Array#delete(obj) should return obj when there is an object that is equal in the array

11/26/2012 01:00 PM - hasari (Hiro Asari)

<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>marcandre (Marc-Andre Lafortune)</td>
</tr>
<tr>
<td>Target version</td>
<td>2.0.0</td>
</tr>
<tr>
<td>ruby -v</td>
<td>ruby 2.0.0dev (2012-11-25 trunk 37839) [x86_64-darwin12.2.1]</td>
</tr>
</tbody>
</table>

### Description
According to [http://www.ruby-doc.org/core-1.9.3/Array.html#method-i-delete](http://www.ruby-doc.org/core-1.9.3/Array.html#method-i-delete), Array#delete(obj) should return "obj" when there are objects in the array that are "equal to obj" (internally, "==" is used, it seems).

Notice that the documentation does not state that the return value is an element of the array itself. However, 1.9.3 and trunk both return a member of the Array, rather than the argument.

This issue was raised in [https://github.com/jruby/jruby/issues/411](https://github.com/jruby/jruby/issues/411)

```ruby
#!/usr/bin/env ruby

class Foo
  attr_reader :name, :age

  def initialize name, age
    @name = name
    @age = age
  end

  def ==(other)
    other.name == name
  end

  foo1 = Foo.new "John Shahid", 27
  foo2 = Foo.new "John Shahid", 28
  array = [foo1]
  temp = array.delete foo2 #=> foo1, not foo2
```

### Associated revisions

- **Revision 37872 - 11/26/2012 08:14 PM - marcandre (Marc-Andre Lafortune)**
  - array.c: Fix rdoc for Array#delete [#7437]

- **Revision 37872 - 11/26/2012 08:14 PM - marcandre (Marc-Andre Lafortune)**
  - array.c: Fix rdoc for Array#delete [#7437]

- **Revision 37872 - 11/26/2012 08:14 PM - marcandre (Marc-Andre Lafortune)**
  - array.c: Fix rdoc for Array#delete [#7437]

- **Revision 37872 - 11/26/2012 08:14 PM - marcandre (Marc-Andre Lafortune)**
  - array.c: Fix rdoc for Array#delete [#7437]

- **Revision 37872 - 11/26/2012 08:14 PM - marcandre (Marc-Andre Lafortune)**
  - array.c: Fix rdoc for Array#delete [#7437]

- **Revision 37872 - 11/26/2012 08:14 PM - marcandre (Marc-Andre Lafortune)**
  - array.c: Fix rdoc for Array#delete [#7437]
array.c: Fix rdoc for Array#delete [#7437]

History

#1 - 11/26/2012 01:21 PM - hasari (Hiro Asari)
- File ruby-7437.patch added

Here's the patch. Where should the tests go? RubySpec?

#2 - 11/26/2012 03:23 PM - marcandre (Marc-Andre Lafortune)
- Category set to core
- Target version set to 2.0.0

Indeed, the documentation does not match the code and there is no test for this.

It was clearly Matz' intention to return the (last) deleted element: https://bugs.ruby-lang.org/projects/ruby-trunk/repository/revisions/18527

This is the most helpful behavior anyways.

I'll note that the documentation is clearly wrong in the second form too.

Unless there is objection, I'll commit the following patch:

diff --git a/array.c b/array.c
index df0a0a4..481eebc 100644
--- a/array.c
+++ b/array.c
@@ -2605,12 +2605,12 @@ rb_ary_keep_if(VALUE ary)
 /*
 callers: *
 o * ary.delete(obj)            -> obj or nil
+ o * ary.delete(obj) { block }  -> obj or nil
 o * ary.delete(obj)            -> element or nil
 o * ary.delete(obj) { block }  -> element or result of block *
 o * Deletes all items from +self+ that are equal to +obj+. *
 o * Returns the last deleted item, or +nil+ if no matching item is found. *
 o * If the optional code block is given, the result of the block is returned if *
 the item is not found. (To remove +nil+ elements and get an informative diff --git a/test/ruby/test_array.rb b/test/ruby/test_array.rb index 5e8d26d9..64668fc3 100644 -- a/test/ruby/test_array.rb b/test/ruby/test_array.rb @@ -598,14 +598,14 @@ class TestArray <
 Test::Unit::TestCase a = @cls['cab'..'cat'].to_a] assert_equal(99, a.delete('cup') { 99 } ] assert_equal(@cls['cab'..'cat'].to_a], a] +
 o o = Object.new
 o def o.==(other); true; end
 o o2 = Object.new
 o def o2.==(other); true; end
 o a = @cls[1, o, o2, 2]
 o assert_equal(o2, a.delete(42))
 o assert_equal([1, 2], a) end

def test_delete_at

#3 - 11/27/2012 12:20 AM - mame (Yusuke Endoh)
- Status changed from Open to Assigned
- Assignee set to marcandre (Marc-Andre Lafortune)

#4 - 11/27/2012 01:44 AM - headius (Charles Nutter)
marcandre (Marc-Andre Lafortune) wrote:

Indeed, the documentation does not match the code and there is no test for this.

It was clearly Matz' intention to return the (last) deleted element: https://bugs.ruby-lang.org/projects/ruby-trunk/repository/revisions/18527

This is the most helpful behavior anyways.

I'll note that the documentation is clearly wrong in the second form too.
Unless there is objection, I'll commit the following patch:

Yes, it appears to have been an explicit behavioral change in the 1.9.1/1.8.7 timeframe that never got reflected in documentation.

#5 - 11/27/2012 05:16 AM - marcandre (Marc-Andre Lafortune)
- Status changed from Assigned to Closed

Documentation fixed, thanks for bringing this up.

**Files**
ruby-7437.patch  522 Bytes  11/26/2012  hasari (Hiro Asari)