
Subject: Re: READ_ASCII truncating data
Posted by [David Fanning](#) on Wed, 12 Mar 2014 13:02:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Simon Mitchell writes:

> I have an issue with READ_ASCII truncating data (IDL 8.01 Win7 64bit).
> I am sure it is something to do with regional settings, the decimal place is set to "," instead of ".".
>
> The file that I am reading is space delimiter (HEX char 20) and my READ_ASCII statement reflects this (ie is default for the delimiter)
>
> red_data = READ_ASCII(redfile, DATA_START = 6, COUNT = nl)
>
> An example of the data from the input file is:
> 0,4571 1334,62 9,356
>
> But when looking at the array after reading, the values are
> 0,0 1334,0 9,0
>
> The same issue occurs if I include DELIMITER=string(20)
>
>
> I do know that if I change the win7 regional settings to have the decimal place "." the data is read correctly without truncating.
> But I do not want to change the settings as I have other software which does not work if it is changed.
> Also, if the input file is changed to replace the "," with "." then the file is read correctly.
>
> Would anyone have any idea as to why the READ_ASCII is behaving this way and have a solution?
> Is it a mistake that I have made in the code, or is there some settings that need to be changed in IDL?
> I cannot find any reference to this problem anywhere and there is nothing on regionalizing decimal place characters that I can find.

You can simply open READ_ASCII up and have a look at why it is behaving like this. I find this bit of code near the top of the file and without investigating too thoroughly imagine this would be a good place to start:

```
; If there are no strings in the line, just split on  
; commas and return.  
if (ARRAY_EQUAL(isQuote, 0b)) then begin  
    return, STRTOK(lineIn, STRING(44b), /EXTRACT, /PRESERVE_NULL)  
endif
```

I think if it were me, I would just write a short routine to read this

file myself, line by line, as READ_ASCII does, but I would do the substitutions of commas for decimal points myself with Mike Galloy's wonderful routine STR_REPLACE (found in the public folder of the Coyote Library because of its usefulness):

```
IDL> line = '123,4 18,456'
IDL> modified_line = str_replace(a, ',', '.', /Global)
IDL> print, modified_line
123.4 18.456
IDL> data = fltarr(2)
IDL> reads, modified_line, data
IDL> print, data
    123.400    18.4560
```

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>

Sepore ma de ni thue. ("Perhaps thou speakest truth.")
