Subject: Re: reading in a long line of data
Posted by ben.bighair on Thu, 08 May 2008 16:38:53 GMT
View Forum Message <> Reply to Message

On May 8, 12:18 pm, te...@atmsci.msrc.sunysb.edu wrote:
> Hi, I am trying to read in a long line which has around 2520 values
> all in scientific notation (ie 1.2E+02, etc). I tried the following:
>
> x=strarr(2520)
> x[*]='12345678'
> readu,unit,x
>
> Then converting to floating point. This doesn't seem to work since
> the data is in scientific notation. Does anyone have any suggestions?
>

Hi,

The easiest way might be the following...

```
x = ""
READF, U, x
x = FLOAT(STRSPLIT(x, ",",/EXTRACT))
```

which assumes comma delimited data.

If you know exactly how many items are in a row you can get fancier
with explicit format statements. But this simpler form should be OK.

Cheers,
Ben

Subject: Re: reading in a long line of data
Posted by David Fanning on Thu, 08 May 2008 16:43:50 GMT
View Forum Message <> Reply to Message

teich@atmsci.msrc.sunysb.edu writes:

> Hi, I am trying to read in a long line which has around 2520 values
> all in scientific notation (ie 1.2E+02, etc). I tried the following:
>
> x=strarr(2520)
> x[*]='12345678'
> readu,unit,x
>
> Then converting to floating point. This doesn't seem to work since
> the data is in scientific notation. Does anyone have any suggestions?

I would suggest less fuzzy thinking. :-)

What is a "line" of data? Computers don't do
"around" or "about" anything. They are like
children. They have to be told explicitly.

Cheers,

David

P.S. Do you know anything more exact about this data?
Do you know, for example, if it is really saved in
an unformatted data file?

--
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: reading in a long line of data
Posted by teich on Thu, 08 May 2008 16:56:57 GMT
View Forum Message <> Reply to Message

Sorry, maybe I should have been more explicit. The data is not
unformatted but I tried that because otherwise I get an error messege:

Input line is too long for input buffer of 32767 characters.

Ben, thanks for your suggestion, but I also get that error messege
trying your suggestion.

To be more explicit, each line of data looks something like this
(where there are 2520 columns):

1.0117065e+003 1.0114794e+003 1.0112352e+003 1.0110832e+003
1.0109401e+003 ...

Thanks,

Howard

On May 8, 12:43 pm, David Fanning <n...@dfanning.com> wrote:
> te...@atmsci.msrc.sunysb.edu writes:

>> Hi, I am trying to read in a long line which has around 2520 values
>> all in scientific notation (ie 1.2E+02, etc). I tried the following:
>
>> x=strarr(2520)
>> x[*]='12345678'
>> readu,unit,x
>
>> Then converting to floating point. This doesn't seem to work since
>> the data is in scientific notation. Does anyone have any suggestions?
>
> I would suggest less fuzzy thinking. :-)
>
> What is a "line" of data? Computers don't do
> "around" or "about" anything. They are like
> children. They have to be told explicitly.
>
> Cheers,
>
> David
>
> P.S. Do you know anything more exact about this data?
> Do you know, for example, if it is really saved in
> an unformatted data file?
>
> --
> 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: reading in a long line of data
Posted by David Fanning on Thu, 08 May 2008 17:08:22 GMT
View Forum Message <> Reply to Message

teich@atmsci.msrc.sunysb.edu writes:

> Sorry, maybe I should have been more explicit.

Well, only if you want a sensible answer to your question. :-)

> The data is not
> unformatted but I tried that because otherwise I get an error messege:
>
> Input line is too long for input buffer of 32767 characters.
>
> Ben, thanks for your suggestion, but I also get that error messege
> trying your suggestion.

>
> To be more explicit, each line of data looks something like this
> (where there are 2520 columns):
>
>
> 1.0117065e+003 1.0114794e+003 1.0112352e+003 1.0110832e+003
> 1.0109401e+003 ...

Try this then:

```
Openr, lun, 'yourdatafile.dat', /Get_Lun
data = fltarr(2520)
ReadF, lun, data
Free_lun, lun
```

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: reading in a long line of data
Posted by teich on Thu, 08 May 2008 17:14:05 GMT
View Forum Message <> Reply to Message

On May 8, 1:08 pm, David Fanning <n...@dfanning.com> wrote:
> te...@atmsci.msrc.sunysb.edu writes:
>> Sorry, maybe I should have been more explicit.
>
> Well, only if you want a sensible answer to your question. :-)
>
>> The data is not
>> unformatted but I tried that because otherwise I get an error messege:
>
>> Input line is too long for input buffer of 32767 characters.
>
>> Ben, thanks for your suggestion, but I also get that error messege
>> trying your suggestion.
>
>> To be more explicit, each line of data looks something like this
>> (where there are 2520 columns):
>
>> 1.0117065e+003 1.0114794e+003 1.0112352e+003 1.0110832e+003
>> 1.0109401e+003 ...

