
Subject: Re: Putting bytes into structures
Posted by [Marty Ryba](#) on Thu, 23 Oct 1997 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

This is a multi-part message in MIME format.

-----2BE345064975
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit

Marty Ryba wrote:

> Brian Jackel wrote:
>> What's the best way to load a given number of bytes into a structure?
> Here's a routine called READB.PRO that does exactly what you want.
> It recursively descends structures and places the bytes into them.

Oops, I noticed it uses a routine called NBYTES.PRO; here it is.
Also, both routines need updating to handle type DCOMPLEX for IDL 5.0

--
Dr. Marty Ryba | Of course nothing I say here is official
MIT Lincoln Laboratory | policy, and Laboratory affiliation is
ryba@ll.mit.edu | for identification purposes only,
| blah, blah, blah, ...

-----2BE345064975
Content-Type: text/plain; charset=us-ascii; name="nbytes.pro"
Content-Transfer-Encoding: 7bit
Content-Disposition: inline; filename="nbytes.pro"

```
;+
; Name:
; nbytes
; Purpose:
; Return the number of bytes in the variable
; Usage:
; nb = nbytes(variable)
; Inputs:
; variable = any IDL variable
; Optional Inputs or Keywords:
; help = flag to print header
; Outputs:
; nb = number of bytes in variable
; Common blocks:
; none
; Procedure:
; Idea from David Stern.
; Modification history:
```

```

; write, 22 Feb 92, F.K.Knight
; increase speed by writing to disk only for structures, 10 Sep 92, FKK
; eliminate Unix-specific file (from ali@rsinc.com), 11 Sep 92, FKK
;-
function nbytes,variable,help=help
;
;
; =====>> HELP
;
on_error,2
if keyword_set(help) then begin & doc_library,'nbytes' & return,-1 & endif
;
; =====>> CHOOSE OPTION BASED ON TYPE OF VARIABLE
;
sz = size(variable)
type = sz(sz(0)+1)
nelements = sz(sz(0)+2)
case type of
0:return,0 ; UNDEFINED
1:return,nelements ; BYTE
2:return,nelements*2 ; INT
3:return,nelements*4 ; LONG
4:return,nelements*4 ; FLOAT
5:return,nelements*8 ; DOUBLE
6:return,nelements*8 ; COMPLEX
7:return,long(total(strlen(variable))) ; STRING: DOESN'T COUNT NULLS
8:begin ; STRUCTURE: COMPILER-SPECIFIC LENGTH
file = filepath('IDL',/tmp)
openw,lun,file,/get_lun,/delete
writeu,lun,variable(0) ; SO WRITE 1ST ELEMENT TO FILE
stat = fstat(lun)
close,lun
free_lun,lun
return,stat.size*nelements ; AND RETURN TOTAL LENGTH
end
else: message, 'unknown type = '+strtrim(type,2)
endcase
end

```

-----2BE345064975--
