
Subject: Re: the last line of a large file

Posted by [Jean H.](#) on Thu, 09 Aug 2007 20:07:16 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

If you are sure it is the last line that you want to omit, you could do something like:

```
nbOfLines = FILE_LINES(YourFileName)
```

```
for i = 0, nbOfLines - 2 do begin
```

```
  readf, in_unit, input_line
```

```
endfor
```

Jean

> Hi,

>

> I am processing a large file with data in the same format till the
> last line. In the last line, it states how many records are included
> in this file.

>

> So the structure of my program is:

>

> while not eof(in_unit) do begin

> readf, in_unit, input_line

> if(input_line ne 'last line) then begin

>

> else

>

> endif

> endwhile

>

> Do I have to use 'if/then' to test whether every input_line is the
> last line of the file? Since there are many data records in the file,
> repeat calls to 'if/then' can be time consuming. But if I don't do the
> test, the program will be halted when it read in the last line.

>

> A Easy way I can think of is to delete the last line, but sometime we
> are not supposed to change the input files.

>

Subject: Re: the last line of a large file

Posted by [Paul Van Delst\[1\]](#) on Thu, 09 Aug 2007 21:22:46 GMT

[View Forum Message](#) <> [Reply to Message](#)

Jean H. wrote:

```
> Hi,  
>  
> If you are sure it is the last line that you want to omit, you could do  
> something like:  
>  
> nbOfLines = FILE_LINES(YourFileName)  
>  
> for i = 0, nbOfLines - 2 do begin  
>   readf, in_unit, input_line  
> endfor
```

Or, if each record is the same format and consists only of numbers (no A-Z characters), you could probably get rid of the above loop too. E.g. say you have 10 columns of numbers per record (line)

```
n_Records = FILE_LINES(YourFileName) - 1L  
...open the file....  
All_the_Useful_Data = FLTARR(10,n_Records)  
readf, in_unit, All_the_Useful_Data
```

?

I think that should work (or something like it).

Of course, if you have records split over multiple lines, then it gets hairier.

```
>>  
>> I am processing a large file with data in the same format till the  
>> last line. In the last line, it states how many records are included  
>> in this file.  
>>  
>> So the structure of my program is:  
>>  
>> while not eof(in_unit) do begin  
>>   readf, in_unit, input_line  
>>   if( input_line ne 'last line) then begin  
>>     ....  
>>   else  
>>     ....  
>>   endif  
>> endwhile  
>>  
>> Do I have to use 'if/then' to test whether every input_line is the  
>> last line of the file? Since there are many data records in the file,  
>> repeat calls to 'if/then' can be time consuming. But if I don't do the  
>> test, the program will be halted when it read in the last line.  
>>
```

>> A Easy way I can think of is to delete the last line, but sometime we
>> are not supposed to change the input files.

You should also tell the people who wrote the code that creates the files that the number of records information is more useful if they stick it at the beginning of the file. :o)

cheers,

paulv

Subject: Re: the last line of a large file
Posted by [Conor](#) on Fri, 10 Aug 2007 12:34:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Aug 9, 5:22 pm, Paul van Delst <Paul.vanDe...@noaa.gov> wrote:

> Jean H. wrote:

>> Hi,

>

>> If you are sure it is the last line that you want to omit, you could do
>> something like:

>

>> nbOfLines = FILE_LINES(YourFileName)

>

>> for i = 0, nbOfLines - 2 do begin

>> readf, in_unit, input_line

>> endfor

>

> Or, if each record is the same format and consists only of numbers (no A-Z characters),

> you could probably get rid of the above loop too. E.g. say you have 10 columns of numbers
> per record (line)

>

> n_Records = FILE_LINES(YourFileName) - 1L

> ...open the file....

> All_the_Useful_Data = FLTARR(10,n_Records)

> readf, in_unit, All_the_Useful_Data

>

> ?

>

> I think that should work (or something like it).

>

> Of course, if you have records split over multiple lines, then it gets hairier.

