pst-poly

A PSTricks package for drawing polygons; v.1.63

Denis Girou
Herbert Voß

April 16, 2023

Contents

`pst-poly` allows to draw easily various kinds of regular or non regular polygons, using the unique macro `\PstPolygon`, with various customization parameters. It is also a good example of the great power and flexibility of PSTricks, as in fact it is a very short program (it body is only 100 lines long) but nevertheless really powerful. And last, it is also a good pedagogical example of how to design and program high level graphic objects above PSTricks own ones.

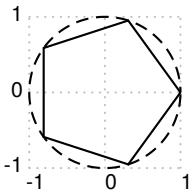
1 Introduction

pst-poly offers a unique macro (plus some aliases to define some often used polygons) with few parameters to interact on it. But we can also use all the relevant PSTricks parameters to change the size, the characteristics of lines, to add filling, etc. The polygons are always drawn counter clockwise. The syntax is simply:

```
\PstPolygon * [Options]
\PstTriangle [Options]
\PstSquare [Options]
\PstPentagon [Options]
\PstHexagon [Options]
\PstHeptagon [Options]
\PstOctogon [Options]
\PstNonagon [Options]
\PstDecagon [Options]
\PstDodecagon [Options]
\PstStarFiveLines [Options]
\PstStarFive [Options]
\pspolygonbox [Options] {Text}
```

As for PSTricks closed objects, the * version uses a solid style to fill the polygon, use the line color for fill color and set the linewidth to 0.

By default the polygons are set with a radius of 1 unit for the outer circle.



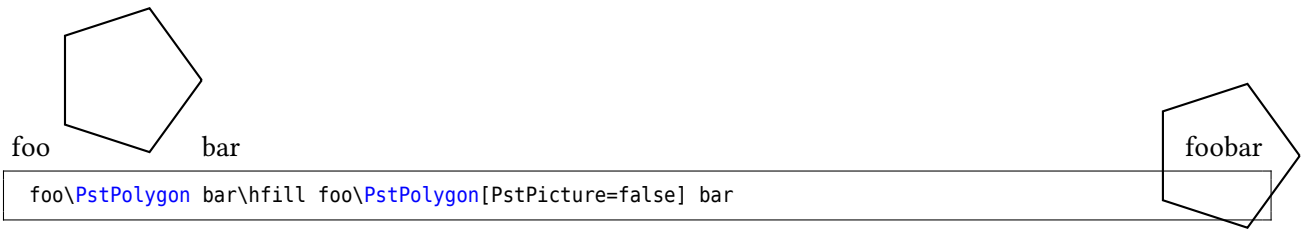

```
\begin{pspicture}[showgrid=true](-1,-1)(1,1)
  \PstPolygon[PstPicture=false]
  \pscircle[linestyle=dashed]{1}
\end{pspicture}
```

There is no special optional argument for this radius, the polygon can be scaled by using the key unit. With unit=1.5, the outer radius will be of 1.5cm when the current unit is set to 1cm.

2 Optional arguments

There are eight specific optional arguments defined to change the way the polygons are defined:

PstPicture (boolean): to define or not a pspicture environment for the polygon. We have to define this parameter to false if we want to mix the polygon with other PSTricks objects — see examples later (*Default: true* — which is not the case for basic PSTricks objects). With PstPicture=false the image doesn't reserve any space, it overwrites the text. The resulting box has a width and a height of 0pt.

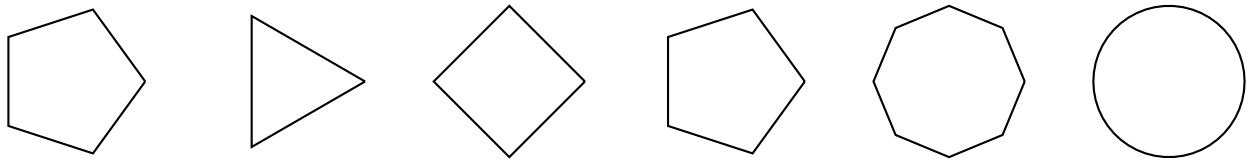



```
foo\PstPolygon bar\hfill foo\PstPolygon[PstPicture=false] bar
```

PolyRotation (real): rotation angle applied to the polygon (*Default: 0* — no rotation).

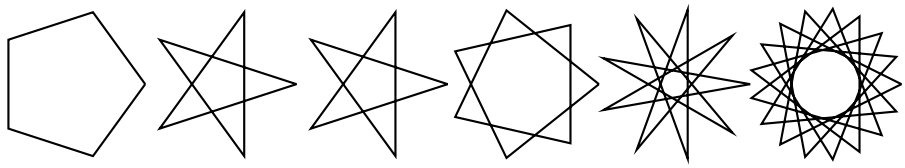
```
\PstPolygon\hfill
\PstPolygon[PolyRotation=18]\hfill
\PstPolygon[PolyRotation=36]\hfill
\PstPolygon[PolyRotation=45]
```

PolyNbSides (integer): number of sides of the polygon (*Default: 5*).



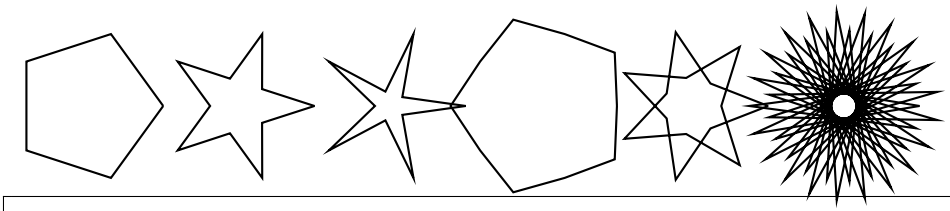
```
\PstPolygon\hfill
\PstPolygon[PolyNbSides=3]\hfill
\PstPolygon[PolyNbSides=4]\hfill
\PstPolygon[PolyNbSides=5]\hfill
\PstPolygon[PolyNbSides=8]\hfill
\PstPolygon[PolyNbSides=50]
```

PolyOffset (integer): number of nodes to bypass to obtain each time the next one (*Default: 1* — no node bypassed).