>
> Try this then:
>
> Openr, lun, 'yourdatafile.dat', /Get_Lun
> data = fltarr(2520)
> ReadF, lun, data
> Free_lun, lun
>
> 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.")

That's typically what I do. I think the version of IDL I am using
doesn't like this because the line of data is too long so I still get
the same error.

Subject: Re: reading in a long line of data
Posted by David Fanning on Thu, 08 May 2008 17:24:25 GMT
View Forum Message <> Reply to Message

teich@atmsci.msrc.sunysb.edu writes:

> That's typically what I do. I think the version of IDL I am using
> doesn't like this because the line of data is too long so I still get
> the same error.

Sigh...

I'm out of time to work on this, but next time you
try a question, please let us know what you have tried,
what didn't work, and what the details are. I makes it
a LOT easier to find time to help.

I remember a discussion about this on the newsgroup
recently. I can't remember what the bottom line was.
Probably try to divide the file up into shorter lines.
But I have to go.

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: reading in a long line of data
Posted by Spon on Fri, 09 May 2008 11:15:44 GMT
View Forum Message <> Reply to Message

On May 8, 6:24 pm, David Fanning <n...@dfanning.com> wrote:
> te...@atmsci.msrc.sunysb.edu writes:
>> That's typically what I do. I think the version of IDL I am using
>> doesn't like this because the line of data is too long so I still get
>> the same error.
>
> Sigh...
>
> I'm out of time to work on this, but next time you
> try a question, please let us know what you have tried,
> what didn't work, and what the details are. I makes it
> a LOT easier to find time to help.
>
> I remember a discussion about this on the newsgroup
> recently. I can't remember what the bottom line was.
> Probably try to divide the file up into shorter lines.
> But I have to go.
>
> 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.")

David,

you once wrote a nice little programme called Read_Tab_Lines which may
be of some use here:

http://groups.google.com/group/comp.lang.idl-pvwave/msg/7b09 330c920aefea

I'm not sure how ReadF handles cross-platform compatability regarding
carriage return / line feed use to determine where a row ends.
Sometimes ReadF seems to stop at the end of a row for me; sometimes it
thinks the whole file is one row, even if I can *see* the carriage

returns in it when I open it in WordPad.

Regards,
Chris

Subject: Re: reading in a long line of data
Posted by David Fanning on Fri, 09 May 2008 13:28:32 GMT
View Forum Message <> Reply to Message

Spon writes:

> you once wrote a nice little programme called Read_Tab_Lines which may
> be of some use here:

Humm, I don't think so, since the problem is that the file
is filling up the character buffer.

I think the usual solution it to open this file in some
kind of word processor, turn on word wrapping, and save
the file with some reasonable length lines. But some word
processors won't allow you to load a line of that length
either.

In that case, I think you have to make a large byte
array and try to read the data that way, then process
the byte array to obtain the data. I guess I would
try something like this.

```
OpenR, lun, 'myfile.dat', /Get_Lun
info = FSTAT(lun)
data = BytArr(info.size)
ReadU, lun, data
Free_lun, lun
actualData = Float(StrSplit(String(data), /EXTRACT))
```

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: reading in a long line of data

OK, I must be missing something. Doesn't READ_ASCII work on this type
of file?

Ed

On May 8, 12:56 pm, te...@atmsci.msrc.sunysb.edu wrote:
> Sorry, maybe I should have been more explicit. The data is not
> unformatted but I tried that because otherwise I get an error messege:
>
> Input line is too long for input buffer of 32767 characters.
>
> Ben, thanks for your suggestion, but I also get that error messege
> trying your suggestion.
>
> To be more explicit, each line of data looks something like this
> (where there are 2520 columns):
>
> 1.0117065e+003 1.0114794e+003 1.0112352e+003 1.0110832e+003
> 1.0109401e+003 ...
>
> Thanks,
>
> Howard
>
> On May 8, 12:43 pm, David Fanning <n...@dfanning.com> wrote:
>
>> te...@atmsci.msrc.sunysb.edu writes:
>>> Hi, I am trying to read in a long line which has around 2520 values
>>> all in scientific notation (ie 1.2E+02, etc). I tried the following:
>
>>> x=strarr(2520)
>>> x[*]='12345678'
>>> readu,unit,x
>
>>> Then converting to floating point. This doesn't seem to work since
>>> the data is in scientific notation. Does anyone have any suggestions?
>
>> I would suggest less fuzzy thinking. :-)
>
>> What is a "line" of data? Computers don't do
>> "around" or "about" anything. They are like
>> children. They have to be told explicitly.
>
>> Cheers,
>
>> David

>
>> P.S. Do you know anything more exact about this data?
>> Do you know, for example, if it is really saved in
>> an unformatted data file?
>
>> --
>> 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: reading in a long line of data
Posted by David Fanning on Fri, 09 May 2008 14:26:37 GMT
View Forum Message <> Reply to Message

edward.s.meinel@aero.org writes:

> OK, I must be missing something. Doesn't READ_ASCII work on this type
> of file?

