# Ruby - Bug #17719

## Irregular evaluation order in hash literals

03/12/2021 01:05 PM - nobu (Nobuyoshi Nakada)

| | | | |
|---|---|---|---|
| **Status:** | Closed | | |
| **Priority:** | Normal | | |
| **Assignee:** | | | |
| **Target version:** | | | |
| **ruby -v:** | | **Backport:** | 2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: UNKNOWN |

**Description**

@mame (Yusuke Endoh) pointed out an irregular evaluation order example.

```
$ ruby -e '{foo:p(1), bar:p(2), foo:p(3)}'
-e:1: warning: key :foo is duplicated and overwritten on line 1
1
3
2
```

It feels like a bug.
https://github.com/nobu/ruby/tree/duplicated-keys-order

Or, probably it would be better to turn into an error?

---

**Associated revisions**

**Revision fac2c0f73cafb5d65bfbba7aa8018fa427972d71 - 10/18/2021 04:09 PM - jeremyevans (Jeremy Evans)**

Fix evaluation order of hash values for duplicate keys

Fixes [Bug #17719]

Co-authored-by: Nobuyoshi Nakada nobu@ruby-lang.org
Co-authored-by: Ivo Anjo ivo@ivoanjo.me

**Revision fac2c0f73cafb5d65bfbba7aa8018fa427972d71 - 10/18/2021 04:09 PM - jeremyevans (Jeremy Evans)**

Fix evaluation order of hash values for duplicate keys

Fixes [Bug #17719]

Co-authored-by: Nobuyoshi Nakada nobu@ruby-lang.org
Co-authored-by: Ivo Anjo ivo@ivoanjo.me

**Revision fac2c0f7 - 10/18/2021 04:09 PM - jeremyevans (Jeremy Evans)**

Fix evaluation order of hash values for duplicate keys

Fixes [Bug #17719]

Co-authored-by: Nobuyoshi Nakada nobu@ruby-lang.org
Co-authored-by: Ivo Anjo ivo@ivoanjo.me

---

**History**

**#1 - 03/12/2021 08:40 PM - xtkoba (Tee KOBAYASHI)**

To me the evaluation order of 1->3->2 is a bit unexpected, but it might be OK unless specified otherwise.

IMO, duplicate keys in hash literals should be allowed in a scripting language like Ruby. For example, they are explicitly allowed in another scripting language named Python [1]. On the other hand, compiled languages should disallow them.

[1] https://mail.python.org/pipermail/python-ideas/2019-March/055726.html

**#2 - 07/15/2021 06:22 AM - nobu (Nobuyoshi Nakada)**

I think it can be fixed by removing the optimization in the parser and leaving it to the compiler.

**#3 - 10/14/2021 10:55 PM - jeremyevans0 (Jeremy Evans)**

nobu (Nobuyoshi Nakada) wrote in #note-2:

> I think it can be fixed by removing the optimization in the parser and leaving it to the compiler.

I tried the approach of removing the optimization from the parser completely (
https://github.com/jeremyevans/ruby/commit/bead10831e4f7ea6fc517fe66796f12579915fb3), but it doesn't handle duplicate keys in keyword
arguments, resulting in incorrect unknown keyword errors: https://github.com/jeremyevans/ruby/runs/3900240357#step:15:168

I also tried rebasing your duplicated-keys-order branch against master, and that appears to fix the issue. I submitted a pull request for that:
https://github.com/ruby/ruby/pull/4969

**#4 - 10/18/2021 04:09 PM - jeremyevans (Jeremy Evans)**

- *Status changed from Open to Closed*

Applied in changeset git|fac2c0f73cafb5d65bfbba7aa8018fa427972d71.

---

Fix evaluation order of hash values for duplicate keys

Fixes [Bug #17719]

Co-authored-by: Nobuyoshi Nakada nobu@ruby-lang.org
Co-authored-by: Ivo Anjo ivo@ivoanjo.me