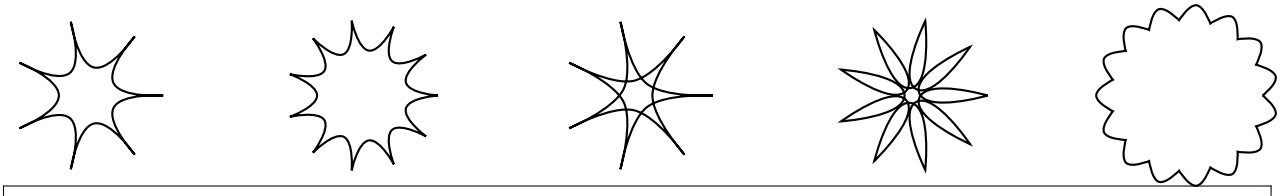
```
\PstPolygon
\PstPolygon[PolyOffset=2]
\PstPolygon[PolyOffset=3]
\PstPolygon[PolyNbSides=7,PolyOffset=2]
\PstPolygon[PolyNbSides=9,PolyOffset=4]
\PstPolygon[PolyNbSides=17,PolyOffset=6]
```

PolyIntermediatePoint (real): position of the intermediate point used to join each time the next node (*Default: empty* — not used).



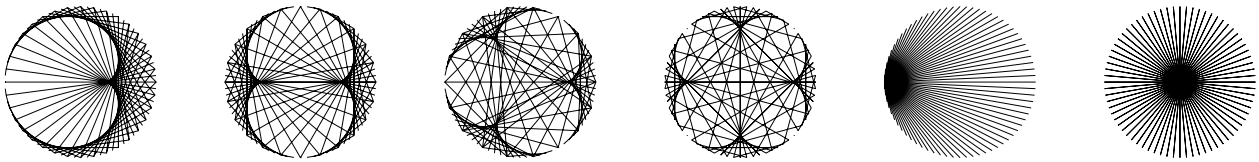
```
\PstPolygon
\PstPolygon[PolyIntermediatePoint=0.38]
\PstPolygon[PolyIntermediatePoint=0.2]
\PstPolygon[PolyIntermediatePoint=1.2]
\PstPolygon[PolyNbSides=7,PolyOffset=2,
        PolyIntermediatePoint=0.38]
\PstPolygon[PolyNbSides=21,PolyOffset=2,
        PolyIntermediatePoint=-1.25]
```

PolyCurves (boolean): boolean value to choose between straight line and curve to join each time the next node (*Default:* `false` — straight lines).



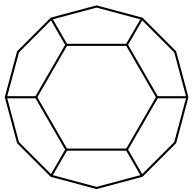
```
\psset{PolyCurves=true}
\PstPolygon[PolyNbSides=7,PolyIntermediatePoint=0.38]\hfill
\PstPolygon[PolyNbSides=11,PolyIntermediatePoint=0.6]\hfill
\PstPolygon[PolyNbSides=7,PolyIntermediatePoint=0.2,
        PolyOffset=2]\hfill
\PstPolygon[PolyNbSides=9,PolyIntermediatePoint=0.1]\hfill
\PstPolygon[PolyNbSides=15,PolyIntermediatePoint=1.2]
```

PolyEpicycloid (boolean): boolean value to choose between polygon and epicycloid (*Default:* `false` — polygon).



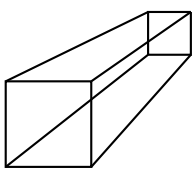
```
\psset{linewidth=0.001,PolyNbSides=72,PolyEpicycloid=true}
% Epicycloid of factor 1 is cardioid and of factor 2 nephroid
\multido{\i=2+1}{4}{\PstPolygon[PolyOffset=\i]\hfill}
\PstPolygon[PolyOffset=72]\hfill% Epicycloid of factor 71
\PstPolygon[PolyOffset=73] % Epicycloid of factor 72
```

PolyName (string): name of the polygon, useful to have different names for the nodes of different polygons (*Default: empty* — no name).
The center of the polygon has name `PolyName0` and the nodes (vertices) have names `PolyName1` to `PolyNameN`. With this parameter, we can connect as we want nodes of different polygons:



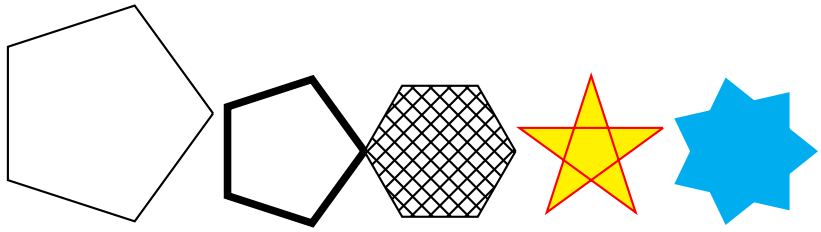
```
\psset{PstPicture=false}
\begin{pspicture}(-1,-1)(1,1)
  \PstPolygon[unit=0.8,PolyName=A,PolyNbSides=6]
  \PstPolygon[unit=1.2,PolyName=B,PolyNbSides=12]
\end{pspicture}
\multido{\i=1+2}{6}{%
  \ncline{A\the\multidocount}{B\i}}
```

It is also a way (limited in fact...) to define three dimensional objects in perspective:

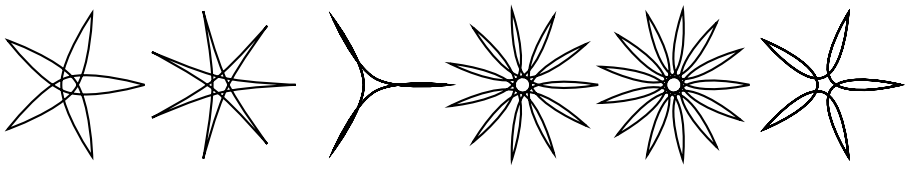


```
\psset{unit=0.8}
\begin{pspicture}(3,2.5)
% \PstSquare is described later
  \rput[lb](0,0){\PstSquare[PolyName=A]}
    \rput[lb](2.5,2){\PstSquare[unit=0.5,PolyName=B]}
    \multido{\i=1+1}{4}{\ncline{A\i}{B\i}}
  \end{pspicture}
```

Of course, we can mix specific parameters of `pst-poly` with relevant PSTricks ones and combine it with other generic macros (for repetitions, projection in the 3d space, etc.)



