Ruby master - Bug #13228
s[i]=c(assigning a character) for String is slower than Array on Linux
02/18/2017 02:37 PM - yoshiokatsuneo (Tsuneo Yoshioka)

Status: Rejected
Priority: Normal
Assignee:
Target version:
ruby -v: ruby 2.4.0p0 (2016-12-24 revision 57164) [x86_64-linux]
Backport: 2.2: UNKNOWN, 2.3: UNKNOWN, 2.4: UNKNOWN

Description
s[i]=c(assigning a character) for String is slower than for Array on Linux.

If I split the String to Array, and assign characters, and join the Array to String,
then it is much faster than assigning characters directly to the string.

Somehow, I don't see the performance difference on Mac OS X.

~$ time ruby -e 'N=100000; s="a"*N; N.times{s[Random.rand(N)]="Z"}; puts s' >/dev/null
real 0m0.879s
user 0m0.836s
sys 0m0.012s

~$ time ruby -e 'N=100000; s="a"*N; s=s.split(""); N.times{s[Random.rand(N)]="Z"}; puts s.join("")' >/dev/null
real 0m0.153s
user 0m0.108s
sys 0m0.016s

~$ uname -a
Linux aaaaaaaaa 4.4.0-43-generic #63-Ubuntu SMP Wed Oct 12 13:48:03 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux

Associated revisions
Revision cc68af3d - 04/17/2017 01:38 PM - nobu (Nobuyoshi Nakada)
string.c: improve insertion performace
  string.c (rb_str_splice_0); improve performace of single byte optimizable cases, insertion 7bit string to 7bit string. [ruby-dev:49984] [Bug #13228]
git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@58383 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 58383 - 04/17/2017 01:38 PM - nobu (Nobuyoshi Nakada)
string.c: improve insertion performace
  string.c (rb_str_splice_0); improve performace of single byte optimizable cases, insertion 7bit string to 7bit string. [ruby-dev:49984] [Bug #13228]

Revision 58383 - 04/17/2017 01:38 PM - nobu (Nobuyoshi Nakada)
string.c: improve insertion performace
  string.c (rb_str_splice_0); improve performace of single byte optimizable cases, insertion 7bit string to 7bit string. [ruby-dev:49984] [Bug #13228]
**Revision 58383 - 04/17/2017 01:38 PM - nobu (Nobuyoshi Nakada)**

string.c: improve insertion performace

- string.c (rb_str_splice_0): improve performace of single byte optimizable cases, insertion 7bit string to 7bit string. [ruby-dev:49984] [Bug #13228]

**History**

**#1 - 02/19/2017 01:12 AM - wanabe (_wanabe)**

- ruby -v set to ruby 2.4.0p0 (2016-12-24 revision 57164) [x86_64-linux]

perf shows that ruby spent most of the time in search_nonascii().

```bash
$ perf record ruby -ve 'n=100000; s = "a" * n; t = Time.now; n.times do |i| s[i] = "z"; end; p Time.now - t'
ruby 2.5.0dev (2017-02-18 trunk 57652) [x86_64-linux]
5.271689721
[ perf record: Woken up 4 times to write data ]
[ perf record: Captured and wrote 0.858 MB perf.data (21558 samples) ]
```

$ perf report -n --stdio|head -20

# To display the perf.data header info, please use --header/--header-only options.

### Total Lost Samples: 0
### Samples: 21K of event 'cycles'
### Event count (approx.): 15739606081

<table>
<thead>
<tr>
<th>Overhead</th>
<th>Samples</th>
<th>Command</th>
<th>Shared Object</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>96.14%</td>
<td>20654</td>
<td>ruby</td>
<td>ruby</td>
<td>search_nonascii</td>
</tr>
<tr>
<td>0.18%</td>
<td>45</td>
<td>ruby</td>
<td>ruby</td>
<td>ruby_yyparse</td>
</tr>
<tr>
<td>0.18%</td>
<td>38</td>
<td>ruby</td>
<td>[wl]</td>
<td>osl_readl</td>
</tr>
<tr>
<td>0.17%</td>
<td>38</td>
<td>ruby</td>
<td>ruby</td>
<td>vm_exec_core</td>
</tr>
<tr>
<td>0.13%</td>
<td>27</td>
<td>ruby</td>
<td>[kernel.kallsyms]</td>
<td>delay_tsc</td>
</tr>
<tr>
<td>0.07%</td>
<td>16</td>
<td>ruby</td>
<td>ruby</td>
<td>rb_str_splice_0</td>
</tr>
<tr>
<td>0.07%</td>
<td>15</td>
<td>ruby</td>
<td>ruby</td>
<td>gc_page_sweep</td>
</tr>
<tr>
<td>0.06%</td>
<td>15</td>
<td>ruby</td>
<td>ruby</td>
<td>rb_enc_from_index</td>
</tr>
<tr>
<td>0.06%</td>
<td>13</td>
<td>ruby</td>
<td>ruby</td>
<td>rb_str_update</td>
</tr>
</tbody>
</table>

I wonder the script uses only ASCII characters, and we have RUBY_ENC_CODERANGE_7BIT. But rb_str_splice_0() calls rb_str_modify() and clear code-range information by ENC_CODERANGE_CLEAR().

**#2 - 04/17/2017 09:36 AM - sorah (Sorah Fukumori)**

Difference of locale configuration, not OS?

sorah@yuuki - $ uname -a
Linux yuuki 4.9.6-gentoo-r1 SMP Sun Feb 12 01:20:31 UTC 2017 x86_64 Intel(R) Celeron(R) CPU N3050 @ 1.60GHz GenuineIntel GNU/Linux

sorah@yuuki - $ time env LANG=C ruby -e 'N=100000; s="a"*N; N.times{s[Random.rand(N)]="Z"}; puts s' >/dev/null
real   0m0.387s
user   0m0.229s
sys    0m0.085s

sorah@yuuki - $ time env LANG=en_US.UTF-8 ruby -e 'N=100000; s="a"*N; N.times{s[Random.rand(N)]="Z"}; puts s' >/dev/null
real   0m3.015s
user   0m2.919s
sys    0m0.079s

**#3 - 04/17/2017 09:39 AM - naruse (Yui NARUSE)**

- Status changed from Open to Rejected

It's natural because String index access requires character counting.
If you need performance and the string is ASCII or Binary, you can set encoding of the string by String#force_encoding. Then ruby can use direct index access.

Maybe your Mac's locale is LANG=C and strings are handled as single byte encoding.
You can confirm this by Encoding.locale_charmap.

#4 - 04/17/2017 01:33 PM - nobu (Nobuyoshi Nakada)
- Description updated