
Subject: Re: Reading and Plotting big txt. File
Posted by [Conor](#) on Wed, 01 Aug 2007 16:43:09 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Aug 1, 12:31 pm, "incognito.me" <incognito...@gmh.de> wrote:

> On 1 Aug., 18:15, Conor <cmanc...@gmail.com> wrote:

>

>

>

>> On Aug 1, 10:49 am, "incognito.me" <incognito...@gmh.de> wrote:

>

>>> On 1 Aug., 14:44, Conor <cmanc...@gmail.com> wrote:

>

>>>> On Aug 1, 6:25 am, greg.a...@googlemail.com wrote:

>

>>>> > On Aug 1, 11:33 am, "incognito.me" <incognito...@gmh.de> wrote:

>

>>>> > > I'm trying to read and plot (surface) a very big text (.txt) file

>>>> > > (1020, 1024) with a 5 line string Header in IDL. My file looks like a

>>>> > > circle made of numbers!!!. That means in some lines and columns there

>>>> > > are no numbers only blanks!!!for example my file contains integers

>>>> > > between rows 633 and 390 and between columns 650 and 406.At the left

>>>> > > side of the file, there are the numbers of rows (1023,1022,1021,...,0)

>>>> > > my code should not read, but it does. And I also notice, that my code

>>>> > > don't begin to read where the data starts!!By running the code I have

>>>> > > the following error message: READF: End of file encountered. Unit: 1.

>>>> > > Can someone help me?

>>>> > > This is how my code looks like

>>>> > > pro readfile, filename

>

>>>> > > ; file=strupcase(filename)

>>>> > > rows=file_lines(file)

>>>> > > ;open the file and read the five line header.

>>>> > > openr,1,file

>>>> > > header=strarr(5)

>>>> > > readf,1,header

>>>> > > ; Find the number of columns in the file

>>>> > > cols=fix(strmid(header(3),14,4))

>>>> > > ; Number of rows of the data

>>>> > > rows_data=rows-n_elements(header)

>

>>>> > > ;Create a big array to hold the data

>>>> > > data=intarr (cols, rows_data)

>>>> > > ; All blanks should be replaced by zero

>>>> > > data[where(data eq ' ')] = 0

>>>> > > ; A small array to read a line

>>>> > > s=intarr(cols)

>>>> > > n=0

```

>>>> >> while (~ eof(1) and (n lt rows_data -1 )) do begin
>>>> >>   ; Read a line of data
>>>> >>   readf,1,s
>>>> >>   ; Store it in data
>>>> >>   data[*,n]=s
>>>> >>   n=n+1
>>>> >> end
>>>> >> data=data[*,0:n-1]
>
>>>> >> CLOSE,1
>>>> >> Shade_surf, data
>>>> >> end
>
>>>> >> thanks
>
>>>> >> incognito
>
>>>> > I'm suspicious of the line converting blanks to zeros before you've
>>>> > even read them. I don't think the blanks will come out the way you're
>>>> > expecting, anyway. I'd suggest you write a program to correctly read
>>>> > your first line of data before you go for the whole thing.
>
>>>> > Greg
>
>>>> For starters, I'm not sure why you are converting blanks to zeroes
>>>> there at all. As far as I can tell, you haven't even initialized any
>>>> data yet. It seems like you are trying to convert blanks to zeros on
>>>> an integer array which is already filled with zeroes anyway. When I
>>>> tried to do that, I got this error:
>
>>>> % Type conversion error: Unable to convert given STRING to Integer.
>
>>>> Which isn't a fatal error, so your code would still run but the line
>>>> 'data[where(data eq ' ')] = 0' wouldn't actually do anything. As for
>>>> the rest of your problem, I think what you need is a format
>>>> statement. I believe what is happening is that because you haven't
>>>> included an explicit format statement (telling it how many columns are
>>>> on each line) it simply reads in entries until it fills up a row in
>>>> your data array. For instance, look at this file:
>
>>>> 12 34 698 934
>>>> 16      18
>>>> 17 20    13
>>>> 14 23 234 123
>
>>>> being read by this pseudo-code:
>
>>>> readf,lun,file,/get_lun