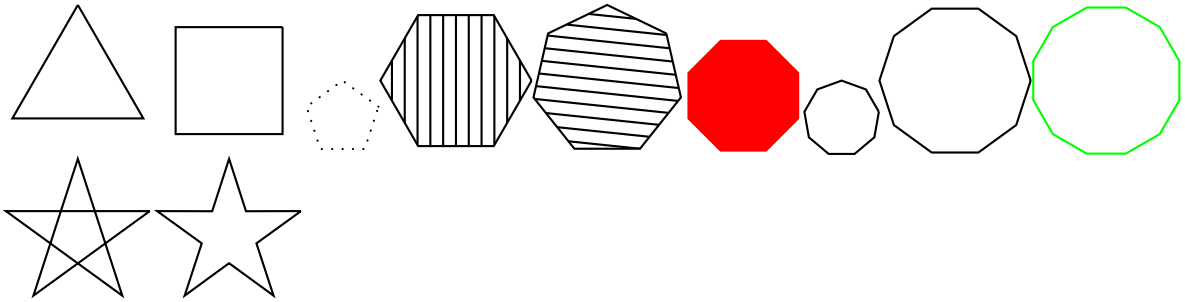
```
\PstPolygon[unit=1.5]
\PstPolygon[linewidth=1mm]
\PstPolygon[fillstyle=crosshatch,PolyNbSides=6]
\PstPolygon[linecolor=red,fillstyle=solid,fillcolor=yellow,
        PolyRotation=18,PolyOffset=2]
\PstPolygon*[linecolor=cyan,PolyNbSides=7,PolyOffset=2]
```



```
\multido{\i=5+2}{6}{%
  \PstPolygon[PolyNbSides=\i,PolyCurves=true,
          PolyIntermediatePoint=0.1,PolyOffset=3]}
```

3 Pre-defined polygons

Some often used polygons and other related geometric objects are pre-defined, for immediate usage :



```
\PstTriangle
\PstSquare
\PstPentagon[unit=0.5,linestyle=dotted]
\PstHexagon[fillstyle=hlines,hatchangle=90]
\PstHeptagon[fillstyle=vlines]
\PstOctogon*[unit=0.8,linecolor=red]
\PstNonagon[unit=0.5]
\PstDecagon
\PstDodecagon[linecolor=green] \par
\PstStarFiveLines
\PstStarFive
```

4 Non regular polygons

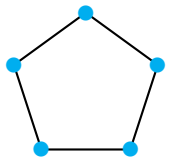
Until now, we have described only the so-called *regular* polygons, which are from far the most useful ones (all of them have equal edges and angles). Nevertheless, it is not so difficult to extend these polygon to *non regular*

ones, using a different value for horizontal and vertical units (nevertheless, the code is more tricky, as we must do all the trigonometry explicitely...)

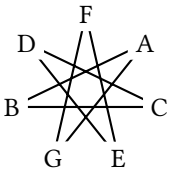
```
\PstPentagon[xunit=0.5]\hfill
\PstHexagon[yunit=0.5]\hfill
\PstStarFive[xunit=0.5,yunit=1.5]\hfill
\PstPolygon[xunit=0.8,yunit=1.5,PolyNbSides=9,PolyOffset=2,
         PolyIntermediatePoint=0.1,PolyCurves=true]
```

5 Nodes (vertices)

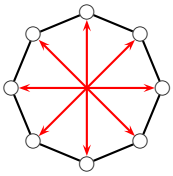
And another powerful possibility is to define a command `\PstPolygonNode` which will be excuted at each node (*Default: empty* — nothing executed). The counter name for nodes is `INode`, starting from 0. The `\multidocount` counter, from the `\multido` command, start itself from 1.



```
\providecommand{\PstPolygonNode}{%
  \psdots[dotsize=0.2,linecolor=cyan](1;\INode)}
\PstPentagon
```

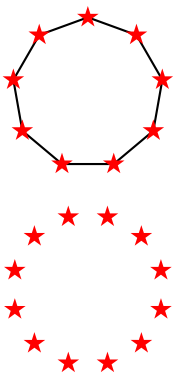



```
\newcounter{Letter}
\providecommand{\PstPolygonNode}{%
  \setcounter{Letter}{\the\multidocount}%
  \rput*{*0}(1;\INode){\small\Alph{Letter}}}
\PstHeptagon[PolyOffset=3]
```

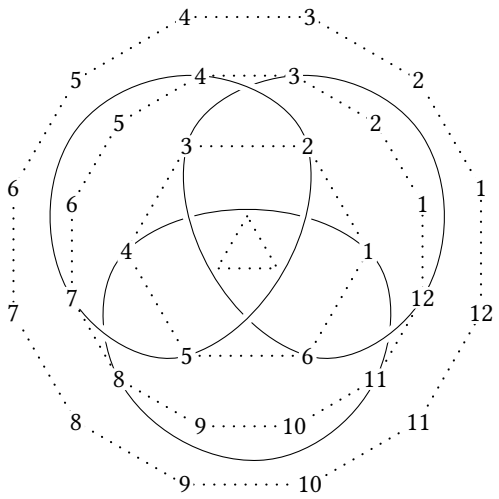


```
\providecommand{\PstPolygonNode}{%
  \psdots[dotstyle=o,dotsize=0.2](1;\INode)
  \psline[linecolor=red]{->}(0.9;\INode)}
\PstPolygon[PolyNbSides=8]
```

It is also a way to nest polygons:



```
\newbox{\Star}
\savebox{\Star}{%
  \PstStarFive*[unit=0.15,linecolor=red]}
\providecommand{\PstPolygonNode}{%
  \rput{*0}(1;\INode){\usebox{\Star}}}
\shortstack{\PstNonagon\\[5mm]
          \PstDodecagon[linestyle=none]}
```

