Ruby master - Bug #11068
unable to omit an optional keyarg if the previous arg is an optional hash
04/14/2015 06:31 PM - bughit (bug hit)

<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></td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>ruby -v:</td>
<td>ruby 2.2.2p95 (2015-04-13 revision 50295) [x86_64-linux]</td>
</tr>
<tr>
<td>Backport:</td>
<td>2.0.0: UNKNOWN, 2.1: UNKNOWN, 2.2: UNKNOWN</td>
</tr>
</tbody>
</table>

**Description**

```ruby
irb(main):001:0> def foo(hash = {}, a: nil); [hash, a] end
=> :foo
irb(main):002:0> foo({b: 1}, a: 1)
=> [{:b=>1}, 1]
irb(main):003:0> foo({b: 1})
ArgumentError: unknown keyword: b
  from (irb):1 in `foo'
  from (irb):3
from /home/alex/.rbenv/versions/2.2.2/bin/irb:11 in `<main>,'
irb(main):004:0>
```

**Related issues:**

- Related to Ruby master - Feature #14183: “Real” keyword argument added

**History**

**#1 - 07/06/2019 12:00 AM - jeremyevans0 (Jeremy Evans)**
- Related to Feature #14183: “Real” keyword argument added

**#2 - 09/02/2019 04:24 AM - jeremyevans0 (Jeremy Evans)**
- Status changed from Open to Closed

With the acceptance of #14183, this code now emits warnings as the behavior will change in Ruby 3:

```ruby
foo({b: 1})
# (irb):53: warning: The last argument is used as the keyword parameter
# (irb):51: warning: for `foo' defined here
# ArgumentError (unknown keyword: :b)
```

In Ruby 3, you will not get a warning, and the code will work as expected. To get the Ruby 3 behavior currently, you can do:

```ruby
foo({b: 1}, **({}))
# => [{:b=>1}, nil]
```