
Subject: Re: Reading a set of data with string type entries

Posted by [izimine](#) on Fri, 30 Nov 2007 07:07:31 GMT

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

```
IDL> s = strjoin(['info', string(randomu(seed,10))], ' ')
```

```
IDL> vals = float((strsplit(s, ' ', /regex, /extract))[1:*])
```

For large files I sometimes read the whole thing into a bytarr and
split on line breaks
something like

```
openr, lun, 'big.txt', /get_lun
```

```
fs = fstat(lun)
```

```
buf = bytarr(fs.size)
```

```
readu, lun, buf
```

```
free_lun, lun
```

```
buf = strsplit(string(buf), string([13b,10b]), /extract) ; split on CR
```

```
+LF
```

Subject: Re: Reading a set of data with string type entries

Posted by [AlbertoPaolo.Meroni](#) on Fri, 30 Nov 2007 07:32:40 GMT

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

On 30 Nov, 08:07, izim...@gmail.com wrote:

```
> IDL> s = strjoin(['info', string(randomu(seed,10))], ' ')
```

```
> IDL> vals = float((strsplit(s, ' ', /regex, /extract))[1:*])
```

```
>
```

```
> For large files I sometimes read the whole thing into a bytarr and
```

```
> split on line breaks
```

```
> something like
```

```
>
```

```
> openr, lun, 'big.txt', /get_lun
```

```
> fs = fstat(lun)
```

```
> buf = bytarr(fs.size)
```

```
> readu, lun, buf
```

```
> free_lun, lun
```

```
> buf = strsplit(string(buf), string([13b,10b]), /extract) ; split on CR
```

```
> +LF
```

I suggest to use the following code:

```
; -----
```

```
FUNCTION ReadUnformattedAscii, FileName
```

```
  COMPILE_OPT IDL2
```

```
  IF ~FILE_TEST(FileName) THEN RETURN, 0
```

```
; # of line
```

```
  nLines = FILE_LINES(FileName)
```

```

OPENR,Lun,FileName,/Get_Lun
; Read First Line to establish # of Values inside the record
dummy = "
READF,Lun,Dummy
POINT_LUN,Lun,0
Dummy = STRCOMPRESS(dummy)
void = STRSPLIT(dummy,Count=nFields)
nValues = NFields-1
; Define my output structure
str_out = {Text:", $
           Values: FLTARR(nValues)}
str_out = REPLICATE(str_out,nLines)
; Define Tmp Vars
TmpArr = FLTARR(nValues)
TmpStr = "
FOR j = 0, nlines-1 DO BEGIN
  READF,Lun,Dummy
  dummy = STRCOMPRESS(dummy)
  IText = STRPOS(dummy,' ')
  rForm = '(A'+STRCOMPRESS(STRING(IText))+','+$
          STRCOMPRESS(STRING(nValues))+F0)'
  READS,dummy,TmpStr,TmpArr,FORMAT=rForm
  str_out[j].Text = TmpStr
  Str_out[j].Values = TmpArr
ENDFOR
FREE_Lun,Lun
RETURN,str_out
END
; -----

```

Subject: Re: Reading a set of data with string type entries

Posted by [rkombiyil](#) on Fri, 30 Nov 2007 12:53:32 GMT

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

On Nov 30, 2:44 pm, mystea <idllear...@gmail.com> wrote:

```

> Dear all,
>
> I need to read in a large set of data. Each entry is started with a
> name, followed by many numbers.
> The numbers are separated by spaces, but the length of each number is
> not fixed.
>
> My problem is like this: Since the data starts with a string, I can't
> use free-format readf (because by default, the string slot will read
> the whole line.) and due to the unknown length of numbers, I can't use
> a format code, either.

```

>
> What should I do?
>
> Gene

Don't know if one has to resort to regex... I suggest using "structures" to read variable length data as "string variables and specify format as floats". I use this method to read satellite data where the length of the "same datum" is "not fixed."

For example, I have used the following and worked for me. The "a" format is where the variable length numbers are ---> so I counted the total number of spaces from the end of one datum to end of, and including the next datum, which is variable - and this space is "fixed" (this fixed space is specified by the "a" format, and when you declare them as floats, you get the variable length datum only, minus the empty spaces). So count the space separating and including the variable length datum :)

fmt='(2(i2,1x),i4,2(1x,i2),7x,2(a15),a16)'
followed by the record structure
record={dd:0L,mo:0L,yyyy:0L,hh:0L,mm:0L,AA:0.0,BB:0.0,CC:0.0 }

Then the whole datafile consisting of various records can be read, each with the same format as above.

Hope this helps,
/rk

Subject: Re: Reading a set of data with string type entries
Posted by [Peter Clinch](#) on Fri, 30 Nov 2007 13:08:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

mystea wrote:
> I need to read in a large set of data. Each entry is started with a
> name, followed by many numbers.
> The numbers are separated by spaces, but the length of each number is
> not fixed.
>
> My problem is like this: Since the data starts with a string, I can't
> use free-format readf (because by default, the string slot will read
> the whole line.) and due to the unknown length of numbers, I can't use
> a format code, either.
>
> What should I do?

In this sort of situation I usually write a quick hack in Perl to turn

the input file into something that's easy to process. Perl deals with any text you can find pretty much /any/ pattern in remarkably well and it's often easier to pre-process that and leave IDL for the pictures and number crunching.

(Also very handy for going through verbose data files and binning all the shite you're not interested in before you read it into IDL)

Pete.

--

Peter Clinch Medical Physics IT Officer
Tel 44 1382 660111 ext. 33637 Univ. of Dundee, Ninewells Hospital
Fax 44 1382 640177 Dundee DD1 9SY Scotland UK
net p.j.clinch@dundee.ac.uk <http://www.dundee.ac.uk/~pjclinch/>

Subject: Re: Reading a set of data with string type entries

Posted by [lasse](#) on Sat, 01 Dec 2007 16:16:08 GMT

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

On 30 Nov, 05:44, mystea <idllear...@gmail.com> wrote:

> Dear all,
>
> I need to read in a large set of data. Each entry is started with a
> name, followed by many numbers.
> The numbers are separated by spaces, but the length of each number is
> not fixed.
>
> My problem is like this: Since the data starts with a string, I can't
> use free-format readf (because by default, the string slot will read
> the whole line.) and due to the unknown length of numbers, I can't use
> a format code, either.
>
> What should I do?
>
> Gene

Hi,

am I the only one who uses READ_ASCII and ASCII_TEMPLATE in these cases?

Admittedly, READ_ASCII without having made a template beforehand is horrible. But you can make one, save it and then restore it just before using READ_ASCII.

Cheers
Lasse Clausen

Subject: Re: Reading a set of data with string type entries
Posted by [David Fanning](#) on Sat, 01 Dec 2007 16:28:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

Lasse Clausen writes:

> am I the only one who uses READ_ASCII and ASCII_TEMPLATE in these
> cases?

I think so. It's too slow for everyone else. :-)

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.")