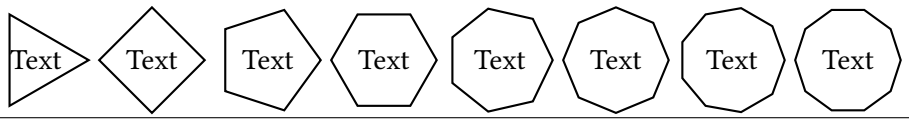


```
\psset{unit=2,PstPicture=false}
\begin{pspicture}(-1.6,-1.6)(1.6,1.6)
%   Just to name the nodes, to be able to join some of them.
%   by solid curves. We can't draw the polygons now,
%   as the node numbers must erase the solid curves...
  \PstHexagon[unit=0.8,linestyle=none,PolyName=H]\PstDodecagon[unit=1.2,linestyle=none,PolyName=D]
  {\SpecialCoor
   \psset{linewidth=0.4pt,border=2pt,nodesep=0.45}
   \psccurve(H1)(H4)(D8)([angle=-30]D9)(D11)\psccurve(D3)(H3)(H6)(D12)([angle=90]D1)
   \psccurve(D4)(H2)(H5)(D7)([angle=90]D6)}
  \psset{linestyle=dotted,framesep=1pt}
  \PstTriangle[unit=0.23]
  \providecommand{\PstPolygonNode}{\rput*{*0}(1;\INode){\small\the\multidocount}}
  \PstHexagon[unit=0.8] \PstDodecagon[unit=1.2] \PstDodecagon[unit=1.6]
\end{pspicture}
```

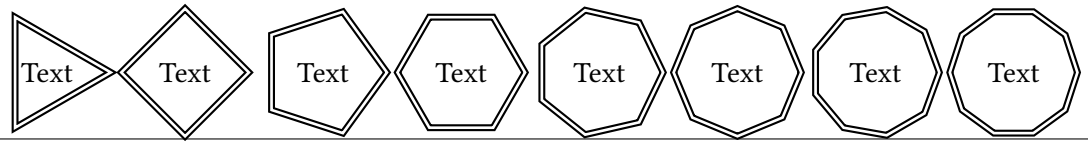
6 Polygonbox

The valid options with the predefined values are `PolyNbSides=3` and `PolyRotation=0`

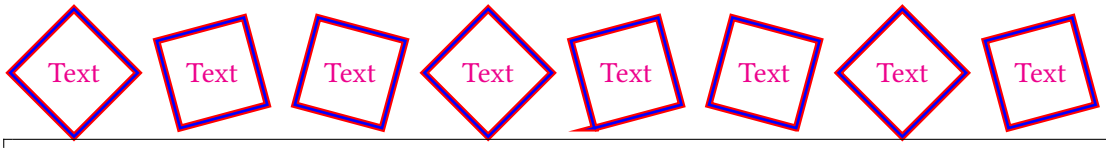
- There maybe some problems with linearcs and rounding errors.
- To rotate the text inside the box, one can use the `\rotatebox` macro from the `rotating` package (see examples).



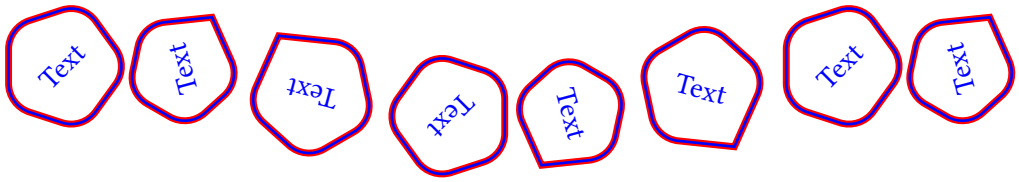

```
\multido{\nA=3+1}{8}{%
  \pspolygonbox[PolyNbSides=\nA,framesep=3mm]{Text}~}
```



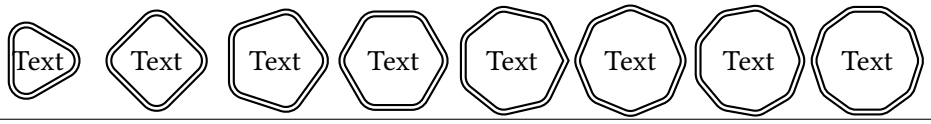

```
\multido{\nA=3+1}{8}{\pspolygonbox[PolyNbSides=\nA,%
    framesep=2mm,doubleline=true]{Text}~}
```



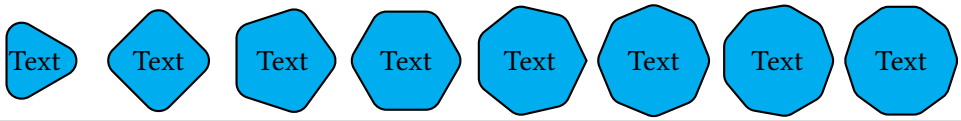

```
\psset{PolyNbSides=4}
\multido{\nA=0+60}{8}{\pspolygonbox[PolyRotation=\nA,%
   framesep=2mm,doubleline=true,%
   linecolor=red,doublecolor=blue]{\magenta Text}~}
```



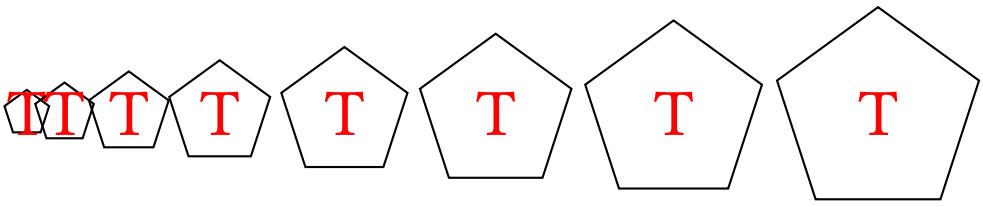

```
\multido{\nA=0+60,\nB=45+60}{8}{\pspolygonbox%
  [PolyRotation=\nA,framesep=2mm,doubleline=true,%
   linecolor=red,doublecolor=blue,linearc=0.4]%
   {\rotatebox{\nB}{\blue Text}}~}\\[1cm]
```



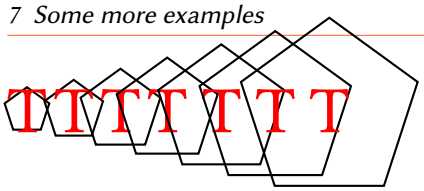

```
\multido{\nA=3+1}{8}{%
 \pspolygonbox[PolyNbSides=\nA,rot=60,framesep=2pt,%
  doubleline=true,linearc=0.2]{Text}~}
```




```
\multido{\nA=3+1}{8}{%
  \pspolygonbox[PolyNbSides=\nA,framesep=10pt,%
  fillstyle=solid,fillcolor=cyan,linearc=0.2]{Text}~}
```

