

Ruby master - Feature #16995

Sets: <=> should be specialized

06/26/2020 08:43 PM - marcandre (Marc-Andre Lafortune)

Status: Open	
Priority: Normal	
Assignee:	
Target version:	
Description	
<p>This is quite minor, but Set#<=> should be refined.</p> <p>Reminder: Set defines <, >, etc. as inclusion, but does not have a corresponding <=>:</p> <pre>Set[1] < Set[1, 2] # => true Set[1] <=> Set[1, 2] # => nil, should be -1 Set[1] <=> Set[2] # => nil, ok, not orderable</pre> <p>The official stated reason for Set to <i>not</i> implement is that some sets are not comparable. That is exactly what nil result type is for IMO. Sets are partially ordered and <=> should reflect that. https://en.wikipedia.org/wiki/Partially_ordered_set</p> <pre>Set[1] < Set[1, 2] # => true [Set[1], Set[1, 2]].sort # => ArgumentError, should be [Set[1], Set[1, 2]] [Set[1], Set[2]].sort # => ArgumentError, ok, can't be ordered</pre> <p>This is <i>exactly the same</i> idea as Class, which correctly refines <=>:</p> <pre>Array < Enumerable # => true Array <=> Enumerable # => -1, ok [Array, Enumerable].sort # => [Array, Enumerable] [Array, String].sort # => ArgumentError (comparison of Class with Class failed), ok</pre>	
Related issues:	
Related to Ruby master - Feature #16989: Sets: need ♥️	Assigned

History

#1 - 06/26/2020 08:47 PM - marcandre (Marc-Andre Lafortune)

- Related to Feature #16989: Sets: need ♥️ added