

Ruby - Bug #14748

Inconsistent (and regressed from 2.4) behavior on `Array#min` and max between array literals and arrays

05/10/2018 11:41 PM - nelhage (Nelson Elhage)

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	ruby 2.5.1p57 (2018-03-29 revision 63029) [x86_64-darwin17]	Backport: 2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN
Description Array#min and Array#max have different behavior depending on whether or not they are called with array literals: <pre>\$ ruby -e 'puts([1.0, 1].min); one_f = 1.0; one = 1; puts([one_f, one].min)'</pre> <pre>1.0 1</pre> This is a regression from 2.4, where both forms would return 1.0. I believe this to be a bug introduced by https://bugs.ruby-lang.org/issues/12172 -- disassembly reveals that only the second form uses opt_newarray_min		

History

#1 - 05/11/2018 03:04 AM - Hanmac (Hans Mackowiak)

Related to this?

<https://bugs.ruby-lang.org/issues/14350>

#2 - 05/11/2018 03:09 AM - mame (Yusuke Endoh)

- Status changed from Open to Closed

This is already fixed in trunk at r61766. Please wait for ruby 2.5.2.

(Note that [#12172](#) is innocent because opt_newarray_min is included in 2.4 :-)