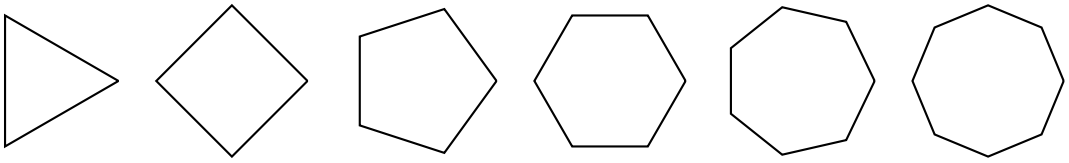



```
\multido{\nA=-5+5}{8}{\pspolygonbox[framesep=\nA pt,%
  PolyRotation=90]{\Huge\red T}~}
```

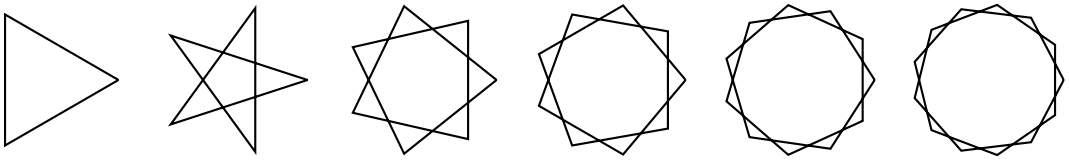


```
\multido{\nA=-5+5}{7}{\pspolygonbox[framesep=\nA pt,%
  PolyRotation=90,boxsep=false]{\Huge\red T}~}\par
\rule{0pt}{1cm}
```

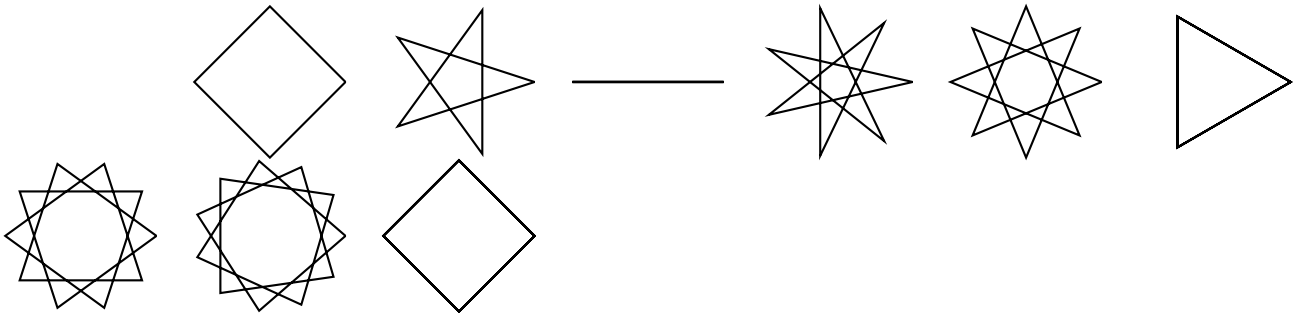
7 Some more examples



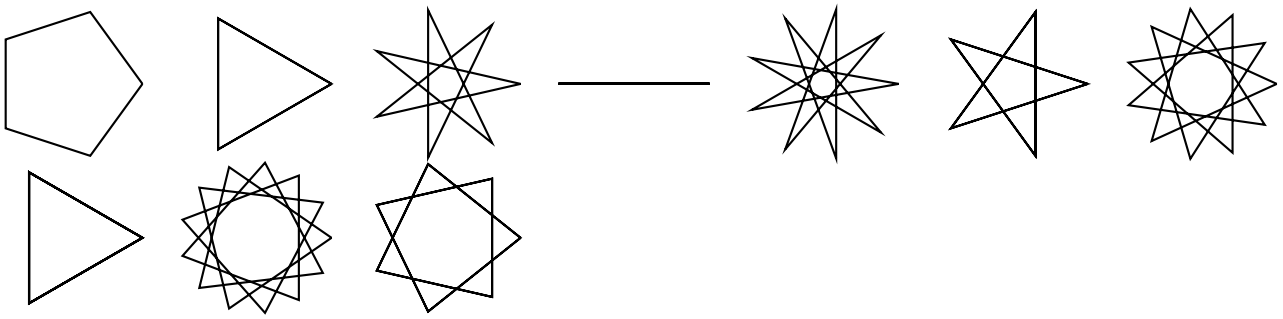
```
\multido{\i=3+1}{6}{%
  \PstPolygon[PolyNbSides=\i]\hspace{5mm}}
```



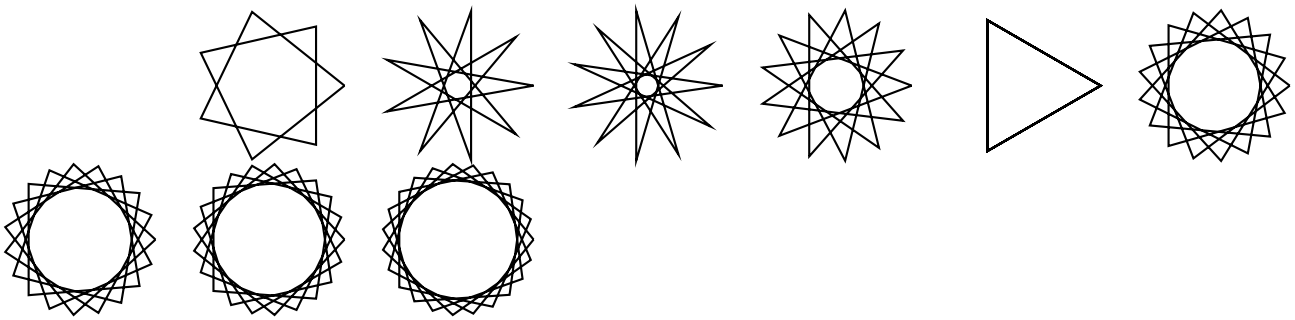
```
\multido{\i=3+2}{6}{%
  \PstPolygon[PolyOffset=2,PolyNbSides=\i]\hspace{5mm}}
```



