

---

Subject: Re: Reading in text data

Posted by [Paul van Delst](#) on Wed, 09 Aug 2000 07:00:00 GMT

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

---

> Brian Reardon (reardonb@my-deja.com) writes:

>

>> I am reading in text data (columns and rows of numbers) and I would  
>> like to know if there is a more elegant way of doing it. Currently, the  
>> user must specify how many columns there are. In my case the number of  
>> columns is manually inserted into the first line of the file like this:

>>

>> 3

>> 0 1 2

>> 1 2 3

>> 2 3 4

>> 3 4 5

>> 4 5 6

>> 5 6 7

>> 6 7 8

>> 7 8 9

>> 8 9 10

>> 9 10 11

>>

>> The attached procedure reads in the data. Is there a way to read in the  
>> data such that the user does not have to a priori know how many columns  
>> there are and such that IDL does not have to reserve a large amount of  
>> memory for the number of rows?

>

Wot about DDREAD.PRO (and associated routines) by F.K.Knight? I use it  
all the time. It allows you skip row, columns so the first line being a  
single number shouldn't matter.

Check out

[http://www.astro.washington.edu/deutsch/idl/htmlhelp/library 38.html](http://www.astro.washington.edu/deutsch/idl/htmlhelp/library%2038.html)

where you'll find:

Routine Descriptions

DDREAD

[\[Next Routine\]](#) [\[List of Routines\]](#)

Name:

ddread

Purpose:

This routine reads data in formatted (or unformatted) rows and columns.

The name stands for Data Dump Read. By default, comments are skipped and the number of columns is sensed. Many options exist, e.g., selecting rows and columns, reading binary data, and selecting non-default data type and delimiters.

Examples:

```
junk = ddread(/help)           ; get information only
array = ddread(file)           ; read ASCII data
array = ddread(file,/formatted) ; ditto
array = ddread(file,object=object) ; read binary data
array = ddread(file,columns=[0,3]) ; get only 1st & 4th
```

columns

```
array = ddread(file,rows=lindgen(10)+10); get only 2nd 10 rows
array = ddread(file,offset=10,last=19) ; get rows (10,19)
array = ddread(file,/countall)         ; count comment lines
array = ddread(file,/verbose)          ; echo comment lines
array = ddread(file,type=1)            ; return bytes, not
```

floats or longs

```
array = ddread(file,range=['start text','stop text']) ; text
```

delimiters

```
 ; Place the detailed output from a Lowtran run in a 2-D
```

array---wow!

```
output = ddread('lowtran.out',range=['(CM-1)
(MICRN)','0INTEGRATED ABSORPTION'])
% DDREAD: Read 69 data lines selecting 14 of 14 columns; skipped
395 comment lines.
```

Usage:

```
array = ddread([file],[,options][,/help])
```

Optional Inputs:

file = file with data; if omitted, then call pickfile.

Keywords:

/formatted, /unformatted = flags to tell IDL whether data format

is

binary or ASCII. ddread tries to determine the type of data but it's not foolproof.

object = a string containing the IDL declaration for one instance

of the object in an unformatted file, e.g.,  
'fltarr(4)'

or

'{struct,dwell:0.,pitch:0.,yaw:0.,roll:0.}'

rows = an array to select a subset of the rows in a formatted file

Does not count comment lines, unless /countallrows is set!

columns = likewise for columns

type = data type of the output D=float (if '.' appears) or long

delimiter = column separator, D=whitespace

/help = flag to print header

range = start and stop row or strings,  
e.g. range = ['substring in 1st line', 'substring in last line']

offset = start row (read to end of file, unless last set)

last = stop row (read from start of file, unless offset set)

/countallrows = flag to count comment rows as well as data rows (D=0)

/verbose = flag to echo comments to screen

Outputs:  
array = array of data from the lines (ASCII) or objects (binary)

Common blocks:  
none

Procedure:  
After deciding on ASCII or binary, read file and return array.

#### Restrictions:

- Comments can be either an entire line or else an end of a line, e.g.,

```
/* C comment. */
; IDL comment
Arbitrary text as a comment
Comment in Fortran
The next line establishes # of columns (4) & data type
```

(float):

```
6. 7 8 9
This line and the next are both considered comments.
6 comment because only one of 4 columns appears
1 2 3 4 but this line has valid data and will be read as
```

data

- Even if a range of lines is selected with offset, range or last, all lines are read. This could be avoided.

- Other routines needed:

```
pickfile.pro - to choose file if none is given
nlines.pro   - to count lines in a file
nbytes.pro   - to count bytes in a variable
replicas.pro - to replicate arrays (not scalars as in
```

replicate.pro)

typeof.pro - to obtain the type of a variable

Modification history:

write, 22-26 Feb 92, F.K.Knight (knight@ll.mit.edu)

allow reading with arbitrary delimiter using reads, 23 Mar 92,

FKK

add countallrows keyword and modify loop to read as little  
data as possible, 20 May 92, FKK

correct bug if /formatted set, 6 Jul 92, FKK

add verbose keyword to print comments, 6 July 92, FKK

correct bug if /rows=.../countall set, 6 July 92, FKK & EJA

add a guard against a blank line being converted to a  
number, 21 Aug 92, FKK

allow parital line just before the EOF. Possibly this isn't the  
right thing to do, but I decided to allow it. If the final

line

is incomplete, the values are still read and the remainder of  
the line is filled with zeroes. 26 Oct 92, FKK

allow range keyword to be a string array, 2 Dec 92, FKK

make default for countallrows be true if range is present, 2 Dec  
92, FKK

add new function (typeof); called in a few places, 2 Dec 92, FKK

--

Paul van Delst            Ph: (301) 763-8000 x7274

CIMSS @ NOAA/NCEP       Fax: (301) 763-8545

Rm.202, 5200 Auth Rd.   Email: pvandelst@ncep.noaa.gov

Camp Springs MD 20746

---