Subject: Re: End of File-Error message in simple readf routine - What's the problem?

Posted by Conor on Mon, 20 Aug 2007 13:16:24 GMT

View Forum Message <> Reply to Message

On Aug 20, 4:31 am, britta....@gmail.com wrote:
> On 20 Aug., 09:23, britta....@gmail.com wrote:
>
>
>
>> On 20 Aug., 02:41, "mgal...@gmail.com" <mgal...@gmail.com> wrote:
>
>>> On Aug 19, 4:53 am, b...@uni-mainz.de wrote:
>
>>>> Hello,
>
>>>> i'm sorry to post a new message, but i did not find the answer to my
>>>> question, when i searched for it (or i did not use the fitting words
>>>> for the search).
>
>>>> I try to read the values of a file with three columns and 2072758 rows
>>>> (the file was created from a bmp-picture by the routine read_bmp).
>
>>>> My code is as follows:
>
>>>> file='E:\Dissertation\mz_cam\spectral_calib\dat-files
>>>> \070807\run24\070807_24_0013.dat'
>
>>>> n_pixel= 2072758
>>>> n_channels=3
>
>>>> dummy_ein=fltarr(n_channels)
>>>> dummy = ''
>
>>>> r=fltarr(n_pixel)
>>>> g=fltarr(n_pixel)
>>>> nir=fltarr(n_pixel)
>
>>>> close, 1
>>>> openr,1, file
>
>>>> for i=0, n_pixel-1 do begin
>>>>   readf, 1, dummy_ein
>>>>   g(i)=dummy_ein(0)
>>>>   r(i)=dummy_ein(1)
>>>>   nir(i)=dummy_ein(2)
>>>> endfor
>

```
>>>>  end
>
>>>>  Could the problem be related to the array length? I mean that there
>>>>  are too many values for an array? I checked the number of values
>>>>  several times, so i am quite sure that the number for n_pixel is
>>>>  correct.
>
>>>  To keep the loop, you definitely need the "L", that should fix the
>>>  "loop expression too large for loop variable type." error.
>
>>>  You didn't mention what the original problem was though. You did
>>>  mention later that there was an "end of file error." So I would make
>>>  sure there really are n_pixel number of rows in the data file. (Use
>>>  "nLines = file_lines(file)" to find out.)
>
>>>  By the way, you can do this without a loop:
>
>>>  file = 'E:\Dissertation\mz_cam\spectral_calib\dat-files
>>>  \070807\run24\070807_24_0013.dat'
>
>>>  n_pixel = 2072758
>>>  n_channels =3
>
>>>  data = fltarr(n_channels, n_pixel)
>
>>>  openr, lun, file, /get_lun
>>>  readf, lun, data
>>>  free_lun, lun
>
>>>  ; if you need the data in separate arrays
>>>  g = data[0, *]
>>>  r = data[1, *]
>>>  nir = data[2, *]
>
>>>  Mike
>>>  --www.michaelgalloy.com
>
>>  Hello,
>
>>  oh yes, i forgot in the original post to mention my problem. Sorry.
>>  I'll try to fix my problem with your suggestions.
>
>>  Thank you,
>
>>  Britta
>
>  Hello again,
>
```

> i tried your suggestions, but i still get the "end of file-error".
> Even if i tried the suggested code without the loop. Do you have any
> idea what could be further wrong? I checked the number of lines with
> "nLines = file_lines(file)". The number is correct :-(.
>
> Yours,
> Britta

Another possibility is that there are not exactly three columns in each line of text. Is there any missing data? Take the following (common) example:

Contents of File:

```
1234 5678 9012
1234      9012
1234 5678 9012
```

```
data = intarr(3,3)
openr, lun, 'File', /get_lun
readf, lun, data
free_lun, lun
```

In this case you will also get an EOF error. The reason is because IDL is trying to read 8 values into a 9-element array. By default, IDL's read operation seems to be a string-split operation. It splits up the line wherever it encounters spaces. So, when it reads the second line it splits it up and only finds two values. Since it needs three values in a row, it goes ahead and continues to the next line and reads the first value in that line. Now, IDL will be one element ahead of the game, and when it tries to read in another three values for the third row of your data element, it prematurely reaches the end of the file because there are only two actual data values remaining in the file. The solution in this case is to replace the blank spaces with junk data that is easily recognizable as "no data". For instance:

New Contents of File:

```
1234 5678 9012
1234 -999 9012
1234 5678 9012
```

Then, IDL will no longer have an EOF error, and you simply select out any data less than -900