

Ruby master - Bug #5127

ruby 1.9.3 prev1 Array#[]= (Array.new) 1.9.2 2

08/01/2011 09:42 AM - Anonymous

Status: Closed	
Priority: Normal	
Assignee: authorNari (Narihiro Nakamura)	
Target version: 1.9.3	
ruby -v: -	Backport:
Description	
Array Array#[]= ruby 1.9.3 prev1 1.9.2 p290 (require 'benchmark' Benchmark.bm(22) do x x.report "Array.new" do 1_000_000.times do Array.new(100) end end x.report "Array#[]= with Array" do a = Array.new(1_000_000) 1_000_000.times do i a[i] = Array.new(100) end end end \$ ruby19 -v bm_array.rb ruby 1.9.2p290 (2011-07-09 revision 32553) [x86_64-darwin11.0.0] user system total real Array.new 0.600000 0.140000 0.740000 (0.740639) Array#[]= with Array 8.590000 0.300000 8.890000 (8.874507) \$ ruby -v bm_array.rb ruby 1.9.3dev (2011-07-31 revision 32789) [x86_64-darwin11.0.0] user system total real Array.new 0.670000 0.160000 0.830000 (0.826589) Array#[]= with Array 16.160000 0.280000 16.440000 (16.597618) • OS : Mac OS X 10.7 • CPU : Intel Core i5 • Memory : 4GB • compiler: gcc 4.2.1 	

Associated revisions

Revision e26cead1 - 08/09/2011 04:26 AM - nari

- gc.c (gc_lazy_sweep): if sweep target slots are not found, we try heap_increment() because it might be able to expand the heap. [Bug #5127] [ruby-dev:44285]
- gc.c (gc_clear_mark_on_sweep_slots): if a sweeping was interrupted, we expand the heap if at all possible.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@32894 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 32894 - 08/09/2011 04:26 AM - nari

- gc.c (gc_lazy_sweep): if sweep target slots are not found, we try heap_increment() because it might be able to expand the heap. [Bug #5127] [ruby-dev:44285]
- gc.c (gc_clear_mark_on_sweep_slots): if a sweeping was interrupted, we expand the heap if at all possible.

Revision 32894 - 08/09/2011 04:26 AM - nari

- gc.c (gc_lazy_sweep): if sweep target slots are not found, we try heap_increment() because it might be able to expand the heap. [Bug #5127] [ruby-dev:44285]
- gc.c (gc_clear_mark_on_sweep_slots): if a sweeping was interrupted, we expand the heap if at all possible.

Revision 32894 - 08/09/2011 04:26 AM - nari

- gc.c (gc_lazy_sweep): if sweep target slots are not found, we try heap_increment() because it might be able to expand the heap. [Bug #5127] [ruby-dev:44285]
- gc.c (gc_clear_mark_on_sweep_slots): if a sweeping was interrupted, we expand the heap if at all possible.

Revision 32894 - 08/09/2011 04:26 AM - nari

- gc.c (gc_lazy_sweep): if sweep target slots are not found, we try heap_increment() because it might be able to expand the heap. [Bug #5127] [ruby-dev:44285]
- gc.c (gc_clear_mark_on_sweep_slots): if a sweeping was interrupted, we expand the heap if at all possible.

Revision 32894 - 08/09/2011 04:26 AM - nari

- gc.c (gc_lazy_sweep): if sweep target slots are not found, we try heap_increment() because it might be able to expand the heap. [Bug #5127] [ruby-dev:44285]


```
x.report "Array#[]= with Array" do
a = Array.new(1_000_000)
1_000_000.times do |i|
a[i] = Array.new(100)
end
end
puts "objects count = #{ ObjectSpace.each_object{} }"
end
```

```
$ ruby-1.9.2 bm_array.rb
user system total real
Array.new 1.470000 1.220000 2.690000 ( 2.705700)
objects count = 1007204
Array#[]= with Array 1.910000 0.030000 1.940000 ( 2.055783)
objects count = 1003992
```

```
$ ruby-trunk bm_array.rb
user system total real
Array.new 1.510000 1.230000 2.740000 ( 2.753933)
objects count = 1005182
Array#[]= with Array 1.940000 0.010000 1.950000 ( 1.977659)
objects count = 1002551
```

#3 - 08/08/2011 05:11 PM - authorNari (Narihiro Nakamura)

- Assignee set to authorNari (Narihiro Nakamura)

GC

#4 - 08/09/2011 01:26 PM - authorNari (Narihiro Nakamura)

- Status changed from Open to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r32894.
, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

-
- gc.c (gc_lazy_sweep): if sweep target slots are not found, we try heap_increment() because it might be able to expand the heap. [Bug [#5127](#)] [ruby-dev:44285]
 - gc.c (gc_clear_mark_on_sweep_slots): if a sweeping was interrupted, we expand the heap if at all possible.