```

```

>>>> data = intarr(4)
>>>> readf,lun,data
>>>> print,data
>>>> ; 12 34 698 934
>>>> readf,lun,data
>>>> print,data
>>>> ; 16 13 17 20
>>>> readf,lun,data
>>>> print,data
>>>> ; 14 23 234 123
>>>> readf,lun,data
>>>> % READF: End of file encountered. Unit: 100, File: test
>
>>>> See, because you have no format specified, each readf keeps reading
>>>> data in until the data array is filled. You are assuming that readf
>>>> reads one line at a time, but that's not happening, which is why your
>>>> data isn't where it's supposed to be. Also, because it is reading
>>>> faster than one line at a time, you are reading to the end of the file
>>>> before you call readf (rows_data) times, and then you get the EOF
>>>> error. The solution is to give it a format:
>
>>>> IDL> openr,lun,'test',/get_lun
>>>> IDL> format = '(i3, 1x, i3, 1x, i3, 1x, i3)'
>>>> IDL> readf,lun,test,format=format
>>>> IDL> print,test
>>>> 12 34 698 934
>>>> IDL> readf,lun,test,format=format
>>>> IDL> print,test
>>>> 16 0 0 18
>>>> IDL> readf,lun,test,format=format
>>>> IDL> print,test
>>>> 17 20 0 13
>>>> IDL> readf,lun,test,format=format
>>>> IDL> print,test
>>>> 14 23 234 123- Zitierten Text ausblenden -
>
>>>> - Zitierten Text anzeigen -
>
>>> Hi Conor,
>
>>> Thanks for your suggestions!!I muss agree,to fill the blanks with
>>> zeroes was not so cute!!! have to read how one uses the keyword format
>>> with readf again,because I should confest I haven't unsterstood
>>> yet.Could you please give me a hint?
>>> Thanks a lot,
>>> Kind regards
>>> C.
>

```

```

>> Unfortunately, I'm not so great with format statements, I don't use
>> them so much, and I've never used them for reading files. The general
>> idea for reading floats is that you specify the total number of
>> characters to read, and how many numbers come after the decimal
>> place. So, for instance the number:
>
>> 123.456789
>
>> would be specified by the statement:
>
>> (f10.6)
>
>> There are ten characters that must be read (9 digits, plus the decimal
>> point) and there are 6 digits after the period. For spaces you use
>> '1x' (or '2x' for two spaces, etc...). So for instance the line:
>
>> 134.367 123.45 123.92
>
>> would be specified by:
>
>> (f7.3, 1x, f6.2, 1x, f6.2)
>
>> Also, you can specify that IDL should "repeat" a format statement.
>> For instance, you could also represent the last one with:
>
>> (f7.3, 2(1x, f6.2) )
>
>> This last part is very important to you because you won't want to
>> write out the format statement for all 1000 of your columns. In fact,
>> IDL won't let you specify that many anyway. With any luck, all the
>> columns have the same fixed width (or at least a repeating pattern) so
>> you can do something like this:
>
>> (f10.5, 999(1x, f12.1) )
>
>> Exactly how it will work I don't know. You might just have to play
>> around with it. As I said, I'm not terribly familiar with format
>> statements myself, so this might not be the best way to do it. Maybe
>> someone else has some suggestions?- Zitierten Text ausblenden -
>
>> - Zitierten Text anzeigen -
>
> Hi Conor,
>
> I'm now better understanding how the format statement works.I will
> jetzt
> managed to understand how it works with negative integers.I think,it
> won't

```

> be so different. Thanks a lot for the hint. It was very helpful.
> Kind regards,
> C.

Negative integers are pretty similar.

-1234

would be:

(i5)
