Ruby - Bug #2545

Array#delete_if is borked if user calls 'break'

01/02/2010 05:54 AM - candlerb (Brian Candler)

Status:	Closed		
Priority:	Normal		
Assignee:	matz (Yukihiro Matsumoto)		
Target version:			
ruby -v:	ruby 1.8.7 (2009-06-12 patchlevel 174) [x86_64-linux]	**Backport:**	

Description

=begin
Array is corrupted if you break out of a delete_if { ... } loop. I would expect that the elements already marked as deleted would be deleted, and the remainder of the array would be unchanged.

```
a = [5,6,7,8,9,10]
=> [5, 6, 7, 8, 9, 10]
a.delete_if { |x| break if x > 8; x < 7 }
=> nil
a
=> [7, 8, 7, 8, 9, 10]



RUBY_VERSION
=> "1.8.7"
RUBY_PATCHLEVEL
=> 174
=end
```

Related issues:

Related to Ruby - Bug #10722: Array#keep_if is borked if user calls 'break'	**Closed**	**01/09/2015**
Related to Ruby - Feature #10714: Array#reject! nonlinear performance problem	**Closed**	**01/08/2015**

Associated revisions

Revision 31e6b1e8 - 07/01/2011 10:17 PM - nobu (Nobuyoshi Nakada)

- array.c (rb_ary_reject_bang, rb_ary_delete_if): rejected
  elements should be removed. fixed [Bug #2545]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_8@32360 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision e7047b8a - 07/01/2011 10:17 PM - nobu (Nobuyoshi Nakada)

- array.c (rb_ary_reject_bang, rb_ary_delete_if): rejected
  elements should be removed. fixed [Bug #2545]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@32360 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision f0299205 - 07/03/2011 04:44 AM - nobu (Nobuyoshi Nakada)

- array.c (ary_reject_bang): should not remove elements which are
  not yielded. [Bug #2545]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_8@32373 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 6343e30c - 07/03/2011 04:44 AM - nobu (Nobuyoshi Nakada)

- array.c (ary_reject_bang): should not remove elements which are not yielded. [Bug #2545]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@32373 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 01/02/2010 12:29 PM - nobu (Nobuyoshi Nakada)

Hi,

At Sat, 2 Jan 2010 05:55:00 +0900,
Brian Candler wrote in [ruby-core:27366]:

> Array is corrupted if you break out of a delete_if { ... } loop. I would expect that the elements already marked as deleted would be deleted, and the remainder of the array would be unchanged.

The behavior would be an implementation detail, and should be undefined (or implementation defined), I guess.

```
Index: array.c
===================================================================
--- array.c (revision 26229)
+++ array.c (working copy)
@@ -2307,7 +2307,18 @@ rb_ary_reject_bang(VALUE ary)
     for (i1 = i2 = 0; i1 < RARRAY_LEN(ary); i1++) {
  VALUE v = RARRAY_PTR(ary)[i1];
- if (RTEST(rb_yield(v))) continue;
  if (i1 != i2) {
+     int state = 0;
+     if (RTEST(rb_protect(rb_yield, v, &state))) continue;
      rb_ary_store(ary, i2, v);
+     if (state) {
+  VALUE *ptr = RARRAY_PTR(ary);
+  long len = RARRAY_LEN(ary);
+  MEMCPY(ptr + i2 + 1, ptr + i1 + 1, VALUE, len - i1 - 1);
+  ARY_SET_LEN(ary, len - i1 + i2);
+  rb_jump_tag(state);
+     }
+ }
+ else {
+     if (RTEST(rb_yield(v))) continue;
  }
  i2++;
```

--
Nobu Nakada

#2 - 01/02/2010 06:45 PM - shyouhei (Shyouhei Urabe)

- Status changed from Open to Assigned

- Assignee set to matz (Yukihiro Matsumoto)

I don't think so. I'd also expect as the reporter did. Isn't it a bug?

Assigning to matz because this can be a design issue.

#3 - 06/30/2011 11:46 AM - daz (Dave B)

Just met this problem:
ruby 1.8.7 (2011-02-18 patchlevel 334) [i386-mingw32]

I'd also consider it a bug and that the ruby implementation should be hidden from the user. Once an element has been selected for deletion, at the end of this iteration, it should be expected to be gone. When using very large arrays, where the programmer knows of a shortcut (e.g. the rest of the array need not be considered), s/he should be encouraged to handle it with 'break'. In testing, I was left wondering whether 'delete_if' was non-destructive, because nothing had changed, and started looking for a bang! method.
To achieve the current behaviour, I only need an Array#dup above the loop.

Thanks to Nobu for working a patch.

daz

#4 - 07/02/2011 07:17 AM - nobu (Nobuyoshi Nakada)

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r32360.
Brian, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

- array.c (rb_ary_reject_bang, rb_ary_delete_if): rejected elements should be removed. fixed [Bug #2545]

#5 - 07/02/2011 07:18 AM - nobu (Nobuyoshi Nakada)

- Project changed from 2 to Ruby

- Category changed from core to core

- Target version deleted (Ruby 1.8.7)

#6 - 07/02/2011 08:15 PM - nagachika (Tomoyuki Chikanaga)

Hi,

According to test added by r32360,

```
a = [ 5, 6, 7, 8, 9, 10 ]
a.delete_if {|i| break i if i > 8; i < 7}
```

it results
a # => [7, 8]

But I feel it could be [7, 8, 9, 10] because block didn't *return true* for 9, 10.
Matz, How do you think about it?

#7 - 01/09/2015 08:53 AM - wanabe (_ wanabe)

- Related to Bug #10722: Array#keep_if is borked if user calls 'break' added

#8 - 01/09/2015 11:30 PM - nobu (Nobuyoshi Nakada)

- Related to Feature #10714: Array#reject! nonlinear performance problem added