Ruby master - Bug #6378

irb incorrectly considers line incomplete when parenthesis are excluded

04/29/2012 03:40 PM - devans (Daniel Evans)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>keiju (Keiju Ishitsuka)</td>
</tr>
<tr>
<td>Target version:</td>
<td>ruby -v: ruby 1.9.3p125 (2012-02-16 revision 34643) [x86_64-darwin11.2.0]</td>
</tr>
<tr>
<td>Backport:</td>
<td></td>
</tr>
</tbody>
</table>

**Description**
Steps to reproduce:

start irb
attempt to execute the following line:

```
[1,2,3].reduce :+
```

**Expected:**
a result of => 6 should be printed and a new line should be available

**Actual:**
The result is not printed and the shell indicates more lines are expected

```
irb(main):001:0> [1,2,3].reduce :+
irb(main):002:0*
```

**Workaround:**
add parenthesis to the method call:

```
[1,2,3].reduce(++)
```

**Notes:**
The line is executed, as evidenced by changing the symbol to represent a message to which FixNum does not respond, such as :_
after which the error message will be printed immediately.

This is also reproducible in ruby 1.8.7 (2010-06-23 patchlevel 299) [x86_64-linux]

**Irb versions tested:**
- irb 0.9.6(09/06/30)
- irb 0.9.5(05/04/13)

**Associated revisions**

Revision 7b34c2f8 - 12/27/2012 02:48 PM - keiju (Keiju Ishitsuka)

- lib/irb/ruby-lex.rb: make lex_state to EXPR_END when next token is an operator after SYMBEG [Bug #6378].

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@38632 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 38632 - 12/27/2012 02:48 PM - keiju (Keiju Ishitsuka)

- lib/irb/ruby-lex.rb: make lex_state to EXPR_END when next token is an operator after SYMBEG [Bug #6378].

Revision 38632 - 12/27/2012 02:48 PM - keiju (Keiju Ishitsuka)

- lib/irb/ruby-lex.rb: make lex_state to EXPR_END when next token is an operator after SYMBEG [Bug #6378].

Revision 38632 - 12/27/2012 02:48 PM - keiju (Keiju Ishitsuka)

- lib/irb/ruby-lex.rb: make lex_state to EXPR_END when next token is an operator after SYMBEG [Bug #6378].
Revision 38632 - 12/27/2012 02:48 PM - keiju (Keiju Ishitsuka)

- lib/irb/ruby-lex.rb: make lex_state to EXPR_END when next token is an operator after SYMBEG [Bug #6378].

Revision 38632 - 12/27/2012 02:48 PM - keiju (Keiju Ishitsuka)

- lib/irb/ruby-lex.rb: make lex_state to EXPR_END when next token is an operator after SYMBEG [Bug #6378].

Revision 38632 - 12/27/2012 02:48 PM - keiju (Keiju Ishitsuka)

- lib/irb/ruby-lex.rb: make lex_state to EXPR_END when next token is an operator after SYMBEG [Bug #6378].

History

#1 - 05/03/2012 12:50 PM - mame (Yusuke Endoh)
- Status changed from Open to Assigned
- Assignee set to keiju (Keiju Ishitsuka)

Thanks, assigning it to keiju who is a developer/maintainer of irb.

--
Yusuke Endoh mame@tsg.ne.jp

#2 - 05/12/2012 07:01 AM - devans (Daniel Evans)

I noticed while working today that this only affects certain symbols, particularly :+ and :-

If you attempt the same code with :* it works perfectly.

```
irb(main):001:0> [1,2,3,4].reduce :-
irb(main):002:0> [1,2,3,4].reduce :+
irb(main):003:0> [1,2,3,4].reduce :*
=> 24
irb(main):004:0>
```

#3 - 12/27/2012 11:48 PM - keiju (Keiju Ishitsuka)

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

This issue was solved with changeset r38632.
Daniel, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

--

- lib/irb/ruby-lex.rb: make lex_state to EXPR_END when next token is an operator after SYMBEG [Bug #6378].