>

>

>

>

>

>>> I am processing a large file with data in the same format till the

```
>>> last line. In the last line, it states how many records are included
>>> in this file.
>
>>> So the structure of my program is:
>
>>> while not eof(in_unit) do begin
>>>   readf, in_unit, input_line
>>>   if( input_line ne 'last line') then begin
>>>     ....
>>>   else
>>>     ....
>>>   endif
>>> endwhile
>
>>> Do I have to use 'if/then' to test whether every input_line is the
>>> last line of the file? Since there are many data records in the file,
>>> repeat calls to 'if/then' can be time consuming. But if I don't do the
>>> test, the program will be halted when it read in the last line.
>
>>> A Easy way I can think of is to delete the last line, but sometime we
>>> are not supposed to change the input files.
>
> You should also tell the people who wrote the code that creates the files that the number
> of records information is more useful if they stick it at the beginning of the file. :o)
>
> cheers,
>
> paulv
```

lol! Really! What in the world is the point of putting the number of lines at the end of the file? That's got to be about the silliest thing I've ever heard of. By the time you've actually read to the end of the file, you already know how many lines there are!

Subject: Re: the last line of a large file
Posted by [queiny](#) on Fri, 10 Aug 2007 14:00:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

I think the file_lines works!

Since the file is very large, I can't have one array to read them all in once. IDL complained and said there are too many elements in the array, so it didn't even let me declare it.

The data files are compiled from multiple sources so they only know how many records they have in the end.

>

> lol! Really! What in the world is the point of putting the number of
> lines at the end of the file? That's got to be about the silliest
> thing I've ever heard of. By the time you've actually read to the end
> of the file, you already know how many lines there are!

Subject: Re: the last line of a large file

Posted by [David Fanning](#) on Fri, 10 Aug 2007 14:10:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

Conor writes:

> Really! What in the world is the point of putting the number of
> lines at the end of the file? That's got to be about the silliest
> thing I've ever heard of. By the time you've actually read to the end
> of the file, you already know how many lines there are!

I think you might be overestimating the abilities of young programmers who have been trained by video games and text messaging. :-)

Cheers,

David

P.S. You could spend a couple of days answering my e-mails if you wish a more comprehensive view of the world. ;-)

--

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: the last line of a large file

Posted by [Paul Van Delst\[1\]](#) on Fri, 10 Aug 2007 14:37:03 GMT

[View Forum Message](#) <> [Reply to Message](#)

queiny wrote:

> I think the file_lines works!

>

> Since the file is very large, I can't have one array to read them all
> in once. IDL complained and said there are too many elements in the
> array, so it didn't even let me declare it.

Weird. Exactly how many elements are there? Or, how large is the file?

- > The data files are compiled from multiple sources so they only know
- > how many records they have in the end.

Well, that's really not an excuse for poorly designed code, but I guess it's understandable. As a bloke said recently on another newsgroup:

"I always prefer practical, then elegant, then subtle, or I'd never get anything done at all."

cheers,

paulv

- >> lol! Really! What in the world is the point of putting the number of
- >> lines at the end of the file? That's got to be about the silliest
- >> thing I've ever heard of. By the time you've actually read to the end
- >> of the file, you already know how many lines there are!

>
>

Subject: Re: the last line of a large file

Posted by [Carsten Lechte](#) on Fri, 10 Aug 2007 15:00:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

Conor wrote:

- > lol! Really! What in the world is the point of putting the number of
- > lines at the end of the file?

One legitimate reason would be that sometimes you only know how much data you have until after you have processed it all, especially if the data sets are so large that you only ever have a small subset in RAM.

A legitimate example are zip archives, where the table of contents is written to the end of the file, because the the compressed sizes of the archive members cannot be known in advance, and it would double the running time to determine the compressed size beforehand, it would furthermore use twice the disk space to re-write the file with the contents in front, it would be impossible to keep the whole archive in RAM before writing it, and finally, one may not be able leave space for the contents table at the beginning of the file, to be filled in later, because one would have to know how long the table will be beforehand...

Of course, this does not mean that the original poster's data has a legitimate reason for being organised like this.

For the original poster's problem, one idea is to get the file size

in bytes, skip to position `file_size-1000`, read that small chunk and parse it for the desired metadata. This might even be faster than actually counting the lines with `FILE_LINES`, but it is probably only worth it if the metadata contains more useful information than just the number of lines in the file.

chl
