

Ruby master - Feature #8422

add Enumerable#reverse_sort and Enumerable#reverse_sort_by

05/18/2013 05:09 PM - Hanmac (Hans Mackowiak)

Status:	Feedback
Priority:	Normal
Assignee:	
Target version:	
Description they are better when you want descending order, enum.sort.reverse does work too, but it is too slow because it needs to build that result array twice they both can be defined just like #sort and #sort_by but negates the result of the compare block	

History

#1 - 05/18/2013 06:46 PM - matz (Yukihiro Matsumoto)

- Status changed from Open to Feedback

It can be done by `a.sort{|a,b| b<=>a}`. Do we really need to define new methods?

Matz.

#2 - 05/18/2013 11:31 PM - marcandre (Marc-Andre Lafortune)

matz (Yukihiro Matsumoto) wrote:

It can be done by `a.sort{|a,b| b<=>a}`.

That will typically be much slower though.

Currently, the best way to do a reverse sort, performance-wise, is `a.sort.reverse!`

For a random 100 number array, for example:

`sort.reverse!` is faster than `sort.reverse` by $10\% \pm 1.0\%$
`sort.reverse` is faster than `sort_by` by $6x \pm 0.1$
`sort_by` is faster than `sort` by $60\% \pm 1.0\%$

Code: <https://gist.github.com/marcandre/5604592>

#3 - 05/22/2013 02:31 AM - Anonymous

-1, feature creep

But I'm for defining `reverse!` as $O(1)$, not really reversing anything, just treating the last element as first and first as last. (I do not know how collections are exactly implemented, so I am not sure whether this is possible.)

#4 - 06/07/2013 12:02 PM - drbrain (Eric Hodel)

How do you sort an infinite Enumerable?

```
ruby -rprime -e 'p Prime.each.reverse_sort.take 10'
```

#5 - 06/08/2013 02:41 PM - marcandre (Marc-Andre Lafortune)

drbrain (Eric Hodel) wrote:

How do you sort an infinite Enumerable?

```
ruby -rprime -e 'p Prime.each.reverse_sort.take 10'
```

I'm not sure what your point is. How is that different from `p Prime.each.sort.take 10`? Moreover, an alternative request could be to add `Array#reverse_sort_by`.

Maybe we should consider a variation on sort that allows multiple criteria and independent sort directions?

#6 - 06/12/2013 03:21 AM - drbrain (Eric Hodel)

Neither `Prime.each.reverse_sort.take 10` nor `Prime.each.sort.take 10` will return as there is no last value for infinite streams. You can neither reverse nor sort an infinite stream.

I believe adding the feature to `Enumerable` doesn't fit with its existing functionality (all beginning-of-stream oriented).

We need feedback from the submitter. Without it this issue should be closed.

#7 - 06/14/2013 01:33 AM - marcandre (Marc-Andre Lafortune)

drbrain (Eric Hodel) wrote:

Neither `Prime.each.reverse_sort.take 10` nor `Prime.each.sort.take 10` will return as there is no last value for infinite streams.

Of course. But you are aware that `Enumerable` has a `sort` method that acts as a shortcut to `to_a.sort`?

We need feedback from the submitter. Without it this issue should be closed.

The relevant question here is "should there be a dedicated way to do a reverse sort". The question of where that method should be (`Array` or `Enumerable`) is accessory.

#8 - 06/14/2013 03:19 AM - Hanmac (Hans Mackowiak)

my idea would be this, for ruby it may be cood if its converted into C++ code

```
module Enumerable
  def reverse_sort(&block)
    block ||= proc {|x,y| x <=> y }
    sort {|x,y| - block.call(x,y)}
  end

  def reverse_sort_by
    return to_enum(__method__) unless block_given?
    sort_by{|x| - yield(x)}
  end
end
```