

## Ruby master - Bug #4473

### Calling return within begin still executes else

03/06/2011 04:37 PM - thuss (Todd Huss)

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b> mame (Yusuke Endoh)	
<b>Target version:</b> 1.9.3	
<b>ruby -v:</b> -	<b>Backport:</b>
<b>Description</b>	
<pre>=begin I see this issue in 1.9.2-p0 through the current version 1.9.2-p136 on Mac OS X. The following code prints 'else executed'. Whereas in Ruby 1.8.x it would not execute the else statement if you called return from within a begin block.  In summary, I would expect the following code to not print anything, but in Ruby 1.9.2 it actually prints 'else executed':</pre>	
<b>This code prints 'else executed'</b>	
<pre>def test_return_in_method begin return rescue puts 'rescue executed' else puts 'else executed' end end test_return_in_method =end</pre>	

#### Associated revisions

##### Revision b0f115da - 07/10/2011 02:42 AM - mame (Yusuke Endoh)

- parse.y (reduce\_nodes\_gen): NODE\_RETURN in rescue body must not be reduced when there is an else clause. This caused bizarre behavior in [Bug #4473] [ruby-core:35629] [ruby-core:37884].

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@32483 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

##### Revision 32483 - 07/10/2011 02:42 AM - mame (Yusuke Endoh)

- parse.y (reduce\_nodes\_gen): NODE\_RETURN in rescue body must not be reduced when there is an else clause. This caused bizarre behavior in [Bug #4473] [ruby-core:35629] [ruby-core:37884].

##### Revision 32483 - 07/10/2011 02:42 AM - mame (Yusuke Endoh)

- parse.y (reduce\_nodes\_gen): NODE\_RETURN in rescue body must not be reduced when there is an else clause. This caused bizarre behavior in [Bug #4473] [ruby-core:35629] [ruby-core:37884].

##### Revision 32483 - 07/10/2011 02:42 AM - mame (Yusuke Endoh)

- parse.y (reduce\_nodes\_gen): NODE\_RETURN in rescue body must not be reduced when there is an else clause. This caused bizarre behavior in [Bug #4473] [ruby-core:35629] [ruby-core:37884].

##### Revision 32483 - 07/10/2011 02:42 AM - mame (Yusuke Endoh)

- parse.y (reduce\_nodes\_gen): NODE\_RETURN in rescue body must not be reduced when there is an else clause. This caused bizarre behavior in [Bug #4473] [ruby-core:35629] [ruby-core:37884].

##### Revision 32483 - 07/10/2011 02:42 AM - mame (Yusuke Endoh)

- parse.y (reduce\_nodes\_gen): NODE\_RETURN in rescue body must not be reduced when there is an else clause. This caused bizarre behavior in [Bug #4473] [ruby-core:35629] [ruby-core:37884].

#### Revision 32483 - 07/10/2011 02:42 AM - mame (Yusuke Endoh)

- parse.y (reduce\_nodes\_gen): NODE\_RETURN in rescue body must not be reduced when there is an else clause. This caused bizarre behavior in [Bug #4473] [ruby-core:35629] [ruby-core:37884].

## History

---

### #1 - 03/06/2011 04:40 PM - thuss (Todd Huss)

- File ruby-1.9.2-return-else.rb added

```
=begin
The code didn't paste so well so I'm attaching the source code
=end
```

### #2 - 04/06/2011 09:07 PM - mayank.kohaley (Mayank Kohaley)

```
=begin
I can see the same issue happening on Windows also.
=end
```

### #3 - 06/10/2011 10:53 PM - ko1 (Koichi Sasada)

- ruby -v changed from ruby 1.9.2p136 (2010-12-25 revision 30365) [x86\_64-darwin10.6.0] to -

Hi,

I recognize this issue. Matz, what do you think about it?

(2011/04/06 21:12), Mayank Kohaley wrote:

Issue [#4473](#) has been updated by Mayank Kohaley.

## I can see the same issue happening on Windows also.

Bug [#4473](#): Calling return within begin still executes else  
<http://redmine.ruby-lang.org/issues/4473>

Author: Todd Huss  
Status: Open  
Priority: Normal  
Assignee:  
Category:  
Target version:  
ruby -v: ruby 1.9.2p136 (2010-12-25 revision 30365) [x86\_64-darwin10.6.0]

I see this issue in 1.9.2-p0 through the current version 1.9.2-p136 on Mac OS X. The following code prints 'else executed'. Whereas in Ruby 1.8.x it would not execute the else statement if you called return from within a begin block.

