Ruby master - Bug #13700

 Enumerable#sum may not work for Ranges subclasses due to optimization

06/30/2017 09:05 AM - sos4nt (Stefan Schüßler)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>ruby -v:</td>
<td>ruby 2.4.1p111 (2017-03-22 revision 58053) [x86_64-darwin15]</td>
</tr>
<tr>
<td>Backport:</td>
<td>2.2: UNKNOWN, 2.3: UNKNOWN, 2.4: UNKNOWN</td>
</tr>
</tbody>
</table>

Description

Enumerable#sum is optimized for integer ranges. Unfortunately, this can break subclasses:

class StepTwoRange < Range
  def each(&block)
    step(2, &block)
  end
end

r = StepTwoRange.new(0, 10)
r.to_a #=> [0, 2, 4, 6, 8, 10]

r.to_a.sum #=> 30
r.sum #=> 55

The optimization should therefore only be triggered for instances of Range and not for instances of subclasses. (or more specifically, not for subclasses overriding each)

If this behavior is intentional, it should at least be documented.

Associated revisions

Revision e496e965 - 08/25/2019 07:50 PM - jeremyevans (Jeremy Evans)

Document that Enumerable#sum may not respect redefinition of Range#each

It already documented that it may not respect redefinition of Integer#+.

Fixes [Bug #13700]

History

#1 - 06/30/2017 09:13 AM - sos4nt (Stefan Schüßler)
- Description updated

#2 - 06/30/2017 09:29 AM - sos4nt (Stefan Schüßler)
- Description updated

#3 - 06/30/2017 11:30 AM - shevegen (Robert A. Heiler)

Reminds me a bit of what hanmac wrote elsewhere; I can't find it right now and forgot it mostly already but I think he also mentioned some unexpected behaviour when ... subclassing I think? Or some custom class that he wrote...

#4 - 06/30/2017 11:39 AM - Hanmac (Hans Mackowiak)

shevegen (Robert A. Heiler) wrote:

Reminds me a bit of what hanmac wrote elsewhere; I can't find it right now and forgot it mostly already but I think he also mentioned some unexpected behaviour when ... subclassing I think? Or some custom class that he wrote...

my comment was for https://bugs.ruby-lang.org/issues/13663
with String#upto you can't overwrite the internal String#<=>

03/19/2022
Side note: in my opinion, Enumerable should not check the receiver's class to provide specific optimizations. Instead, Range should override sum and handle the optimization itself. Range#sum would also be a better place to document this behavior. (it still doesn't solve the subclassing issue, though)

Applied in changeset gilie496e96547b64c3a2fa6f285c3bc9bd21a245ac6.

Document that Enumerable#sum may not respect redefinition of Range#each

It already documented that it may not respect redefinition of Integer#+

Fixes [Bug #13700]