Not sure what "type of file" you are talking about.

Our problem is that the first line of the file in question
is so long it is filling up the input buffer such that READF
doesn't work. I've never used READ_ASCII in my life,
but I can't imagine it uses anything *except* READF
to do its job.

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: reading in a long line of data
Posted by Spon on Fri, 09 May 2008 14:31:43 GMT
View Forum Message <> Reply to Message

On May 9, 2:28 pm, David Fanning <n...@dfanning.com> wrote:
> Spon writes:
>> you once wrote a nice little programme called Read_Tab_Lines which may

>> be of some use here:
>
> Humm, I don't think so, since the problem is that the file
> is filling up the character buffer.
>
> I think the usual solution it to open this file in some
> kind of word processor, turn on word wrapping, and save
> the file with some reasonable length lines. But some word
> processors won't allow you to load a line of that length
> either.
>
> In that case, I think you have to make a large byte
> array and try to read the data that way, then process
> the byte array to obtain the data. I guess I would
> try something like this.
>
>   OpenR, lun, 'myfile.dat', /Get_Lun
>   info = FSTAT(lun)
>   data = BytArr(info.size)
>   ReadU, lun, data
>   Free_lun, lun
>   actualData = Float(StrSplit(String(data), /EXTRACT))
>
> 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.")

Is there any reason not to use the following?

OpenR, lun, 'myfile.dat', /Get_Lun
info = FSTAT(lun)
data = FLTARR(Info.Size / 4L)
readu, lun, data
Free_lun, lun

There's something fundamental about this input buffer that I'm not
getting here :-(

I have a file that looks like this:
8.06"89676e-002 9.0884782e-002 5.1953532e-002 8.1742041e-002
5.8772590e-002 6.7513607e-002 5.1806606e-002 "...

And I'm easily able to read 1.8 million floats from it at once, like

this:

```
OPENR, Lun, File, /GET_LUN
FileSize = (FSTAT(Lun)).Size / 4L
Data = FLTARR(FileSize)
READU, Lun, Data
FREE_LUN, Lun
HELP, Data
IDL> DATA           FLOAT     = Array[1800720]
PRINT, Data[1800710:*]
IDL> 1.16934e-019 2.69980e-006 1.67846e-007 6.30809e-010 4.93642e-031
4.14105e-011 1.04315e-008 5.46591e+022 1.02555e-008 1.16934e-019
```

Surely, this is equivalent to (1800720 * 4) bytes, or (1800720 * 4 *
8) bits, or (1800720 * 13) ASCII chars, and therefore massively bigger
than the buffer. Yet IDL has no problem doing this. What am I missing?
Does the buffer only come into play with ReadF? How come?

Regards,
Chris

Subject: Re: reading in a long line of data
Posted by David Fanning on Fri, 09 May 2008 14:37:37 GMT
View Forum Message <> Reply to Message

Spon writes:

> Surely, this is equivalent to (1800720 * 4) bytes, or (1800720 * 4 *
> 8) bits, or (1800720 * 13) ASCII chars, and therefore massively bigger
> than the buffer. Yet IDL has no problem doing this. What am I missing?
> Does the buffer only come into play with ReadF?

Apparently so.

> How come?

You've got me. Maybe Karl or someone else from ITTVIS can
shed some light on this. To be fair, it is usually some kind
of programming error that causes this problem to surface. But
I would like to understand it, since it comes up from time to
time.

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: reading in a long line of data
Posted by Spon on Fri, 09 May 2008 14:42:47 GMT
View Forum Message <> Reply to Message

On May 9, 3:03 pm, "edward.s.mei...@aero.org" <mei...@aero.org> wrote:
> OK, I must be missing something. Doesn't READ_ASCII work on this type
> of file?
>
> Ed

So must I. At the very least, it's surely got to be worth attempting:

File = 'myfile.dat'
Data = READ_BINARY(File, DATA_TYPE = 4)

I guess it depends on what else is in the file, apart from the data.

Regards,
Chris

Subject: Re: reading in a long line of data
Posted by Spon on Sun, 11 May 2008 08:14:12 GMT
View Forum Message <> Reply to Message

On May 9, 12:15 pm, Spon <christoph.b...@gmail.com> wrote:
> Sometimes ReadF seems to stop at the end of a row for me; sometimes it
> thinks the whole file is one row, even if I can *see* the carriage
> returns in it when I open it in WordPad.
>
> Regards,
> Chris

In retrospect, this should probably read 'Sometimes I use ReadF and
sometimes I use ReadU, and I shouldn't be surprised that the two don't
give the same results.' :-D
