Ruby master - Bug #16145

regexp match error if mixing /i, character classes, and utf8

09/05/2019 07:40 PM - zenspider (Ryan Davis)

Status:	Open		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:		**Backport:**	2.5: UNKNOWN, 2.6: UNKNOWN

Description

(reported on behalf of mage@mage.gold -- there appears to be an error in registration or login):

See: ruby-talk @ X-Mail-Count: 440336

```
2.6.3 :049 > 'SHOP' =~ /[xo]/i
=> 2
2.6.3 :050 > 'CAFÉ' =~ /[é]/i
=> 3
2.6.3 :051 > 'CAFÉ' =~ /[xé]/i
=> nil
2.6.3 :052 > 'CAFÉ' =~ /[xÉ]/i
=> 3
```

Expected result:

```
2.6.3 :051 > 'CAFÉ' =~ /[xé]/i
=> 3
```

I tested it on random regex online pages.

It does not match on https://regex101.com/

It matches on:

https://regexr.com/
https://www.regextester.com/
https://www.freeformatter.com/regex-tester.html

(Ignore case turned on).

The reason I suppose it's more like a bug than a feature is the fact that /[é]/i matches 'CAFÉ'. If the //i didn't work for UTF-8 characters then the /[é]/i wouldn't match it either. For example, [é] does not match 'CAFÉ' on https://regex101.com/

I could not find a page or a system that behaves the same way as Ruby does. For example, it matches in PostgreSQL 10 (under FreeBSD 12) too:

```
# select 'CAFÉ'~ '[xé]';

## ?column?

f
(1 row)

# select 'CAFÉ' ~* '[xé]';

## ?column?

t
(1 row)
```

Tested it in IRB on macOS and FreeBSD.

```
$ uname -a && ruby -v && locale
Darwin xxx 18.7.0 Darwin Kernel Version 18.7.0: Thu Jun 20 18:42:21 PDT 2019; root:xnu-4903.270.47~4/RELEASE_X86_64 x86_64
ruby 2.6.3p62 (2019-04-16 revision 67580) [x86_64-darwin18]
LANG="en_US.UTF-8"
LC_COLLATE="en_US.UTF-8"
LC_CTYPE="en_US.UTF-8"
LC_MESSAGES="en_US.UTF-8"
LC_MONETARY="en_US.UTF-8"
LC_NUMERIC="en_US.UTF-8"
LC_TIME="en_US.UTF-8"
LC_ALL="en_US.UTF-8"

$ uname -a && ruby -v && locale
FreeBSD xxx 12.0-RELEASE-p9 FreeBSD 12.0-RELEASE-p9 GENERIC  amd64
ruby 2.6.3p62 (2019-04-16 revision 67580) [x86_64-freebsd12.0]
LANG=en_US.UTF-8
LC_CTYPE="en_US.UTF-8"
LC_COLLATE="en_US.UTF-8"
LC_TIME="en_US.UTF-8"
LC_NUMERIC="en_US.UTF-8"
LC_MONETARY="en_US.UTF-8"
LC_MESSAGES="en_US.UTF-8"
LC_ALL=en_US.UTF-8
```

I installed Ruby with RVM.

History

#1 - 09/06/2019 05:52 AM - duerst (Martin Dürst)

Definitely a bug. Confirmed on master (ruby -v
ruby 2.7.0dev (2019-07-06T03:43:38Z trunk f296c260ef) [x86_64-cygwin])

"CAFÉ" =~ /x|é/i
works. So that may be an alternative until this is fixed. It may also give some hints on where the bug comes from. My current guess is that single-character character classes get reduced to just the actual character, so that's why they work.