Subject: Getting memory overflow on array concat; why?
Posted by bleau on Tue, 24 Jan 1995 20:37:42 GMT
View Forum Message <> Reply to Message

Hello. I'm a fairly new user of IDL; so is a user of mine, on whose behalf I am posting this message. We are using IDL V2.0 on OpenVMS VAX.

The task is to read in a set of numbers from a disk file into a 2-dimensional array. The disk file has 3 numbers per line, about 4000 lines. I include at the end of this post the IDL code which does the read. Problem is, the code never finishes; IDL runs out of memory beforehand. I increased the user's max memory allotment, and the program went a little further before running out of memory. Back-of-envelope calculations showed the user needs about 48Kb to hold this data, of 93 pages (512 bytes/page). IDL itself uses ~6000 pages. I gave her account an upper limit of 25000 pages; enough room for IDL and a 810666x3 array! What is going on???

I suspect the array concatenation operator is the culprit, but I can't say why. Here's the relevant code fragment, which is within a loop:

```
    READF, 1, temp
    IF first THEN BEGIN
      data = temp
      first = ''
    ENDIF ELSE BEGIN
      data = [data,temp]
    ENDELSE
```

Both data and temp have been previously declared FLTARR(1,parameters), where parameters is 3. The help command, when given after the crash, shows DATA as being something like (3100,3), and only 2% of the symbol space used.

As I said, this doesn't make sense. My hunch is there's a side effect of array concatenation that eating up memory, never releasing it. Has anyone has experience with this type of programming bug and how to fix it?

Larry Bleau
University of Maryland
bleau@umdsp.umd.edu
301-405-6223

Complete IDL code:

```
;    this IDL program is for quick and dirty plotting
;    it opens a file, reads a given number of parameters and plots any number
;    of parameters versus one parameter
filename = ' '
PRINT, ' Enter name of data file'
READ, filename
PRINT, ' Enter number of parameters to read'
```

```
READ, parameters
data = FLTARR(1,parameters)
temp = FLTARR(1,parameters)
OPENR, 1, filename
first = 'first'
ret = ''
WHILE NOT EOF(1) DO BEGIN
;&
    temp_ret = size(data)
    if temp_ret(1) mod 1000 eq 0 then begin
    PRINT, ' ENTER RETURN'
    READ, ret
    endif
;&
    ON_IOERROR, go_on
    READF, 1, temp
    IF first THEN BEGIN
      data = temp
      first = ''
    ENDIF ELSE BEGIN
      data = [data,temp]
    ENDELSE
    go_on:
ENDWHILE
CLOSE, 1
;                                       get plot information
plot_start:
xlinlog = ' '
ylinlog = ' '
PRINT, '        Plot information'
PRINT, 'Enter log for logarithmic or lin for linear x axis '
READ, xlinlog
PRINT, 'Enter log for logarithmic or lin for linear y axis '
READ, ylinlog
PRINT, ''
PRINT, 'Enter the number of parameters to plot '
READ, plot_param
y_param = INTARR(plot_param)
PRINT, 'Enter parameter number for the x axis '
READ, x_param
x_param = x_param - 1
y_min = 1.e10
y_max = 0.
FOR i = 0, plot_param-1 DO BEGIN
    PRINT, 'Enter parameter number to plot '
    READ, temp1
    y_param(i) = temp1 - 1
    y_min = y_min < MIN(data(*,temp1-1))
```

```
    y_max = y_max > MAX(data(*,temp1-1))
ENDFOR
;                                       create plot
case_start:
CASE 1 OF
    xlinlog EQ 'log' AND ylinlog EQ 'log' : BEGIN
      PLOT_OO, data(*,x_param), data(*,y_param(0)), $
        YRANGE=[y_min, y_max]
    END
    xlinlog EQ 'log' AND ylinlog EQ 'lin' : BEGIN
      PLOT_OI, data(*,x_param), data(*,y_param(0)), $
       YRANGE=[y_min, y_max]
    END
    xlinlog EQ 'lin' AND ylinlog EQ 'log' : BEGIN
      PLOT_IO, data(*,x_param), data(*,y_param(0)), $
       YRANGE=[y_min, y_max]
    END
    xlinlog EQ 'lin' AND ylinlog EQ 'lin' : BEGIN
      PLOT, data(*,x_param), data(*,y_param(0)), $
       YRANGE=[y_min, y_max]
    END
    ELSE : BEGIN
      PRINT, ' Make sure lin and log entries are in lower case'
      PRINT, 'Enter log for logarithmic or lin for linear x axis '
      READ, xlinlog
      PRINT, 'Enter log for logarithmic or lin for linear y axis '
      READ, ylinlog
      GOTO, case_start
    END
ENDCASE
FOR i = 1, plot_param-1 DO OPLOT, data(*,x_param), data(*,y_param(i))
;                                       make another plot?
answer = ' '
PRINT, ' Do you want to make another plot?'
READ, answer
IF answer EQ 'Y' OR answer EQ 'y' THEN GOTO, plot_start
END
```

