


```
[[ Martin Schultz Max-Planck-Institut fuer Meteorologie  [[
[[ Bundesstr. 55, 20146 Hamburg  [[
[[ phone: +49 40 41173-308  [[
[[ fax: +49 40 441787  [[
[[ martin.schultz@dkrz.de  [[
[[[
```

```
; $Id: sym.pro,v 1.10 1999/01/22 20:12:17 mgs Stab $
```

```
;-----
```

```
;+
```

```
; NAME:
```

```
; SYM
```

```
; PURPOSE:
```

```
; define a standard sequence of plotting symbols
```

```
; CATEGORY:
```

```
; utility
```

```
; CALLING SEQUENCE:
```

```
; PLOT,X,Y,PSYM=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
```

```
; KEYWORD PARAMETERS:
```

```
; OUTPUTS:
```

```
; function returns the symbol number to be used with PSYM= in the
```

```

; PLOT command
;
;
; SUBROUTINES:
; SHOWSYM : Can be used to produce a symbol chart for reference
; (Type .r sym, then showsym, optionally with the /PS option).
; Extra keywords are passed to PLOTS, so you can e.g. choose
; a fancy color for your chart.
;
;
; REQUIREMENTS:
;
;
; NOTES:
; This function produces a side effect in that the USERSYM procedure
; is used to create a symbol definition. It's meant for usage within
; the PLOT, OPLOT, etc. command
;
;
; EXAMPLE:
; PLOT,X,Y,PSYM=SYM(0),SYMSIZE=3
; produces a plot with dots (standard symbol 3)
; FOR I=0,17 DO OPLOT,X+1,Y,PSYM=SYM(I),COLOR=I
; overplots 17 curves each with its own symbol
;
;
; MODIFICATION HISTORY:
; mgs, 22 Aug 1997: VERSION 1.00
; mgs, 10 Sep 1999: - added SHOWSYM procedure
;
;
;-
; 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"
;-----

```

```
pro showsym,ps=ps,_EXTRA=e
```

```
FORWARD_FUNCTION SYM
```

```

psflag = keyword_set(PS)
if (psflag) then begin
  olddev = !D.NAME
  set_plot,'PS'
  device,/COLOR,bits=8,xsize=8,ysize=5,yoffset=3,/INCHES, $

```

```

        filename='symbols.ps'
endif

plot,findgen(18),/NODATA,xstyle=4,YSTYLE=4
for i=0,17 do begin
    plots,1,18-i,PSYM=SYM(i),_EXTRA=e
    xyouts,0.5,18-i-0.2,strtrim(i,2),align=1.
endfor

if (psflag) then begin
    device,/close
    set_plot,olddev
    print,'Symbolist created as symbols.ps.'
endif

return
end

; ~~~~~

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
    print,'invalid symbol number - set to 1'
    result = 1
end

endcase

```

```
return,result  
end
```

File Attachments

1) [sym.pro](#), downloaded 92 times
