Ruby - Bug #8255

File#each_line omits last byte (==\0) if encoding is utf-16

04/11/2013 10:11 PM - arton (Akio Tajima)

Status:	Closed		
Priority:	Normal		
Assignee:			
Target version:	2.1.0		
ruby -v:	ruby 2.1.0dev (2013-04-11) [i386-mswin32_100]	**Backport:**	

Description

If File#each_line was given utf-16 encoded file with 'rb:utf-16', each line lacks the last one byte.
For example if the line is "a\0\r\0\n\0" in binary, the read line contains "a\0\r\0\r".

See the attchement.
This issue is appear both current 2.1.0 and 2.0.0.

History

#1 - 04/12/2013 12:11 AM - naruse (Yui NARUSE)

This is because

- UTF-16 is dummy encoding; you must use UTF-16BE, UTF-16LE, or BOM|UTF-* specifier; OR some other treatment is needed on Ruby.
- default line separator is ASCII \n, not UTF-16 \n. you must explicitly specify UTF-16(BE|LE) \n, or convert to some internal encoding; OR some other special treatment is needed on Ruby

#2 - 04/12/2013 01:13 AM - arton (Akio Tajima)

OK, I've fixed my test code. It had some bugs and change the 2nd arg of File#open to 'rb:UTF-16LE'.

Invoking String#rstrip is OK, but can't encode to another encoding from UTF-16LE.

First, I tried to encode utf-16le line to utf-8 using line.rstrip.encode('utf-8') but it failed.

<"This is not a love song."> expected but was
<"\uFFFE\u5400\u6800\u6900\u7300\u2000\u6900\u7300\u2000\u6E00\u6F00\u7400\u2000
\u6100\u2000\u6C00\u6F00\u7600\u6500\u2000\u7300\u6F00\u6E00\u6700\u2E00\u0A00\u
5400\u6800\u6900\u7300\u2000\u6900\u7300\u2000\u6E00\u6F00\u7400\u2000\u6100\u20
00\u6C00\u6F00\u7600\u6500\u2000\u7300\u6F00\u6E00\u6700\u2E00\u0A00">.

Then I tried to encode the line to CP932 with the code " line.rstrip.encode('cp932') "
The result was an exception.

Encoding::UndefinedConversionError: U+FFFE to Windows-31J in conversion from UTF-16LE to UTF-8 to Windows-31J.

Then I've tried to remove BOM from original line with code below:
p line[0] #=> "\uFFFE"
if line[0] == "\uFFFE" # => false, why ? (maybe BOM is nothing here character, but ...)
line = line[1..-1]
end

But nothing changes because the condition line[0] == "\uFFFE" was evaluated to false because if I put else clause, the clause run.

Is there any way to encode UTF-16LE to utf-8 or CP932 ?

#3 - 04/12/2013 01:15 AM - arton (Akio Tajima)

- File test_utf16.rb added

Attachment is the fixed version of test I'd expected the behaviour.

#4 - 04/12/2013 01:21 AM - arton (Akio Tajima)

- Status changed from Open to Closed

Sorry, I've only changed 'rb:utf-16le' when I wrote above comments.

It's running fine if I chaned 'wb:utf-16le' when writing out the file.

#5 - 04/12/2013 01:52 AM - naruse (Yui NARUSE)

Just FYI, you can propose transparent treatment along UTF-16 series ;-)

Files

test_utf16.rb	746 Bytes	04/11/2013	arton (Akio Tajima)
test_utf16.rb	643 Bytes	04/12/2013	arton (Akio Tajima)