Subject: byte/unicode mismatch
Posted by R.Bauer on Thu, 20 Nov 2008 10:19:37 GMT

View Forum Message <> Reply to Message

Hi

the ascii table is gone.

```
IDL> print,byte('ï¿œ')
 195 188
```

A char has now two bytes

```
IDL> help, byte('ï¿œ')
<Expression>    BYTE      = Array[2]
```

This means all of the fast string replacing routines which are related
to iso encoded ascii one byte characters are broken in 7.0

What is the name of the function to convert byte('ï¿œ') into 252b ?

cheers
Reimar

Subject: Re: byte/unicode mismatch
Posted by Michael Galloy on Thu, 20 Nov 2008 17:23:52 GMT

View Forum Message <> Reply to Message

On Nov 20, 3:19 am, Reimar Bauer <R.Ba...@fz-juelich.de> wrote:

```
>  Hi
>
>  the ascii table is gone.
>
>  IDL> print,byte('ü')
>   195 188
>
>  A char has now two bytes
>
>  IDL> help, byte('ü')
>  <Expression>    BYTE      = Array[2]
>
>  This means all of the fast string replacing routines which are related
>  to iso encoded ascii one byte characters are broken in 7.0
>
>  What is the name of the function to convert byte('ü') into 252b ?
```

I guess it is how you type/enter the ü:

```
IDL> u = string(252B)
IDL> print, u
ü
IDL> help, u
U               STRING    = 'ü'
IDL> print, byte(u)
 252
```

Mike
--
www.michaelgalloy.com
Tech-X Corporation
Associate Research Scientist

Subject: Re: byte/unicode mismatch
Posted by Heinz Stege on Thu, 20 Nov 2008 19:08:17 GMT

View Forum Message <> Reply to Message

On Thu, 20 Nov 2008 09:23:52 -0800 (PST), mgalloy@gmail.com wrote:

```
> On Nov 20, 3:19ï¿½am, Reimar Bauer <R.Ba...@fz-juelich.de> wrote:
>>  Hi
>>
>>  the ascii table is gone.
>>
>>  IDL> print,byte('ï¿½')
>>  ï¿½195 188
>>
>>  A char has now two bytes
>>
>>  IDL> help, byte('ï¿½')
>>  <Expression> ï¿½ ï¿½BYTE ï¿½ ï¿½ ï¿½= Array[2]
>>
>>  This means all of the fast string replacing routines which are related
>>  to iso encoded ascii one byte characters are broken in 7.0
>>
>>  What is the name of the function to convert byte('ï¿½') into 252b ?
>
> I guess it is how you type/enter the ï¿½:
>
> IDL> u = string(252B)
> IDL> print, u
> ï¿½
> IDL> help, u
> U               STRING    = 'ï¿½'
> IDL> print, byte(u)
```

> 252
>
> Mike

Probably most readers here don't have an ï¿½-key on their keybord. So
here is another example:

```
IDL> print,!version
{ x86 Win32 Windows Microsoft Windows 7.0 Oct 25 2007      32      64}
IDL> mu='ï¿½'   ; (the greek letter)
IDL> help,mu
MU              STRING    = 'ï¿½'
IDL> help,byte(mu)
<Expression>    BYTE      = Array[2]
IDL> print,byte(mu)
 194 181
```

The string entered in the workbench command line is encoded in UTF8.
Using this string as a title in direct graphics results in a mu
preceded by an "A" with a hat. Direct graphics don't like UTF8. It
would need string(181b) for a mu.

If I don't miss something, Reimar is asking for a function to convert
the UTF8 string to ISO8859(?).

Heinz

Subject: Re: byte/unicode mismatch

Posted by Allan Whiteford on Fri, 21 Nov 2008 10:04:36 GMT
View Forum Message <> Reply to Message

Heinz Stege wrote:
> On Thu, 20 Nov 2008 09:23:52 -0800 (PST), mgalloy@gmail.com wrote:
>
>
>> On Nov 20, 3:19 am, Reimar Bauer <R.Ba...@fz-juelich.de> wrote:
>>
>>> Hi
>>>
>>> the ascii table is gone.
>>>
>>> IDL> print,byte('ï¿½')
>>> 195 188
>>>

> The string entered in the workbench command line is encoded in UTF8.

