

Ruby - Feature #10329

[PATCH 2/1] optimize: foo == "literal string"

10/06/2014 02:31 AM - normalperson (Eric Wong)

Status: Rejected Priority: Normal Assignee: Target version: 2.2.0	
Description Follow-up-to: [Feature #10326] (optimize: recv << "literal string") This speeds up some string comparisons greatly, vm2_streq1 results: [[["loop_whileloop2", [[0.21688890410587192, 0.21697380719706416, 0.21685226494446397, 0.21750432904809713], [0.21682232478633523, 0.2153299949131906, 0.21504606027156115, 0.21595032326877117]]], ["vm2_streq1", [[0.8923470871523023, 0.9007423971779644, 0.8952413029037416, 0.9235154730267823], [0.42798056406900287, 0.3970582918263972, 0.4244600120000541, 0.4243976562283933]]]]] Elapsed time: 7.020550268 (sec) benchmark results: minimum results in each 4 measurements. Execution time (sec) name a b loop_whileloop2 0.217 0.215 vm2_streq1* 0.675 0.182 Speedup ratio: compare with the result of `a` (greater is better) name b loop_whileloop2 1.008 vm2_streq1* 3.711 I have not figured out the reverse (yoda condition), yet: "literal string" == foo benchmark/bm_vm2_streq1.rb 6 ++++++ compile.c 23 ++++++ insns.def 21 ++++++ test/ruby/test_string.rb 29 ++++++ 4 files changed, 79 insertions(+) create mode 100644 benchmark/bm_vm2_streq1.rb	
Related issues: Related to Ruby - Feature #10326: [PATCH] optimize: recv << "literal string" Rejected10/05/2014	

History

#1 - 10/06/2014 02:32 AM - normalperson (Eric Wong)

- Related to Feature #10326: [PATCH] optimize: recv << "literal string" added

#2 - 10/24/2014 03:28 AM - normalperson (Eric Wong)

- Status changed from Open to Rejected

obsoleted by Feature [#10423](#)

Files

0002-optimize-foo-literal-string.patch	4.36 KB	10/06/2014	normalperson (Eric Wong)
--	---------	------------	--------------------------