Subject: Re: Q: About reading files into an array without knowing the size.
Posted by Pierre Maxted on Mon, 06 Nov 1995 08:00:00 GMT
View Forum Message <> Reply to Message

David van Kuijk <kuijk@mpi.nl> wrote:
> Hi
>
> One of the nice things of IDL is that it is possible to read whole
> ASCII-files of data (e.g. floats) into an array in one swoop, without
> having to go through a while loop which reads all of these numbers one by
> one. E.g.:
>
> OPENR,1, "filename"
> floatss=FLTARR(10000)
> READF, 1, floatss
>
> What is not so nice is that IDL has to know exactly how many datapoints
> there are in the file, otherwise not all the data are read, or you get
> sth like an "End of file encountered"-error. So the size of _floatss_ in
> the example above should be equal to the number of floats in the file.
>
> I know that Mathlab is capable of processing files with an unknown number
> of data.
>
> Does anybody know a way to achieve this in IDL (maybe I missed something)?
>
> David,
>
> ********** ********** **********
> * David van Kuijk | Max-Planck-Institute for Psycholinguistics *
> * E-mail: kuijk@mpi.nl | Wundtlaan 1 *
> * | 6525 XD Nijmegen *
> * | The Netherlands *
> * Tel: +31 (0)24 3521523 | *
> * Fax: +31 (0)24 3521213 | Snail-mail: P.O. Box 310, 6500 AH Nijmegen *
> * "Prots the whoblem?" *
> ********** ********** **********
>

This routine, due to Michael Andersen in Copenhagen, should do the trick if you are on a UNIX machine. The trick here is to use the command "wc" to get the number of lines and "words" in the file. I'm sure it can be converted to VMS without to much trouble.

Included below are the function itself (READ_FLTARR) that reads in the array and below that is the function ACCESS that checks that the file exists.
--
_-_-_ _-_-_

| Dr Pierre Maxted (pflm@star.maps.susx.ac.uk) |
| | Astronomy Centre, University of Sussex | |
| Falmer, Brighton, BN1 9QH |
-_-_- Procrastinate now!!! -_-_-

```
FUNCTION READ_FLTARR, FILE , COLLS , LINES , SILENT = silent
;+
; NAME:
;       READ_FLTARR
;
; PURPOSE:
;       read a 2D float array from an ascii file
;
; CALLING SEQUENCE:
;       array = READ_FLTARR( FILE [ , COLLS , LINES , /SILENT ] )
;
; INPUTS:
;       FILE    String giving file from which to read array
;
; KEYWORDS;       SILENT  If set, the size of ARRAY will not be displayed
;
; OUTPUTS:
;       ARRAY   Float array
;
; OPTIONAL OUTPUTS
;       COLLS   Number of collumns in ARRAY
;
;       LINES   Number of lines in ARRAY
;
; RESTRICTIONS
;       files with header or empty lines cannot be handled
;       uses non standard routine 'ACCESS'
;
; PROCEDURE
;       Uses SPAWN and UNIX 'wc' to establish size of array
;
; MODIFICATION HISTORY:
;       WRITTEN, Michael Andersen CUOBS, August, 1994
;-

  ON_ERROR, 2

  IF NOT ( ACCESS( file ) ) THEN $
  MESSAGE, 'Cannot access file:' + file
```

```
  SPAWN, "wc -lw " + file + $
         " | sed s/" + file + "//" + $
         " | awk '"+'{printf("%s\n%s\n",$1,$2)}'+"'", lw
  lines = LONG( lw( 0 ) )
  colls = LONG( lw( 1 ) ) / lines
  IF ( lw( 0 ) eq 0 ) THEN MESSAGE, 'File with no lines:' + file

  array = fltarr( colls , lines )
  IF( NOT KEYWORD_SET( SILENT ) ) THEN BEGIN
    PRINT, 'Reading array of ' + STRTRIM( colls ) + ' columns'
    PRINT, '             and ' + STRTRIM( lines ) + ' lines'
  ENDIF

  OPENR, l , file , /get
  READF, l , array
  CLOSE, l & FREE_LUN, l

  RETURN, array

END



 ------------END OF READ_FLTARR---------------


FUNCTION ACCESS, file , UNIQ = uniq

;+
;NAME:
;       ACCESS
;
;PURPOSE:
;       Boolean function returning true, if file exist
;
;CATEGORY:
;       Files, IO
;
;CALLING SEQUENCE:
;       a = ACCESS( file [, UNIQUE = unique ] )
;
;INPUTS:
;       file    Scalar string, giving the name of the file to be searched fore
;
;OPTIONAL KEYWORDS
;       UNIQ,   if set, ACCESS returns true only if file is unique
;
;OUTPUTS:
```

```
      a       boolean, true if file exsist, else false
;
;COMMON BLOCKS:
;       none
;
;SIDE EFFECTS:
;       none
;
;RESTRICTIONS:
;       only a single file can be handled at a time
;
;MODIFICATION HISTORY:
;       Written, Michael Andersen, December 1992
;-


  ON_ERROR, 2                       ;return to caller

  IF( STRLEN( file ) EQ 0 ) THEN RETURN, 0
  a = FINDFILE( file , COUNT = nfiles )
  IF( nfiles EQ 1 OR nfiles GT 1 AND NOT KEYWORD_SET( UNIQUE ) ) THEN RETURN, 1
  RETURN, 0

END
```