Larry Bleau
University of Maryland
bleau@umdsp.umd.edu
301-405-6223

Subject: Re: Getting memory overflow on array concat; wh
Posted by knipp on Sat, 28 Jan 1995 13:35:24 GMT

In article r43@umd5.umd.edu, bleau@umdsp.umd.edu writes:
> Hello. I'm a fairly new user of IDL; so is a user of mine, on whose behalf I
> am posting this message. We are using IDL V2.0 on OpenVMS VAX.
>
> The task is to read in a set of numbers from a disk file into a 2-dimensional
> array. The disk file has 3 numbers per line, about 4000 lines. I include at
> the end of this post the IDL code which does the read. Problem is, the code
> never finishes; IDL runs out of memory beforehand. I increased the user's max
> memory allotment, and the program went a little further before running out of
> memory. Back-of-envelope calculations showed the user needs about 48Kb to hold
> this data, of 93 pages (512 bytes/page). IDL itself uses ~6000 pages. I gave
> her account an upper limit of 25000 pages; enough room for IDL and a 810666x3
> array! What is going on???
>
> I suspect the array concatenation operator is the culprit, but I can't say why.
> Here's the relevant code fragment, which is within a loop:
>   READF, 1, temp
>   IF first THEN BEGIN
>   data = temp
>   first = ''
>   ENDIF ELSE BEGIN
>   data = [data,temp]
>   ENDELSE
> Both data and temp have been previously declared FLTARR(1,parameters), where
> parameters is 3. The help command, when given after the crash, shows DATA as
> being something like (3100,3), and only 2% of the symbol space used.
>
> As I said, this doesn't make sense. My hunch is there's a side effect of array
> concatenation that eating up memory, never releasing it. Has anyone has
> experience with this type of programming bug and how to fix it?
>
> Larry Bleau
> University of Maryland
> bleau@umdsp.umd.edu
> 301-405-6223
>
... (stuff deletet)

>
> Larry Bleau
> University of Maryland
> bleau@umdsp.umd.edu
> 301-405-6223

Your idea about "eating up memory, never releasing it" is to my understanding
correct. You may overcome your problem by

- define array DAT=FLTARR(1,n_lines)
- read in using a FOR - loop:
 FOR i=0, n_lines-1l do
  READF, unit, tmp
  DAT(*,i) = tmp
 ENDFOR

- if you do not know the number of lines in your data-file, and want IDL
  to get that number

 - open the file
 - get the filesize via FSTAT
 - define BFILE = bytarr(file-size)
 - readu, unit, bytarr
 - count number of linefeeds (pos = where(BFILE eq 10b))
  n_lines = n_elements(pos) (if pos(0) ne -1)
 - close file
 - set BFILE=0 (relesing memory !!)
 - define your data-array

Hop this helps

 _____
_____

   __ ____ __
    / // _ \ / / Karlheinz Knipp   phone: +49 511 - 762 4922
   / // /_/ // /   University of Hannover   fax: +49 511 - 762 2483
  / // ____// /   Institute for Photogrammetry
 / // /   / /   Nienburger Str.1
/_//_/   /_/   FRG 30167 Hannover   e-mail: knipp@ipi.uni-hannover.de