

Ruby master - Bug #4591

(1.5...2).max #=> 1 (Range#max)

04/21/2011 12:42 AM - tarui (Masaya Tarui)

Status: Closed	
Priority: Normal	
Assignee: tarui (Masaya Tarui)	
Target version: 1.9.3	
ruby -v: -	Backport:

Description

```
(1.5...2).max #> 1
begin
```

```
Range#end Integer#end-1
begin Integer
begin Integer#end Integer
max Error
```

Index: range.c

```
--- range.c (31313)
+++ range.c (31313)
@@ -670,6 +670,9 @@
rb_raise(rb_eTypeError, "cannot exclude non Integer end value");
```

```
}
if (c == 0) return Qnil;
```

- if (!FIXNUM_P(b) && !rb_obj_is_kind_of(b, rb_cInteger)) {
- rb_raise(rb_eTypeError, "cannot exclude end value with non Integer begin value");
- } if (FIXNUM_P(e)) { return LONG2NUM(FIX2LONG(e) - 1); } Index: test/ruby/test_range.rb
===== test/ruby/test_range.rb (31313) +++ test/ruby/test_range.rb (31313) @@ -68,6 +68,8 @@ assert_equal(2.0, (1.0..2.0).max) assert_equal(nil, (2.0..1.0).max) assert_raise(TypeError) { (1.0...2.0).max }
- assert_raise(TypeError) { (1...1.5).max }
- assert_raise(TypeError) { (1.5...2).max }
- assert_equal(-0x80000002, ((-0x80000002)...(-0x80000001)).max)

Related issues:

Related to Ruby master - Bug #974: Range#max	Closed	01/04/2009
Related to Ruby master - Bug #4577: (int...float).max should not raise an error	Rejected	04/14/2011

Associated revisions

Revision d22e08f9 - 07/10/2011 12:47 AM - tarui (Masaya Tarui)

- range.c (range_max): fix behavior with excluded end value. [Bug #4591]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@32482 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 32482 - 07/10/2011 12:47 AM - tarui (Masaya Tarui)

- range.c (range_max): fix behavior with excluded end value. [Bug #4591]

Revision 32482 - 07/10/2011 12:47 AM - tarui (Masaya Tarui)

- range.c (range_max): fix behavior with excluded end value. [Bug #4591]

Revision 32482 - 07/10/2011 12:47 AM - tarui (Masaya Tarui)

- range.c (range_max): fix behavior with excluded end value. [Bug #4591]

Revision 32482 - 07/10/2011 12:47 AM - tarui (Masaya Tarui)

- range.c (range_max): fix behavior with excluded end value. [Bug #4591]

Revision 32482 - 07/10/2011 12:47 AM - tarui (Masaya Tarui)

- range.c (range_max): fix behavior with excluded end value. [Bug #4591]

Revision 32482 - 07/10/2011 12:47 AM - tarui (Masaya Tarui)

- range.c (range_max): fix behavior with excluded end value. [Bug #4591]

History

#1 - 04/21/2011 11:03 AM - Anonymous

=begin

Issue [#4591](#) has been reported by Masaya Tarui.

Bug [#4591](#): (1.5...2).max #=> 1 (Range#max)
<http://redmine.ruby-lang.org/issues/4591>

Author: Masaya Tarui
Status: Open
Priority: Normal
Assignee:
Category: core
Target version:
ruby -v: ruby 1.9.3dev (2011-04-20 trunk 31311) [i386-mswin32_100]

```

(1.5...2).max #=> 1
begin
  Range#end Integer#end-1
  begin Integer#end Integer#end
  begin Integer#end Integer#end
  max Error

```

Ruby range_max

- 1) range
- 2)

```

(1)
type error
error

```

```

end-1 begin end
=end

```

#2 - 04/21/2011 12:23 PM - mrkn (Kenta Murata)

- ruby -v changed from ruby 1.9.3dev (2011-04-20 trunk 31311) [i386-mswin32_100] to -

=begin

test_range.rb test_max

```

assert_equal(2, (1..2).max)
assert_equal(nil, (2..1).max)
assert_equal(1, (1...2).max)

assert_equal(2.0, (1.0..2.0).max)
assert_equal(nil, (2.0..1.0).max)
assert_raise(TypeError) { (1.0...2.0).max }

assert_equal(-0x80000002, ((-0x80000002)...(-0x80000001)).max)

assert_equal(0, (0..0).max)
assert_equal(nil, (0...0).max)

