Bug #7765
Proc#arity bug with optional argument
02/01/2013 05:38 AM - marcandre (Marc-Andre Lafortune)

Status: Closed
Priority: Normal
Assignee: marcandre (Marc-Andre Lafortune)
Target version: 2.0.0
ruby -v: 38984

Description
We have:
Proc.new{|a, b, c, d, e|}.arity #=> 5

Matz decided [ruby-core:46515] we also have:
Proc.new{|a, b, c, d, e=0|}.arity #=> 4

It looks like we currently have:
Proc.new{|a, b=0, c, d, e|}.arity #=> 1, should be same as above

Hopefully there won't be disagreement that this is a bug?

I'm asking in particular because there was a specific test committed by Yui to that effect. I hope this was just an oversight?

assert_equal(0, proc{|x=0, y|}.arity) # Should be 1, not 0. test/ruby/test_proc.rb:67

My patch is ready and I will commit it unless there is objection.

https://github.com/marcandre/ruby/compare/marcandre:trunk...marcandre:proc_curry

Associated revisions
Revision 83610815 - 02/01/2013 10:46 PM - marcandre (Marc-Andre Lafortune)

- proc.c: Add {*}_min_max_arity and refactor.
  [Bug #7765]
- test/ruby/test_proc.rb: Fix wrong test

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@39007 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 39007 - 02/01/2013 10:46 PM - marcandre (Marc-Andre Lafortune)

- proc.c: Add {*}_min_max_arity and refactor.
  [Bug #7765]
- test/ruby/test_proc.rb: Fix wrong test

Revision 39007 - 02/01/2013 10:46 PM - marcandre (Marc-Andre Lafortune)

- proc.c: Add {*}_min_max_arity and refactor.
  [Bug #7765]