Posted by Ali Gamal on Sun, 05 Oct 2014 15:18:44 GMT
View Forum Message <> Reply to Message

On Friday, October 3, 2014 10:39:02 PM UTC+2, Ali Gamal wrote:

my program as

```
;+
; NAME:
;       SYM
;
; PURPOSE:
;
;       This function provides a convenient way to utilize the
;       USERSYM procedure to create an extended choice of plotting
;       symbols, and is intended to be used directly with the PSYM
;       keyword to PLOT, OPLOT, etc.
;
; CALLING SEQUENCE:
;
;       Result=SYM(NUMBER)
;
; INPUTS:
;
;       NUMBER - symbol number
;
;             0 : dot
;             1 : filled circle
;             2 : filled upward triangle
;             3 : filled downward triangle
;             4 : filled diamond
;             5 : filled square
;             6 : open circle
;             7 : open upward triangle
;             8 : open downward triangle
;             9 : open diamond
;            10 : open square
;            11 : plus
;            12 : X
;            13 : star
;            14 : filled rightfacing triangle
;            15 : filled leftfacing triangle
;            16 : open rightfacing triangle
;            17 : open leftfacing triangle
;
; OUTPUTS:
```

```
;
;       The function returns the symbol number to be used with the
;       PSYM keyword in the PLOT, OPLOT, etc. commands
;
; SIDE EFFECTS:
;
;       The USERSYM procedure is used to create a symbol definition.
;
; EXAMPLE:
;
;       To produce a plot using open circles as plotting symbols:
;
;       PLOT,X,Y,PSYM=SYM(6)
;
; MODIFICATION HISTORY:
;
;       Martin Schultz, Harvard University, 22 Aug 1997: VERSION 1.00
;
;       (with some minor changes to the information in this header by
;       D. Windt, windt@bell-labs.com)
;
;-
; Copyright (C) 1997, 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 sym"
 ;------------------------------------------------------------ --


function sym,number

on_error,2                      ; return to caller

if(n_elements(number) eq 0) then return,1 ; default

result=8                        ; default: return psym=8, i.e.
                              ; user defined symbol

; define some help variables for
; circle :
phi=findgen(32)*(!PI*2/32.)
phi = [ phi, phi(0) ]
```

```
case number of

   0  : result = 3            ; dot

   1  : usersym, cos(phi), sin(phi), /fill
                              ; filled circle

   2  : usersym, [ -1, 0, 1, -1 ], [ -1, 1, -1, -1 ], /fill
                              ; filled upward triangle

   3  : usersym, [ -1, 0, 1, -1 ], [  1, -1, 1, 1 ], /fill
                              ; filled downward triangle

   4  : usersym, [ 0, 1, 0, -1, 0 ], [ 1, 0, -1, 0, 1 ], /fill
                              ; filled diamond

   5  : usersym, [ -1, 1, 1, -1, -1 ], [ 1, 1, -1, -1, 1 ], /fill
                              ; filled square

   6  : usersym, cos(phi), sin(phi)
                              ; open circle

   7  : usersym, [ -1, 0, 1, -1 ], [ -1, 1, -1, -1 ]
                              ; open upward triangle

   8  : usersym, [ -1, 0, 1, -1 ], [  1, -1, 1, 1 ]
                              ; open downward triangle

   9  : usersym, [ 0, 1, 0, -1, 0 ], [ 1, 0, -1, 0, 1 ]
                              ; open diamond

   10  : usersym, [ -1, 1, 1, -1, -1 ], [ 1, 1, -1, -1, 1 ]
                              ; open square

   11  : result = 1           ; plus

   12  : result = 7           ; X

   13  : result = 2           ; star

   14  : usersym, [ -1, 1, -1, -1 ], [1, 0, -1, 1 ], /fill
                              ; rightfacing triangle, filled

   15  : usersym, [ 1, -1, 1, 1 ], [1, 0, -1, 1 ], /fill
                              ; leftfacing triangle, filled

   16  : usersym, [ -1, 1, -1, -1 ], [1, 0, -1, 1 ]
                              ; rightfacing triangle, open
```

```
   17  : usersym, [ 1, -1, 1, 1 ], [1, 0, -1, 1 ]
                        ; leftfacing triangle, open

   else : begin
      message,/info,'invalid symbol number - set to 1'
      result = 1
   end
endcase
return,result
end
;.............................................................
Device, Decomposed=0
;       0  1    2   3   4   5   6   7   8   9   10  11  12  13
tvlct,[  0,255, 255,  0,  0,255,255,  0,255,125,125,  0,  0,255],$ ;red
     [  0,255,   0,255,  0,255,  0,255,125,255,  0,255,125,  0],$ ;green
     [  0,255,   0,  0,255,  0,255,255,  0,  0,255,125,255,125]   ;blue
;SET_PLOT, 'PS'
;DEVICE, FILE='24bit.ps', /COLOR, BITS=15

;!P.MULTI = [0,2,2]
;!P.MULTI = 0.
 ;.............................................................. ....
close, 1
file='synthe.dat'
openr,1,file
nn=14
falcc=fltarr(2,nn)
readf,1,falcc
wavelength=falcc(0,*)
abundance=falcc(1,*)
plot,wavelength,abundance,$
 xrange=[330.3,600.4],xstyle=1,yrange=[1.2,1.7],linestyle=0,x title='Wavelength (nm) ',ytitle='?
log(A)Nd',charsize=1.25, $
COLOR=10,background=255,PSYM=SYM(6),symsize = 1.8
xyouts,  336.3,1.47,'.............................................. .....................',color=10,charsize=4
xyouts,  336.3,1.43,'.............................................. .....................',color=10,charsize=4
xyouts,  336.3,1.39,'.............................................. .....................',color=10,charsize=4
;-----------------------------------------------------------
t=tvrd(true=1)
filename='synthe.jpg'
write_jpeg,filename,t,true=1
;-----------------------------------------------------------
end
```