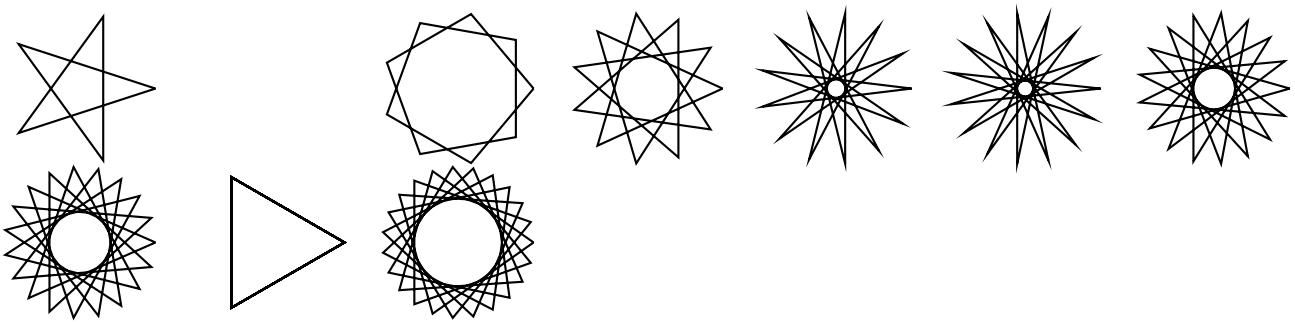
```
\multido{\i=3+1}{10}{%
  \PstPolygon[PolyOffset=3,PolyNbSides=\i]\hspace{5mm}}
```



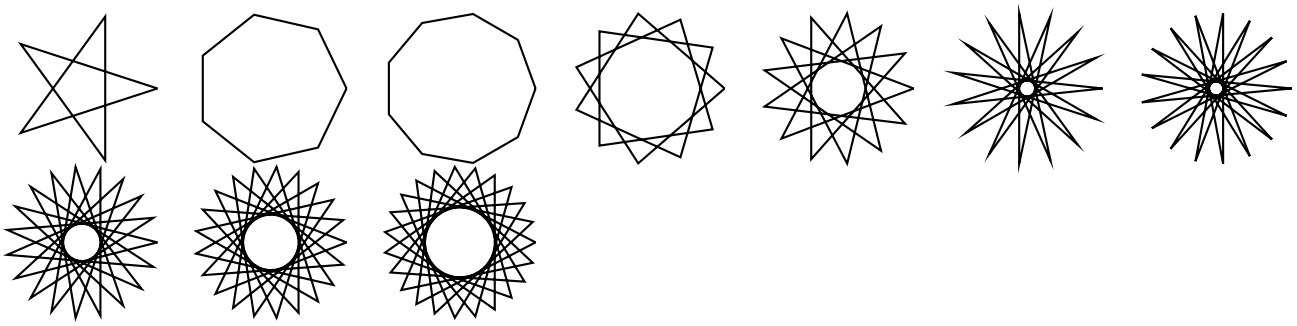
```
\multido{\i=5+1}{10}{%
  \PstPolygon[PolyOffset=4,PolyNbSides=\i]\hspace{5mm}}
```



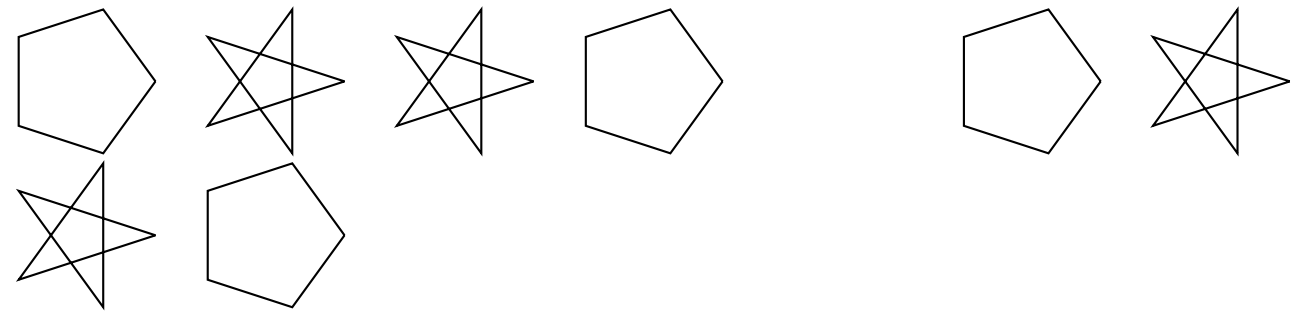
```
\multido{\i=5+2}{10}{%
  \PstPolygon[PolyOffset=5,PolyNbSides=\i]\hspace{5mm}}
```



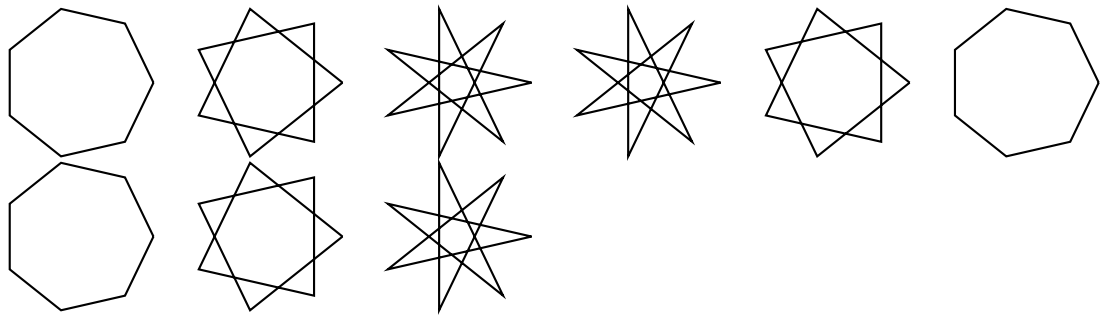
```
\multido{\i=5+2}{10}{%
  \PstPolygon[PolyOffset=7,PolyNbSides=\i]\hspace{5mm}}
```



