Subject: Re: Structure in argument?
Posted by Martin Schultz on Sun, 22 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Skiso wrote:

> Hi,
>
> I need to pass a lot of parameters to a fonction, is the structure the
> best way to do it?
> I don't want to use the confusing common block.
> Does the use of structure can cause a problem?
>
> Thanks
>
> Skiso

Structure or keywords are both fine, depending on how variable your arguments are: if you often specify only a tiny fraction of your arguments, keywords may be easier to use. In both cases you should make sure that you initialize your values properly. For keywords this usually works like:

```
    if (n_elements(KEYWORD1) eq 0) then keyword1 = 0.
   ; or boolean
    keyword1 = keyword_set(KEYWORD1)
```

for structures, I would recommend my chkstru function (attached below) that tests (1) that the argument you pass really is a structure, and (2) that the tag names that you need are really contained in that structure:
e.g.:

```
   if (not chkstru(ARGUMENT,'THISVAL')) then thisval = 1.
```

Hope this helps,

Martin.

--
---------- -------
Dr. Martin Schultz
Department for Engineering&Applied Sciences, Harvard University
109 Pierce Hall, 29 Oxford St., Cambridge, MA-02138, USA

phone: (617)-496-8318
fax : (617)-495-4551

e-mail: mgs@io.harvard.edu

Internet-homepage: http://www-as.harvard.edu/people/staff/mgs/
 ---------- -------

```
; $Id: chkstru.pro,v 1.1 1998/10/09 19:53:32 mgs Exp $
 ;------------------------------------------------------------ --
;+
; NAME:
;       CHKSTRU  (function)
;
; PURPOSE:
;       check validity of a structure and test if necessary
;       fields are contained
;
; CATEGORY:
;       tools
;
; CALLING SEQUENCE:
;       res=CHKSTRU(STRUCTURE,FIELDS [,/VERBOSE])
;
; INPUTS:
;       STRUCTURE --> the structure to be tested. If STRUCTURE is
;           not of type structure, the function will return 0
;
;       FIELDS --> a string or string array with field names to
;           be contained in STRUCTURE. CHKSTRU returns 1 (true)
;           only if all field names are contained in STRUCTURE.
;           The entries of FIELDS may be upper or lowercase.
;
; KEYWORD PARAMETERS:
;       INDEX --> a named variable that will contain the indices of
;           the required field names in the structure. They can then
;           be assessed through structure.(index(i)) . Index will
;           contain -1 for all fields entries that are not in the
;           structure.
;
;       /VERBOSE --> set this keyword to return an error message
;           in case of an error.
;
; OUTPUTS:
;       CHKSTRU returns 1 if successful, otherwise 0.
;
; SUBROUTINES:
;
; REQUIREMENTS:
;
; NOTES:
```

```
;
; EXAMPLE:
;       test = { a:1, b:2, c:3 }
;       required = ['a','c']
;       if CHKSTRU(test,required) then print,'found a and c.'
;
; MODIFICATION HISTORY:
;       mgs, 02 Mar 1998: VERSION 1.00
;       mgs, 07 Apr 1998: - second parameter (FIELDS) now optional
;
;-
; Copyright (C) 1998, Martin Schultz, Harvard University
; This software is provided as is without any warranty
; whatsoever. It may be freely used, copied or distributed
; for non-commercial purposes. This copyright notice must be
; kept with any copy of this software. If this software shall
; be used commercially or sold as part of a larger package,
; please contact the author to arrange payment.
; Bugs and comments should be directed to mgs@io.harvard.edu
; with subject "IDL routine chkstru"
 ;-------------------------------------------------------------- --


function chkstru,structure,fields,index=index,verbose=verbose


    ; default index
    index = -1

    ; first check number of parameters (must be at least 1)
    if (n_params() lt 1) then begin
       if(keyword_set(verbose)) then $
          print,'CHKSTRU: ** invalid number of parameters ! **'
          return,0
       endif


    ; check if the user really passed a structure

    s = size(structure)
    if (s(1+s(0)) ne 8) then begin
       if(keyword_set(verbose)) then $
          print,'CHKSTRU: ** No structure passed ! **'
       return,0
    endif

    ; only one parameter: then we are finished
    if (n_params() eq 1) then return,1
```

```
    ; see if required field names are contained in the structure
    ; and return indices of these fields

    names = tag_names(structure)
    index = intarr(n_elements(fields)) - 1   ; default index to 'not found'

    for i=0,n_elements(fields)-1 do begin
       ind = where(names eq strupcase(fields(i)))
       if (ind(0) lt 0) then begin
          if(keyword_set(verbose)) then $
            print,'CHKSTRU: ** Cannot find field '+fields(i)+' ! **'
       endif else index(i) = ind(0)
    endfor


    ; check minimum value of index field: -1 indicates error
    return,(min(index) ge 0)

end
```

File Attachments

1) chkstru.pro, downloaded 96 times