Subject: Re: Format strings with comma separated lists Posted by davidf on Mon, 10 May 1999 07:00:00 GMT

View Forum Message <> Reply to Message

William Daffer (vapuser@catspaw.jpl.nasa.gov) writes:

- > Apparently, the short answer is you don't need to use format strings
- > on input, provided, I guess, the data is homogeneous and with a
- > constant number of columns(?); IDL understands that you don't want the
- > comma.

Uh, this has been a feature of IDL since...well, since version 1.0, I believe. ;-)

Cheers,

David

P.S. Ain't IDL grand!? Hardly a week goes by I don't learn something new about it. :-)

--

David Fanning, Ph.D. Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com

Covote's Guide to IDL Programming: http://www.dfanning.com/

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Format strings with comma separated lists Posted by Vapuser on Mon, 10 May 1999 07:00:00 GMT

View Forum Message <> Reply to Message

Well, am I'm surprised!

Just for a lark, I tried this

IDL> openw,1,'/tmp/delme.junk'
IDL> v=findgen(20)
IDL> printf,1,v,format='(4(f5.2,:,","))'
(NB, this comes out looking like
0.00, 1.00, 2.00, 3.00,
4.00, 5.00, 6.00, 7.00,
8.00, 9.00,10.00,11.00,
12.00,13.00,14.00,15.00,
16.00,17.00,18.00,19.00)

IDL> close,1

```
IDL> openr,1,'/tmp/delme.junk'
IDL > v[*] = 0
IDL> print,minmax(v)
   0.00000
               0.00000
IDL > v = reform(v, 4, 5)
IDL> help,v
V
          FLOAT
                    = Array[4, 5]
IDL> readf,1,v
IDL> print, v
   0.00000
               1.00000
                          2.00000
                                      3.00000
   4.00000
               5.00000
                          6.00000
                                      7.00000
               9.00000
                           10.0000
                                      11.0000
   8.00000
                          14.0000
                                      15.0000
   12.0000
               13.0000
   16.0000
               17.0000
                           18.0000
                                      19.0000
 ; Or, you can read it into a vector
IDL > v = fltarr(20)
IDL> point lun,1,0
IDL> readf,1,v
IDL> print,v
   0.00000
                          2.00000
                                      3.00000
                                                  4.00000
                                                              5.00000
               1.00000
   6.00000
               7.00000
                          8.00000
                                      9.00000
                                                  10.0000
                                                              11.0000
               13.0000
                           14.0000
                                      15.0000
                                                  16.0000
                                                              17.0000
   12.0000
   18.0000
               19.0000
IDL>
```

Apparently, the short answer is you don't need to use format strings on input, provided, I guess, the data is homogeneous and with a constant number of columns(?); IDL understands that you don't want the comma.

You can even read the data if there isn't a constant number of columns. I edited the file so that it looked like;

```
0.00, 1.00, 2.00,
  4.00, 5.00, 6.00, 7.00,
  8.00, 9.00, 10.00, 11.00,
  12.00,13.00,14.00,15.00,
  16.00,17.00,18.00,19.00
 and read it with;
IDL> v=fltarr(19)
IDL> readf,1,v
IDL> print,v
   0.00000
                           2.00000
                                       4.00000
                                                   5.00000
               1.00000
                                                               6.00000
   7.00000
               8.00000
                           9.00000
                                       10.0000
                                                   11.0000
                                                               12.0000
```

```
13.0000
               14.0000
                           15.0000
                                       16.0000
                                                   17.0000
                                                               18.0000
   19.0000
IDL>
 I was going to suggest a combination of string reads and
STR_SEP. Doesn't look like you need it.
(By the was, this is a
 IDL> print,!version
 { mipseb IRIX unix 5.1.1 Jul 20 1998} system
William
philaldis@geocities.com (Phil Aldis) writes:
> I hoping that someone else understands format strings better than I
> do, because I can't get them to work at all. I was wondering if anyone
> knew how to read a comma delimited list using format strings, such as
>
 Readf, lun, variable, FORMAT='(n(F6.6, :, ","))'
  ...this is the way I would think that you should do it but this just
  tells me that end of input record encountered.
  Does anyone know how to do it?#
>
> Cheers.
> Phil Aldis
William Daffer: 818-354-0161: vapuser@catspaw.jpl.nasa.gov
```

Subject: Re: Format strings with comma separated lists Posted by gurman on Mon, 10 May 1999 07:00:00 GMT View Forum Message <> Reply to Message

In article <3736bb88.14638556@146.80.9.44>, philaldis@geocities.com wrote:

```
    I hoping that someone else understands format strings better than I
    do, because I can't get them to work at all. I was wondering if anyone
    knew how to read a comma delimited list using format strings, such as
    Readf, lun, variable, FORMAT='(n(F6.6, :, ","))'
```

- > ...this is the way I would think that you should do it but this just
- > tells me that end of input record encountered.

>

> Does anyone know how to do it?#

>

- > Cheers,
- > Phil Aldis

Phil -

I believe this takes someone old enough to have learned programming first in FORTRAN....

For formatted reads, you basically want to include only variable descriptors and spaces (nX), so the colon and "," are counterproductive.

Have you tried something of the form:

IDL> READF, lun, variable, FORMAT = $\frac{(n(f6.6, 2x))}{}$

That is, you just want to skip the commas and read the floating-point variables.

Hope this is what you're looking for.

Joe Gurman

--

Joseph B. Gurman / NASA Goddard Space Flight Center / Solar Physics Branch / Greenbelt MD 20771 / work: gurman@gsfc.nasa.gov /other: gurman@ari.net

Government employees are still not allowed to hold opinions while at work, so any opinions expressed herein must be someone else's.

Subject: Re: Format strings with comma separated lists Posted by Martin Schultz on Tue, 11 May 1999 07:00:00 GMT View Forum Message <> Reply to Message

Vapuser wrote:

>

> Well, am I'm surprised!

>

- > [...]
- > Apparently, the short answer is you don't need to use format strings
- > on input, provided, I guess, the data is homogeneous and with a
- > constant number of columns(?); IDL understands that you don't want the
- > comma.

Here is an example of why I am really glad that IDL does support format strings:

23.4 12.2 12.1 10.0 56.4 98.4103.2143.9 200.2232.8165.1 99.7

AND believe it or not: those files exist and there are plenty of them ;-)

Also: format strings are probably the easiest way to (safely) convert a file that is written in e.g. 60 lines with 8 numbers per line to an array

of (40,12) [it's just one statement!]

Martin.

puhuu. didn't know that i am that old - i know something about format!

--

Martin Schultz, DEAS, Harvard University, 29 Oxford St., Pierce 109, Cambridge, MA 02138 phone (617) 496 8318 fax (617) 495 4551 e-mail mgs@io.harvard.edu web http://www-as/people/staff/mgs/