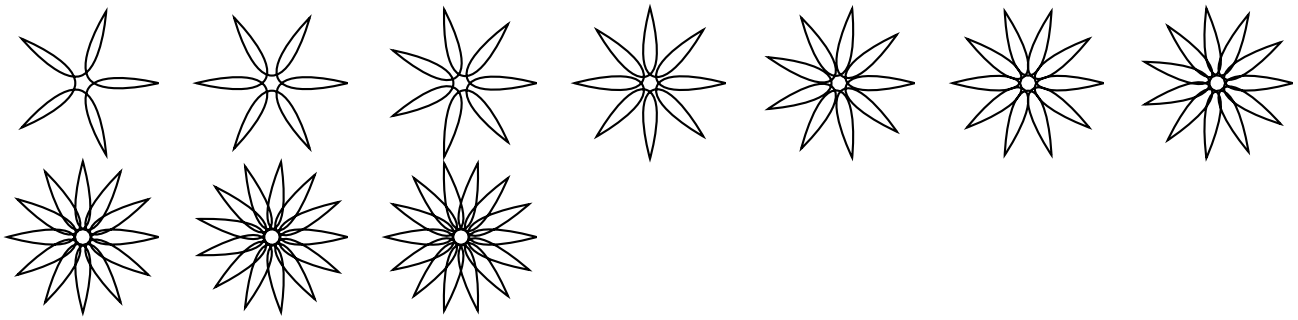
```
\multido{\i=5+2}{10}{%
  \PstPolygon[PolyOffset=8,PolyNbSides=\i]\hspace{5mm}}
```



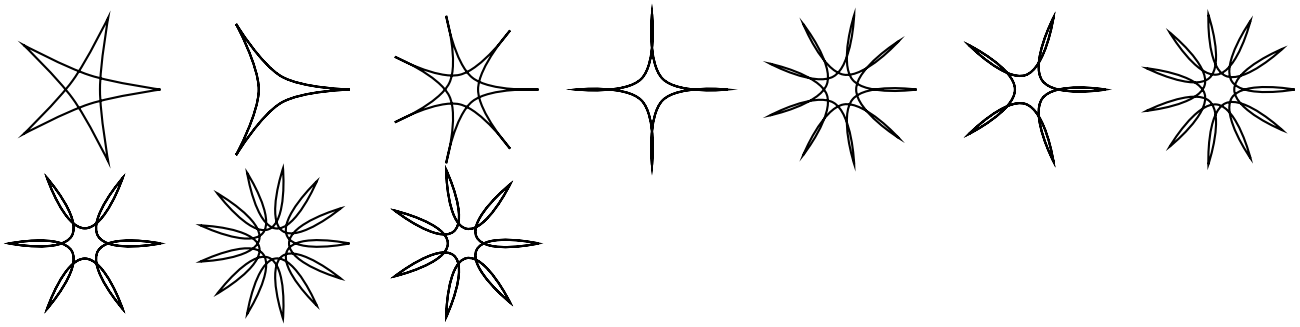
```
\multido{\i=1+1}{10}{%
  \PstPolygon[PolyOffset=\i,PolyNbSides=5]\hspace{5mm}}
```



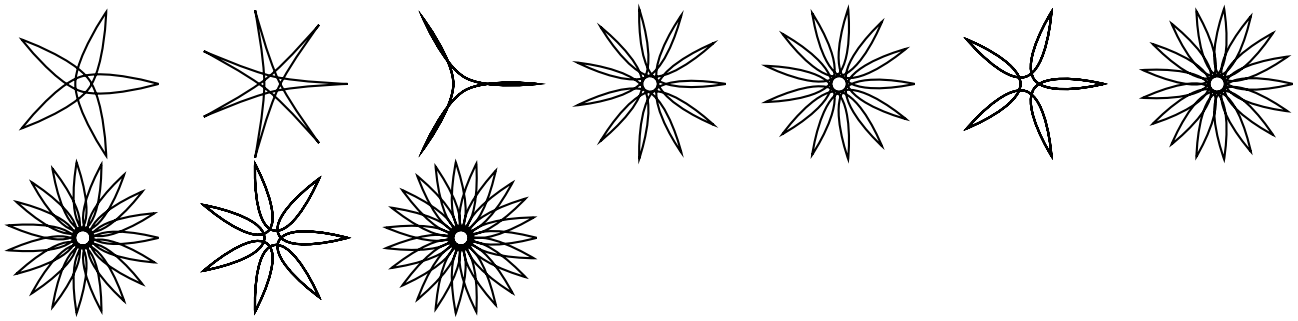
```
\multido{\i=1+1}{10}{%
  \PstPolygon[PolyOffset=\i,PolyNbSides=7]\hspace{5mm}}
```



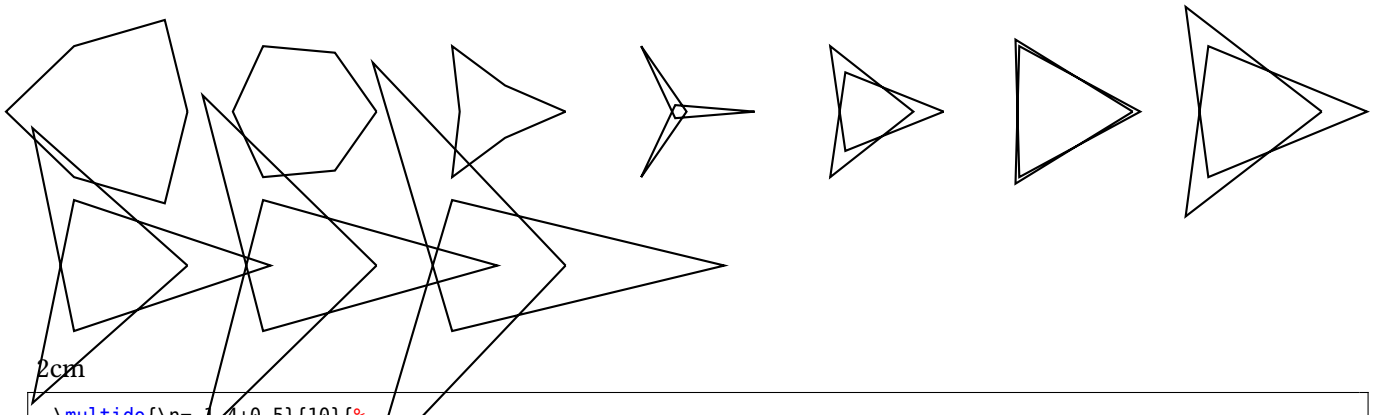
```
\multido{\i=5+1}{10}{%
  \PstPolygon[PolyCurves,PolyIntermediatePoint=0.1,PolyNbSides=\i]
  \hspace{5mm}}
```