In summary, I would expect the following code to not print anything, but in Ruby 1.9.2 it actually prints 'else executed':

## This code prints 'else executed'

```
def test_return_in_method
begin
return
rescue
puts 'rescue executed'
else
puts 'else executed'
end
end
test_return_in_method
```

--  
// SASADA Koichi at atdot dot net

**#4 - 06/10/2011 10:59 PM - matz (Yukihiko Matsumoto)**

Hi,

In message "Re: [ruby-core:36926] Re: [Ruby 1.9 - Bug #4473] Calling return within begin still executes else"  
on Fri, 10 Jun 2011 22:36:19 +0900, SASADA Koichi [ko1@atdot.net](mailto:ko1@atdot.net) writes:

|I recognize this issue. Matz, what do you think about it?

I think it should print nothing, like 1.8 does.

matz.

**#5 - 06/10/2011 11:03 PM - ko1 (Koichi Sasada)**

- Assignee set to ko1 (Koichi Sasada)

Yugui-san,

This fix will introduce incompatibility from 1.9.2. Is it acceptable?

**#6 - 06/10/2011 11:23 PM - ko1 (Koichi Sasada)**

(2011/06/10 22:56), Yukihiko Matsumoto wrote:

In message "Re: [ruby-core:36926] Re: [Ruby 1.9 - Bug #4473] Calling return within begin still executes else"  
on Fri, 10 Jun 2011 22:36:19 +0900, SASADA Koichi [ko1@atdot.net](mailto:ko1@atdot.net) writes:

|I recognize this issue. Matz, what do you think about it?

I think it should print nothing, like 1.8 does.

OK. I'll check it.

--  
// SASADA Koichi at atdot dot net

**#7 - 06/11/2011 04:21 PM - nobu (Nobuyoshi Nakada)**

I bet that it is a bug in 1.8.

**#8 - 06/13/2011 11:23 PM - ko1 (Koichi Sasada)**

(2011/06/11 16:21), Nobuyoshi Nakada wrote:

I bet that it is a bug in 1.8.

Matz, could you give me a comment, again?

--  
// SASADA Koichi at atdot dot net

**#9 - 06/14/2011 12:23 AM - matz (Yukihiko Matsumoto)**

Hi,

In message "Re: [ruby-core:37092] Re: [Ruby 1.9 - Bug #4473] Calling return within begin still executes else"  
on Mon, 13 Jun 2011 23:04:46 +0900, SASADA Koichi [ko1@atdot.net](mailto:ko1@atdot.net) writes:

|(2011/06/11 16:21), Nobuyoshi Nakada wrote:

|> I bet that it is a bug in 1.8.

|Matz, could you give me a comment, again?

He didn't disclosed the reason why it was a 1.8 bug. If he has a reasonable reason, I would accept it.

matz.

**#10 - 06/26/2011 06:46 PM - nahi (Hiroshi Nakamura)**

- Target version set to 1.9.3

**#11 - 07/08/2011 11:57 PM - mame (Yusuke Endoh)**

- Assignee changed from ko1 (Koichi Sasada) to mame (Yusuke Endoh)

Hello,

```
|(2011/06/11 16:21), Nobuyoshi Nakada wrote:  
> I bet that it is a bug in 1.8.  
|  
|Matz, could you give me a comment, again?
```

He didn't disclosed the reason why it was a 1.8 bug. If he has a reasonable reason, I would accept it.

I have no idea which behavior is better. But, at least, the behavior of 1.9 seems to be caused by a wrong optimization:

```
def foo  
begin  
return  
rescue  
else  
p :foo!  
end  
end
```

foo #=> :foo!

```
def foo  
begin  
return  
rescue  
else  
p :foo!  
end  
p :BOO # dummy sentence  
end
```

foo #=> nothing output, as OP expected

I'll commit the following patch unless there is objection.  
Let's discuss the better spec in another thread, if needed.

```
diff --git a/parse.y b/parse.y  
index 9769bb9..ffa649b 100644  
--- a/parse.y  
+++ b/parse.y  
@@ -8713,6 +8713,10 @@ reduce_nodes_gen(struct parser_params *parser, NODE **body)  
if (!subnodes(nd_head, nd_resq)) goto end;  
break;  
case NODE_RESCUE:  
  
  • if (node->nd_else) {  
  • body = &node->nd_resq;  
  • break;  
  • } if (!subnodes(nd_head, nd_resq)) goto end; break; default:
```

--  
Yusuke Endoh [mame@tsq.ne.jp](mailto:mame@tsq.ne.jp)

**#12 - 07/10/2011 11:42 AM - mame (Yusuke Endoh)**

- Status changed from Open to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r32483.  
Todd, thank you for reporting this issue.  
Your contribution to Ruby is greatly appreciated.  
May Ruby be with you.

- 
- parse.y (reduce\_nodes\_gen): NODE\_RETURN in rescue body must not be reduced when there is an else clause. This caused bizarre behavior in [Bug #4473] [ruby-core:35629] [ruby-core:37884].

## Files

---

ruby-1.9.2-return-else.rb

167 Bytes

03/06/2011

thuss (Todd Huss)