
Subject: Re: Probably a really simple read_ascii question....
Posted by [David Fanning](#) on Thu, 01 Oct 2009 14:09:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

Helen writes:

>
> I am trying to read in a text file (which is in column format) using
> read_Ascii. Most of my variables read in fine, with the exception of
> the "time" column, where data is in HH:MM:SS format. (IDL reads these
> in as 1.0000). Anyone know how to get around this?

I've never used READ_ASCII, but I imagine you will have to read this column in as an 8-element string and process it later. (I'd probably use StrSplit to separate the string into its component parts.)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Probably a really simple read_ascii question....
Posted by [greg.addr](#) on Thu, 01 Oct 2009 17:51:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 1 Okt., 15:59, hethomas <het...@googlemail.com> wrote:
> I am trying to read in a text file (which is in column format) using
> read_Ascii. Most of my variables read in fine, with the exception of
> the "time" column, where data is in HH:MM:SS format. (IDL reads these
> in as 1.0000). Anyone know how to get around this?
> I did try playing with ascii_template (although I don't really want to
> have to use it), but for some reason, won't work and gives me the
> error message:
> Variable is undefined: SCREEN_SIZE.
> Error occurred at: AT_WIDGET 1364 C:\RSI\IDL63\lib
> \ascii_template.pro
> ASCII_TEMPLATE 1690 C:\RSI\IDL63\lib
> \ascii_template.pro
>
> I have the annoying feeling that this can be answered

> straightforwardly, and I just can't see it! Thanks in advance....
>
> Helen

read_ascii is painful to use. It's simpler to use function like this:

```
function read_textfile,filename
  n=file_lines(filename)
  s=strarr(n)
  openr,lun,filename,/get_lun
  readf,lun,s
  free_lun,lun
  return,s
end
```

and deal with the strings yourself. For time formats, you can make use of this for decoding:

```
IDL> t='12:34:26'
IDL> time=0d ;you have to declare a double in advance...
IDL> reads,t,time,format='(C(CHI,x,CMI,x,CSI))'
IDL> print,string(time,format='(C())')
Sat Jan 01 12:34:26  1
```

Greg