Picking up on this point (and the one made by Mike) - it's mostly to do with your editor. The workbench seems to be unicode aware so it really is passing a two byte representation of ï¿½ into the interpreter.

If I use the simple command line interface running through an xterm (X.Org 6.8.99.903) which I guess isn't unicode aware then I get 252 with the same version of IDL:

```
IDL> print,!version
{ x86 linux unix linux 7.0 Oct 25 2007      32      64}
IDL> print,byte('ï¿½')
  252
```

but with the workbench:

```
IDL> print,!version
{ x86 linux unix linux 7.0 Oct 25 2007      32      64}
IDL> print,byte('ï¿½')
  195 188
```

I would expect that if you read the character from a file (either as data or in a .pro file) it depends on the program which wrote the file and whether your editor was unicode-aware.

In saying all this, I don't understand unicode properly (does anyone?!?) - I'm just reporting on the fact that it isn't just the IDL interpreter which is the issue, it's to do with the editor which sends the character to the interpreter.

This has already been said - I've just rephrased it using more (unnecessary?) words. I hope it's helpful.

Thanks,

Allan

Subject: Re: byte/unicode mismatch
Posted by R.Bauer on Fri, 21 Nov 2008 10:10:22 GMT

View Forum Message <> Reply to Message

That is all orthogonal.

How can I decode and how can I encode?

cheers
Reimar

Allan Whiteford schrieb:
> Heinz Stege wrote:
>> On Thu, 20 Nov 2008 09:23:52 -0800 (PST), mgalloy@gmail.com wrote:
>>
>>
>>> On Nov 20, 3:19 am, Reimar Bauer <R.Ba...@fz-juelich.de> wrote:
>>>
>>>> Hi
>>>>
>>>> the ascii table is gone.
>>>>
>>>> IDL> print,byte('ï¿½')
>>>> 195 188
>>>>
>
>> The string entered in the workbench command line is encoded in UTF8.
>
> Picking up on this point (and the one made by Mike) - it's mostly to do
> with your editor. The workbench seems to be unicode aware so it really
> is passing a two byte representation of ï¿½ into the interpreter.
>
> If I use the simple command line interface running through an xterm
> (X.Org 6.8.99.903) which I guess isn't unicode aware then I get 252 with
> the same version of IDL:
>
> IDL> print,!version
> { x86 linux unix linux 7.0 Oct 25 2007   32   64}
> IDL> print,byte('ï¿½')
> 252
>
> but with the workbench:
>
> IDL> print,!version
> { x86 linux unix linux 7.0 Oct 25 2007   32   64}
> IDL> print,byte('ï¿½')
> 195 188
>
> I would expect that if you read the character from a file (either as
> data or in a .pro file) it depends on the program which wrote the file
> and whether your editor was unicode-aware.
>
> In saying all this, I don't understand unicode properly (does anyone?!?)
> - I'm just reporting on the fact that it isn't just the IDL interpreter
> which is the issue, it's to do with the editor which sends the character
> to the interpreter.
>
> This has already been said - I've just rephrased it using more
> (unnecessary?) words. I hope it's helpful.

>
> Thanks,
>
> Allan

Subject: Re: byte/unicode mismatch
Posted by Allan Whiteford on Fri, 21 Nov 2008 17:51:13 GMT
View Forum Message <> Reply to Message

Reimar Bauer wrote:
> That is all orthogonal.
>
> How can I decode and how can I encode?
>
> cheers
> Reimar
>

Reimar,

The question (and answer) isn't all that straightforward, byte values over 127 aren't well defined without an encoding system or a codepage.

However, the answer you're probably looking for is:

```
b=byte('ï¿½')   ; assumption 2
print,b[1]+(b[0] eq 195)*64 ; assumption 1
```

which is assuming:

1) you want byte values from (two byte) UTF-8 to ISO-8859-1

and

2) that the u-umlaut character has entered the intepreter from a UTF-8 environment.

