Ruby master - Bug #17529

Ractor Segfaults with GC enabled
01/12/2021 01:06 AM - prajjwal (Prajwal Singh)

Status: Assigned
Priority: Normal
Assignee: ko1 (Koichi Sasada)
Target version:
ruby -v: ruby 3.0.0p0 (2020-12-25 revision 95aff21468) [x86_64-linux]
Backport: 2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: UNKNOWN

Description
I've been benchmarking Ractor on my machine with the following naive prime number generator:

```ruby
# frozen_string_literal: true

def prime?(n)
  2.upto(n - 1).none? { |i| n % i == 0 }
end

NUM_WORKERS = ARGV[0].to_i

producer = Ractor.new do
  i = 1000000

  loop { Ractor.yield i; i += 1 }
end

workers = (1..NUM_WORKERS).map do
  Ractor.new producer do |producer|
    while n = producer.take
      Ractor.yield [n, prime?(n)]
    end
  end
end

loop do
  _r, { number, prime } = Ractor.select(*workers)
  p number if prime
end
```

The code inevitably segfaults, and it appears to be the garbage collector.

If I stick GC.disable in there, the code happily chugs along for several minutes on end without a problem.

Related issues:
Related to Ruby master - Bug #17489: Ractor segfaults

History
#1 - 01/12/2021 01:12 AM - prajjwal (Prajwal Singh)
- File ractor.crash added

#2 - 01/12/2021 01:42 AM - marcandre (Marc-Andre Lafortune)
- Related to Bug #17489: Ractor segfaults added

#3 - 01/12/2021 01:42 AM - marcandre (Marc-Andre Lafortune)
Thanks for the report.

Probably the same bug as #17489
I couldn't reproduce it. Could you tell me ARGV[0]?

BTW please fill "ruby -v:" filed with your environment (even if it is in crash log)

rub 3.0.0p0 (2020-12-25 revision 95aff21468) [x86_64-linux]

It crashes for any value of ARGV[0] between 1 and 25 (that I tested).

The fact that its happening so consistently for me and not for you makes me wonder if the problem stems from my version of Linux or GCC? Some other compile time option perhaps?

Here's my GCC version:

Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/lib/gcc/x86_64-pc-linux-gnu/10.2.0/lto-wrapper
Target: x86_64-pc-linux-gnu
Thread model: posix
Supported LTO compression algorithms: zlib zstd

gcc version 10.2.0 (GCC)

And Linux:

Linux Wraith 5.9.14-arch1-1 #1 SMP PREEMPT Sat, 12 Dec 2020 14:37:12 +0000 x86_64 GNU/Linux

Ruby Configure Args

'--prefix=/home/prajjwal/.rbenv/versions/3.0.0' '--enable-shared' 'LDFLAGS=-L/home/prajjwal/.rbenv/versions/3.0.0/lib ' 'CPPFLAGS=-I/home/prajjwal/.rbenv/versions/3.0.0/include '

Just confirmed that it only segfaults when ruby is configured with the --enable-shared option (which rbenv does by default).

Even more info:

glibc 2.32-5

hmm I can't reproduce it yet. Can someone try it and get more information about it?

Files

raactor.crash 22.5 KB 01/12/2021 prajjwal (Prajwal Singh)