Ruby master - Bug #18518

NoMemoryError + [FATAL] failed to allocate memory for twice 1 << large

01/26/2022 03:30 PM - Eregon (Benoit Daloze)

Status: Open
Priority: Normal
Assignee: 
Target version:
Backport: 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: UNKNOWN, 3.1: UNKNOWN

Description
Repro:

```ruby
exp = 2**40 # also fails with bignum e.g. 2**64

def exc
  begin
    yield
    rescue NoMemoryError => e
      p :NoMemoryError
    end
  end

  p exp
  exc { (1 << exp) }
  exc { (-1 << exp) }
  exc { (bignum_value << exp) }
  exc { (-bignum_value << exp) }
```

Output:

```
$ ruby -v mri_oom.rb
ruby 3.0.2p107 (2021-07-07 revision 0db68f0233) [x86_64-linux]
mri_oom.rb:7: warning: assigned but unused variable - e
1099511627776
:NoMemoryError
[FATAL] failed to allocate memory

3.1.0 seems fine:

$ ruby -v mri_oom.rb
ruby 3.1.0p0 (2021-12-25 revision fb4df44d16) [x86_64-linux]
3.1.0p0.rb:7: warning: assigned but unused variable - e
1099511627776
:NoMemoryError
:NoMemoryError
:NoMemoryError
:NoMemoryError
```

Related issues:
Related to Ruby master - Bug #18517: 0 << (2**40) is NoMemoryError but 0 << (2**80) is 0 added

History

#1 - 01/26/2022 03:31 PM - Eregon (Benoit Daloze)
- Related to Bug #18517: 0 << (2**40) is NoMemoryError but 0 << (2**80) is 0 added

#2 - 01/28/2022 01:29 PM - Eregon (Benoit Daloze)
Actually I'm not sure this is properly fixed on 3.1.0, it looks brittle, for instance it fails in GitHub Actions on macOS: 
https://github.com/ruby/spec/runs/4981061444?check_suite_focus=true

NoMemoryError might also be a bit weird for this.
How about raising `RangeError` or `TypeError` for any exponent which does not fit in a 32-bit signed int?
Such cases don't make much sense anyway.

#3 - 01/28/2022 01:40 PM - Eregon (Benoit Daloze)
From this log it's clear this issue happens on 3.1.0 macOS:
https://github.com/eregon/rubyspec/runs/4981235909?check_suite_focus=true

```
2022-01-28T13:48:1570860Z Integer#<< (with n << m) when m is a bignum or larger than int
2022-01-28T13:48:1659830Z - returns -1 when m < 0 and n < 0
2022-01-28T13:48:1755940Z - returns 0 when m < 0 and n >= 0
2022-01-28T13:48:1856250Z - returns 0 when m > 0 bignum and n == 0
2022-01-28T13:40:01.6597290Z - raises NoMemoryError when m > 0 and n != 0
```