
Subject: Re: Printf and line breaks

Posted by [T Bowers](#) on Mon, 22 Jan 2001 14:49:52 GMT

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

Hey Alan, try this. This function will write an array to a text file as it is formatted in memory. In your case, you want 10 floats on 1 line so:

```
;//create your array to have same format as you want written to the file
dataArray = findgen(10)
```

```
;//write it to a text file
openw, outFile, "out.txt", /get_lun
retCode = tbFWriteTData(outFile, dataArray, SEP="TAB")
close, outFile
```

Now they'll all write to a single, tab-separated line.
If you reformed the array to a [2,5] array, they'd print out that way instead: 2 columns of 5 rows.

```
dataArray= reform(dataArray,2,5)
openw, outFile, "out.txt", /get_lun
retCode = tbFWriteTData(outFile, dataArray, SEP='2A'xb)
close, outFile
```

This way, the data was asterisk (*) separated because I specified the sep to be 2A hexadecimal, the ascii code for *. It'll write as:

```
0.000000*1.00000
2.00000*3.00000
4.00000*5.00000
6.00000*7.00000
8.00000*9.00000
```

WARNING: This was written a while back when I 1st started using IDL. I've never had any problems with it and use it alot but no guarantees! And no laughing at my sloppy/inefficient coding!!;)

good luck,
todd bowers

----cut here----

```
;//////////////////////////////////////
```

```
function tbFWriteTData, outFile, data, SEP=sep
;Fn to write data to tab (by default) delimited text files so the data is
; formatted as the array is in memory.
; Eg, if arr = fltarr(100,5), then tbFWriteTData(outFile, dataArray,
```

```

SEP="TAB")
; will write the data to file in 5 rows of 100 tab separated values.
; If arr is a string, it'll write it out as is, so note that if you have a
; string with spaces between character values and you want this to be
output
; as tab separated values, then YOU should str_sep 1st to get it to an
array.
; NOTE: only 1D (including strings, which for some reason are categorized
as 0
; dimensional (i.e. scalar) data) and 2D arrays supported.

```

```

;9/24/98 Todd Bowers

```

```

; //ASCII values for some common separators
CRLF = string('0D'xb) + string('0A'xb)
TAB = string('09'xb)
SPACE = string('20'xb)

```

```

if n_elements(sep) EQ 0 then $ ;defaults to TAB
  SEPARATOR = TAB $
else if (strupcase(sep) EQ "TAB" OR strupcase(sep[0]) EQ "T") then $
  SEPARATOR = TAB $
else if (strupcase(sep) EQ "SPACE" OR strupcase(sep[0]) EQ "S") then $
  SEPARATOR = SPACE $
else if (strupcase(sep) EQ "CRLF") then $
  SEPARATOR = CRLF $
else $ ;//or U can pass the ASCII value of the sep yourself! Any will do.
  SEPARATOR = string(sep)

```

```

sizeData = size(data)
numDims = sizeData[0]
if numDims EQ 0 then begin
  ;//numDims = 0? .See if it's a string...
  ;// (BTW, shouldn't a string be a 1 dim array?)
  str = "" & sz_str = size(str) & STRINGTYPE = sz_str[1]
  if sizeData[1] EQ STRINGTYPE then begin
    ;data = str_sep(data, " ",/TRIM)
    numCols = n_elements(data)
    numRows = 1
  endif $
  ;//Not a string? Then must be a scalar w/ 1 row 'n 1 col
  else begin
    numCols = 1
    numRows = 1
  endelse
endif $
; //Ok, not a string or scalar, see if 1D row array (string or numeric)
else if numDims EQ 1 then begin

```

```

    numCols = sizeData[1]
    numRows = 1
endif $
; //Must be a 2D array then.
else begin
    numCols = sizeData[1]
    numRows = sizeData[2]
endelse

; //Go ahead and convert data to char's so I don't do it for every write
below...
stringData = strtrim(string(data[*,*]),2)

for j = 0, (numRows - 1) do begin
    for i = 0, (numCols - 1) do begin
        writeu, outFile, (stringData[i,j] + SEPERATOR)
    endfor
    if SEPERATOR NE CRLF then begin
        point_lun, -outFile, fpos
        point_lun, outFile, (fpos - 1)
        writeu, outFile, CRLF
    endif
endfor

return, 1
end;function tbFWriteTData
;/////////////////////////
----cut here----

```

"ALAN FRAZIER" <s007amf@news.wright.edu> wrote in message
news:<94a5ur\$qsfs1@mercury.wright.edu>...

> Sorry to bother everyone again. I am just beginning my enduring
journey
of
> learning IDL and keep running into obstacles.
>
> My current problem is with printf. I am trying to create a text
output
> file that can be imported into a spreadsheet. My problem is that
printf is
> automatically line wrapping. For example:
>
> printf, fileptr, val1, val2, val3, val4, val5, val6,
>
> is producing output where val1, val2, val3, and val4 are on a single
line.
> But, val5 and val6 are placed on the following line. NOTE: I am
actually

> trying to print 10 different floats on the same line.
>
> I did think about converting all the values into one long string and
> printing that string. But, this issue was still a problem.
>
> Anyone have any ideas about how to fix this? Is a solution even
possible?
>
>
> Thanks,
>
>
> Alan
>