```

```

#####

```

```

assert_equal(nil, (2..1).max)
assert_equal(nil, (2.0..1.0).max)
assert_raise(TypeError) { (1.0...2.0).max }

```

```

###tarui #####

```

```

assert_raise(TypeError) { (1.5...2).max }
assert_raise(TypeError) { (1...1.5).max }

```

```

Range#max #####
##### RDoc ##

```

Returns the maximum value in rng. The second uses the block to compare values. Returns nil if the first value in range is larger than the last value.

```

#####

```

```

Range [] Enumerable [] include ##### Range#max [] Enumerable#max #####
##### Range#each ##### succ #####
[] Float ##### TypeError #####

```

```

assert_equal(nil, (2.0..1.0).max)

```

```

#####

```

```

##### succ #####
tarui #####

```

```

assert_raise(TypeError) { (1...1.5).max }

```

```

#####TypeError ##### 1 #####

```

On 2011-04-21 at 8:55, KOSAKI Motohiro wrote:

```

###Ruby#####range#####max#####

```

- 1) range#####
- 2) #####

```

##### succ #####
#####
##### Float [] BigDecimal
##### inexact #####

```

```

Float [] BigDecimal #####
#####

```

```

assert_raise(TypeError) { (1.0...2.0).max }

```

```

#####
### 1 (### 1 #####) [] 1.0 ... 2.0 #####
#####

```

```

Float [] 1.0 []C [] double [] IEEE754##### [1, 1 + 1/251] #####
##### 1.0 ... 2.0 ### 1 #####
#####

```

```

# ##### 1.0 ##### 1 #####

```

```
# 1.0 1
```

```
--
Kenta Murata
Sent with Sparrow
=end
```

#3 - 04/21/2011 01:23 PM - tarui (Masaya Tarui)

```
=begin
snip

assert_equal(nil, (2..1).max)
assert_equal(nil, (2.0..1.0).max)
assert_raise(TypeError) { (1.0...2.0).max }

tarui

assert_raise(TypeError) { (1.5...2).max }
assert_raise(TypeError) { (1...1.5).max }

Range#max
snip
```

```
assert_raise(TypeError) { (1.0...2.0).max }
#974
```

```
matz> *
matz> *
matz>
```

```
end
begin
```

```
bug
max
```

```
#4577
```

```
--
(Masaya TARUI)
No Tool,No Life.
=end
```

#4 - 04/21/2011 01:23 PM - tarui (Masaya Tarui)

```
=begin
begin
end
Range
error

Ruby
range
max

1) range
2)

(1)
type error

end-1
begin
end
```

```
(1..1.5).max
#=>1.5
```

--
Masaya TARUI
No Tool, No Life.
=end

#5 - 06/11/2011 05:02 PM - ko1 (Koichi Sasada)

- Status changed from Open to Assigned
- Assignee set to mrkn (Kenta Murata)

#6 - 06/11/2011 10:29 PM - matz (Yukihiko Matsumoto)

Range#max

In message "Re: [ruby-dev:43732] [Ruby 1.9 - Bug #4591]Assigned.max #=> 1 (Range#max)"
on Sat, 11 Jun 2011 17:03:00 +0900, Koichi Sasada redmine@ruby-lang.org writes:

```
Range#end Integer#end-1  
begin Integer  
begin Integer#end Integer  
|max Error  
|  
|  
|
```

#7 - 06/11/2011 10:29 PM - matz (Yukihiko Matsumoto)

Range#max

In message "Re: [ruby-dev:43732] [Ruby 1.9 - Bug #4591]Assigned.max #=> 1 (Range#max)"
on Sat, 11 Jun 2011 17:03:00 +0900, Koichi Sasada redmine@ruby-lang.org writes:

```
Range#end Integer#end-1  
begin Integer  
begin Integer#end Integer  
|max Error  
|  
|  
|
```

#8 - 06/26/2011 07:03 PM - nahi (Hiroshi Nakamura)

- Target version set to 1.9.3

#9 - 07/08/2011 05:56 PM - mrkn (Kenta Murata)

```
bug  
max  
#4577
```

issue max
bug

#10 - 07/08/2011 06:11 PM - naruse (Yui NARUSE)

- Assignee changed from mrkn (Kenta Murata) to tarui (Masaya Tarui)

#11 - 07/10/2011 09:47 AM - tarui (Masaya Tarui)

- Status changed from Assigned to Closed
- % Done changed from 0 to 100

This issue was solved with changeset r32482.
Masaya, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

-
- range.c (range_max): fix behavior with excluded end value. [Bug [#4591](#)]