

Ruby trunk - Bug #7837

IO.open with three arguments where third argument is bogus reports an arity problem instead of type problem

02/13/2013 03:44 AM - enebo (Thomas Enebo)

Status: Rejected	
Priority: Normal	
Assignee:	
Target version: 2.6	
ruby -v: ruby 2.0.0dev (2013-02-08 trunk 39161) [x86_64-darwin11.4.2]	Backport: 2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN
Description I am reporting this against 2.0.rc2 but it also applies to mri 1.9.3. If I run the following one-liner: <code>mri20 -e 'f = File.open("tmp"); IO.open(f.fileno, "r", :heh)'</code> I get the error: <code>-e:1:in `initialize': wrong number of arguments (3 for 1..2) (ArgumentError)</code> This seems wrong because if I change the third argument to an options hash it works fine. So having an arity of 3 is not a problem for open. It is that :heh is not a hash. I think this should generate a TypeError.	

History

#1 - 02/13/2013 05:14 AM - drbrain (Eric Hodel)

- Category set to core

- Target version set to 2.6

```
=begin
Since Ruby 1.9.3p374 has the same behavior I have marked this for next minor:

$ ruby19 -ve 'f = File.open("tmp"); IO.open(f.fileno, "r", :heh)'
ruby 1.9.3p374 (2013-01-15 revision 38858) [x86_64-darwin12.2.1]
-e:1:in initialize: wrong number of arguments (3 for 1..2) (ArgumentError)
from -e:1:in open'
from -e:1:in `

=end
```

#2 - 02/13/2013 02:39 PM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Rejected

```
=begin
It's common behavior to keyword arguments,

$ ruby -e 'def foo(a, b, c:1);end' -e 'foo(2,3,42)'
-e:1:in foo: wrong number of arguments (3 for 2) (ArgumentError)
from -e:2:in '
```

The optional argument hash is stripped before arity check, so it's "invisible" as arity.
=end