

Ruby - Feature #17115

Optimize String#casecmp? for ASCII strings

08/11/2020 03:16 PM - byroot (Jean Boussier)

Status:Open

Priority:Normal

Assignee:

Target version:

Description

Patch: <https://github.com/ruby/ruby/pull/3369>

casecmp? is a kind of performance trap as it's much slower than using a case insensitive regexp or just casecmp == 0.

```
str = "Connection"
cmp = "connection"
Benchmark.ips do |x|
  x.report('/\A\z/i.match?') { /\Afoo\Z/i.match?(str) }
  x.report('casecmp?') { cmp.casecmp?(str) }
  x.report('casecmp') { cmp.casecmp(str) == 0 }
  x.compare!
end
```

Calculating -----

/\A\z/i.match?	11.447M	(± 1.3%)	i/s -	57.814M	in	5.051489s
casecmp?	6.197M	(± 0.9%)	i/s -	31.138M	in	5.025252s
casecmp	12.753M	(± 1.2%)	i/s -	64.636M	in	5.069195s

Comparison:

casecmp:	12752791.6	i/s				
/\A\z/i.match?:	11446996.1	i/s -	1.11x	(± 0.00)	slower	
casecmp?:	6196886.0	i/s -	2.06x	(± 0.00)	slower	

This is because, unlike the others, it is sensitive to unicode case folding.

However, there are cases where fast case insensitive equality check of known ASCII strings is useful. For instance, matching HTTP headers.

This patch checks if both strings use a single byte encoding, and if so then does a simple iterative comparison with TOLOWER(). This makes casecmp? slightly faster than casecmp == 0 when both strings are ASCII.

	compare-ruby	built-ruby
:-----	-----:	-----:
casecmp-1	11.618M	10.757M
	1.08x	-
casecmp-10	1.849M	1.723M
	1.07x	-
casecmp-100	204.490k	186.798k
	1.09x	-
casecmp-1000	20.413k	20.184k
	1.01x	-
casecmp-nonasci1	19.541M	20.100M
	-	1.03x
casecmp-nonasci10	19.489M	19.914M
	-	1.02x
casecmp-nonasci100	19.479M	20.155M
	-	1.03x
casecmp-nonasci1000	19.462M	20.064M
	-	1.03x
casecmp_p-1	2.214M	12.030M
	-	5.43x
casecmp_p-10	1.373M	2.150M
	-	1.57x
casecmp_p-100	249.292k	231.041k
	1.08x	-

casecmp_p-1000	16.173k	23.592k
	-	1.46x
casecmp_p-nonasciil	651.921k	650.572k
	1.00x	-
casecmp_p-nonasciil10	108.253k	109.006k
	-	1.01x
casecmp_p-nonasciil100	11.749k	11.889k
	-	1.01x
casecmp_p-nonasciil1000	1.140k	1.138k

History

#1 - 08/11/2020 03:17 PM - byroot (Jean Boussier)

- Description updated

#2 - 08/14/2020 03:14 PM - Dan0042 (Daniel DeLorme)

In the benchmark you'd need to change the regexp from `^Afoo\Z/i` to `^Aconnection\z/i`; if you do so you'll find the regexp performance is similar to casecmp?

+1 for special-casing ASCII strings though.

Related: [#13750](#), [#14055](#)

#3 - 08/15/2020 03:48 AM - sawa (Tsuyoshi Sawada)

- Description updated