Please don't just cut and paste the above assuming all will be well.

Thanks,

Allan

> Allan Whiteford schrieb:
>
>> Heinz Stege wrote:
>>

>>> On Thu, 20 Nov 2008 09:23:52 -0800 (PST), mgalloy@gmail.com wrote:
>>>
>>>
>>>
>>>> On Nov 20, 3:19 am, Reimar Bauer <R.Ba...@fz-juelich.de> wrote:
>>>>
>>>>
>>>> >Hi
>>>> >
>>>> >the ascii table is gone.
>>>> >
>>>> >IDL> print,byte('ï¿½')
>>>> >195 188
>>>> >
>>
>>> The string entered in the workbench command line is encoded in UTF8.
>>
>> Picking up on this point (and the one made by Mike) - it's mostly to do
>> with your editor. The workbench seems to be unicode aware so it really
>> is passing a two byte representation of ï¿½ into the interpreter.
>>
>> If I use the simple command line interface running through an xterm
>> (X.Org 6.8.99.903) which I guess isn't unicode aware then I get 252 with
>> the same version of IDL:
>>
>> IDL> print,!version
>> { x86 linux unix linux 7.0 Oct 25 2007   32   64}
>> IDL> print,byte('ï¿½')
>> 252
>>
>> but with the workbench:
>>
>> IDL> print,!version
>> { x86 linux unix linux 7.0 Oct 25 2007   32   64}
>> IDL> print,byte('ï¿½')
>> 195 188
>>
>> I would expect that if you read the character from a file (either as
>> data or in a .pro file) it depends on the program which wrote the file
>> and whether your editor was unicode-aware.
>>
>> In saying all this, I don't understand unicode properly (does anyone?!?)
>> - I'm just reporting on the fact that it isn't just the IDL interpreter
>> which is the issue, it's to do with the editor which sends the character
>> to the interpreter.
>>
>> This has already been said - I've just rephrased it using more
>> (unnecessary?) words. I hope it's helpful.

>>
>> Thanks,
>>
>> Allan

Subject: Re: byte/unicode mismatch
Posted by R.Bauer on Fri, 21 Nov 2008 21:45:32 GMT
View Forum Message <> Reply to Message

Allan Whiteford schrieb:
> Reimar Bauer wrote:
>> That is all orthogonal.
>>
>> How can I decode and how can I encode?
>>
>> cheers
>> Reimar
>>
>
> Reimar,
>
> The question (and answer) isn't all that straightforward, byte values
> over 127 aren't well defined without an encoding system or a codepage.
>
> However, the answer you're probably looking for is:
>
> b=byte('ï¿½')   ; assumption 2
> print,b[1]+(b[0] eq 195)*64   ; assumption 1
>
> which is assuming:
>
> 1) you want byte values from (two byte) UTF-8 to ISO-8859-1
>
> and
>
> 2) that the u-umlaut character has entered the intepreter from a UTF-8
> environment.
>
> Please don't just cut and paste the above assuming all will be well.
>
> Thanks,
>
> Allan
>

Hmm this does confuse me more. Lets see if an other examples helps me.

If I write an output file using the ide e.g.

```
openw, 10, 'testfile.txt'
printf, 10, 'Jï¿½lich'
close, 10
```

If I run this program with iso encoding isn't the result different to utf-8?

Or how can I write it iso encoded independent from the user setting?
In python I have several methods for that.
http://effbot.org/zone/unicode-objects.htm

cheers
Reimar

Subject: Re: byte/unicode mismatch
Posted by Allan Whiteford on Mon, 24 Nov 2008 13:38:10 GMT

View Forum Message <> Reply to Message

Reimar Bauer wrote:
> Allan Whiteford schrieb:
>> Reimar Bauer wrote:
>>> That is all orthogonal.
>>>
>>> How can I decode and how can I encode?
>>>
>>> cheers
>>> Reimar
>>>
>> Reimar,
>>
>> The question (and answer) isn't all that straightforward, byte values
>> over 127 aren't well defined without an encoding system or a codepage.
>>
>> However, the answer you're probably looking for is:
>>
>> b=byte('ï¿½')   ; assumption 2
>> print,b[1]+(b[0] eq 195)*64   ; assumption 1
>>
>> which is assuming:
>>
>> 1) you want byte values from (two byte) UTF-8 to ISO-8859-1
>>
>> and
>>
>> 2) that the u-umlaut character has entered the intepreter from a UTF-8
>> environment.

