

Ruby - Misc #20222

Dedup-ing clarification

01/29/2024 04:33 PM - kddnewton (Kevin Newton)

Status:	Closed	
Priority:	Normal	
Assignee:		
Description		
<pre>source = %q{"foo".freeze.equal?("foo".freeze)}</pre> <pre>RubyVM::InstructionSequence.compile(source).eval # =&gt; true</pre> <pre>RubyVM::InstructionSequence.compile_option = false</pre> <pre>RubyVM::InstructionSequence.compile(source).eval # =&gt; false</pre> "foo".freeze uses opt_str_freeze when optimizations are turned on, which also deduplicates. This means this code has different behavior depending on if optimizations are turned on or off.  To be clear, I'm not saying whether or not this is a problem. I'm asking if this is desired behavior?		
Related issues:		
Related to Ruby - Feature #8992: Use String#freeze and compiler tricks to rep...		Closed

History

#1 - 02/08/2024 01:17 AM - alanwu (Alan Wu)

Seems like a valid optimization to me and is similar to identity of small integers versus heap allocated bignums.

#2 - 02/09/2024 11:23 AM - byroot (Jean Boussier)

- Related to Feature #8992: Use String#freeze and compiler tricks to replace "str"f suffix added

#3 - 02/09/2024 11:25 AM - byroot (Jean Boussier)

For the full context see [Feature [#8992](#)].

It's true that it's a bit questionable that it's treated as an optimization, but realistically speaking no-one is compiling code without optimizations, it's just for MRI developers.

Also erubi and a few other projects heavily rely on this for performance.

#4 - 02/09/2024 03:16 PM - kddnewton (Kevin Newton)

- Status changed from Open to Closed

Thank you! This helps explain it.