control over what's on the tapes).

> *Let's take this ALL THE WAY. Please explain to me exactly and precisely how is it that Q can not exist?*

That's already been done. See my thought experiment and accompanying proof. As Q, unlike P, /always/ returns its result /under all circumstances/, my thought experiment and proof apply to Q.

> *The way that I see it can perfectly well exist, yet sometimes provide incorrect results. I see nothing at all about Q that would in any possible way, by any stretch of the imagination prevent it from existing.*

Then you are conceding that Turing's proof is correct.

Simon