>>
>> Please don't just cut and paste the above assuming all will be well.
>>
>> Thanks,
>>
>> Allan
>>
>
> Hmm this does confuse me more. Lets see if an other examples helps me.
>
> If I write an output file using the ide e.g.
>
> openw, 10, 'testfile.txt'
> printf, 10, 'Jï¿½lich'
> close, 10
>
> If I run this program with iso encoding isn't the result different to utf-8?
>

Yes, copying and pasting that code into an IDL interpreter using a UTF-8
environment/editor will give a different output file to using one
without such awareness.

> Or how can I write it iso encoded independent from the user setting?

I would have said check to see if n_elements(byte("Jï¿½lich")) was the
same as strlen("Jï¿½lich") to see if things were UTF-8 or not but it seems
the IDL strlen function actually just counts bytes (I don't think it
should do this).

I'm not sure there is an elegant solution to this problem. In any case,
I'm about to lose my free wi-fi.

Thanks,

Allan

> In python I have several methods for that.
> http://effbot.org/zone/unicode-objects.htm
>
> cheers
> Reimar
>
>
>
>
>
>

>
>
>
>
>
>

Subject: Re: byte/unicode mismatch
Posted by R.Bauer on Tue, 25 Nov 2008 13:03:38 GMT
View Forum Message <> Reply to Message

me has forwarded a feature request to creaso for an en/de- coding
parameter for open and had 5 minutes ago a phonecall about that. Lets see.

Reimar

Allan Whiteford schrieb:
> Reimar Bauer wrote:
>> Allan Whiteford schrieb:
>>> Reimar Bauer wrote:
>>>> That is all orthogonal.
>>>>
>>>> How can I decode and how can I encode?
>>>>
>>>> cheers
>>>> Reimar
>>>>
>>> Reimar,
>>>
>>> The question (and answer) isn't all that straightforward, byte values
>>> over 127 aren't well defined without an encoding system or a codepage.
>>>
>>> However, the answer you're probably looking for is:
>>>
>>> b=byte('ï¿½')   ; assumption 2
>>> print,b[1]+(b[0] eq 195)*64   ; assumption 1
>>>
>>> which is assuming:
>>>
>>> 1) you want byte values from (two byte) UTF-8 to ISO-8859-1
>>>
>>> and
>>>
>>> 2) that the u-umlaut character has entered the intepreter from a UTF-8
>>> environment.

>>>
>>> Please don't just cut and paste the above assuming all will be well.
>>>
>>> Thanks,
>>>
>>> Allan
>>>
>>
>> Hmm this does confuse me more. Lets see if an other examples helps me.
>>
>> If I write an output file using the ide e.g.
>>
>> openw, 10, 'testfile.txt'
>> printf, 10, 'Jï¿½lich'
>> close, 10
>>
>> If I run this program with iso encoding isn't the result different to
>> utf-8?
>>
>
> Yes, copying and pasting that code into an IDL interpreter using a UTF-8
> environment/editor will give a different output file to using one
> without such awareness.
>
>> Or how can I write it iso encoded independent from the user setting?
>
> I would have said check to see if n_elements(byte("Jï¿½lich")) was the
> same as strlen("Jï¿½lich") to see if things were UTF-8 or not but it seems
> the IDL strlen function actually just counts bytes (I don't think it
> should do this).
>
> I'm not sure there is an elegant solution to this problem. In any case,
> I'm about to lose my free wi-fi.
>
> Thanks,
>
> Allan
>
>> In python I have several methods for that.
>> http://effbot.org/zone/unicode-objects.htm
>>
>> cheers
>> Reimar
>>
>>
>>
>>
>>

```
>>
>>
>>
>>
>>
>>
>>
```