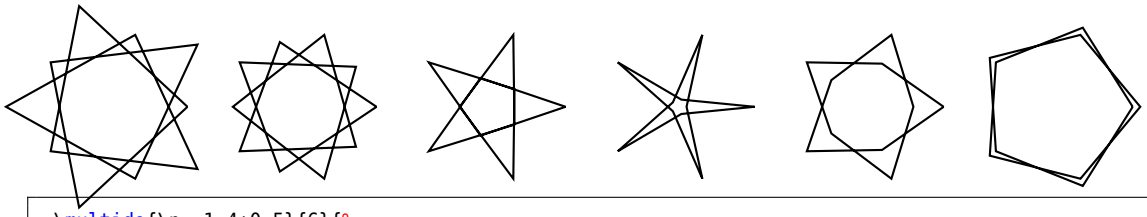
```
\multido{\i=5+1}{10}{%
  \PstPolygon[PolyCurves,PolyIntermediatePoint=0.2,
          PolyOffset=2,PolyNbSides=\i]\hspace{5mm}}
```



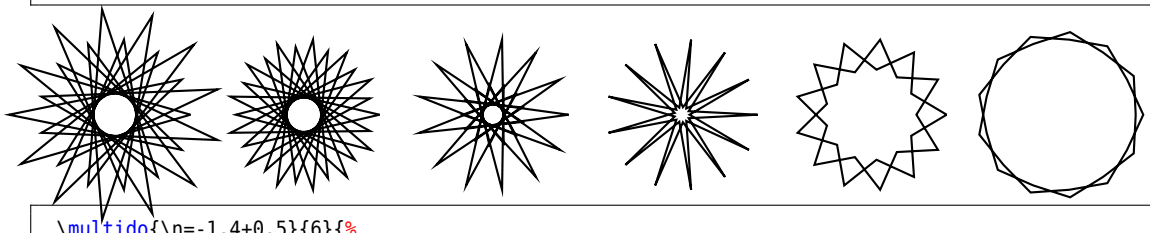
```
\multido{\i=5+2}{10}{%
  \PstPolygon[PolyCurves,PolyIntermediatePoint=0.1,
          PolyOffset=3,PolyNbSides=\i]\hspace{5mm}}
```



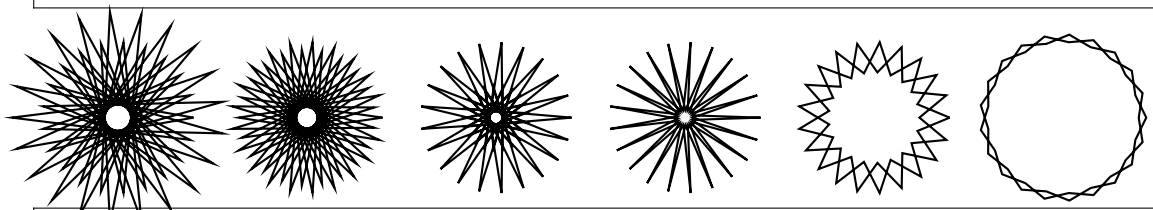

```
\multido{\n=-1.4+0.5}{10}{%
  \PstPolygon[PolyNbSides=3,PolyOffset=2,PolyIntermediatePoint=\n]
  \hspace{5mm}}\par
  \vspace{0pt}{2cm}
```



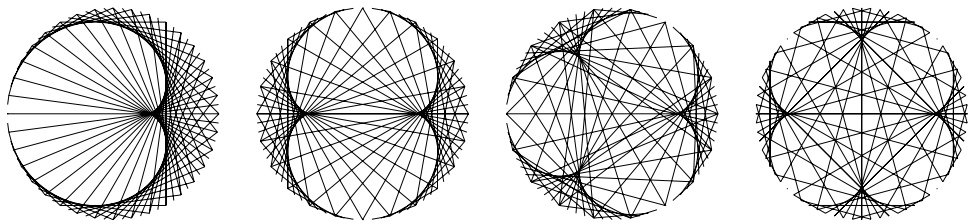
```
\multido{\n=-1.4+0.5}{6}{%
  \PstPolygon[PolyNbSides=5,PolyOffset=2,PolyIntermediatePoint=\n]
  \hspace{5mm}}
```



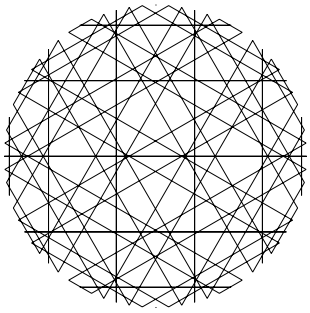
```
\multido{\n=-1.4+0.5}{6}{%
  \PstPolygon[PolyNbSides=13,PolyOffset=2,PolyIntermediatePoint=\n]
  \hspace{5mm}}
```



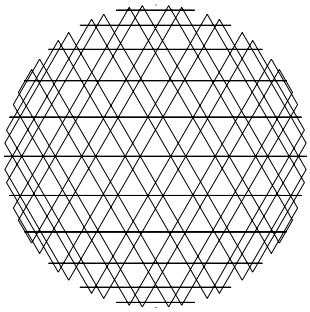
```
\multido{\n=-1.4+0.5}{6}{%
  \PstPolygon[PolyNbSides=21,PolyOffset=2,PolyIntermediatePoint=\n]
  \hspace{5mm}}
```



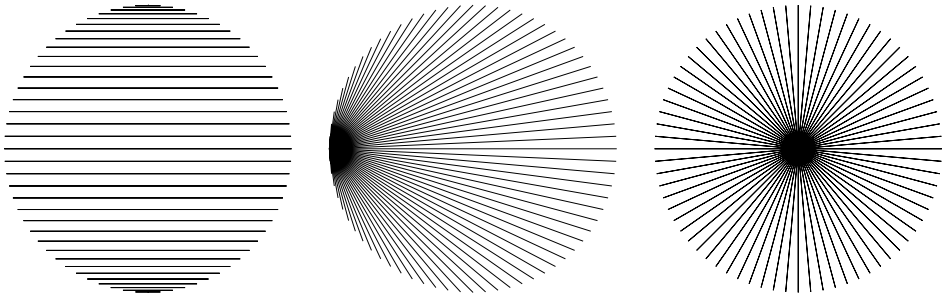
```
\psset{unit=1.4,linewidth=0.001,PolyNbSides=72,PolyEpicycloid}
\multido{\i=2+1}{4}{%
 % Epicycloid of factor 1 is cardioid and of factor 2 nephroid
  \PstPolygon[PolyOffset=\i]\hspace{5mm}}
```



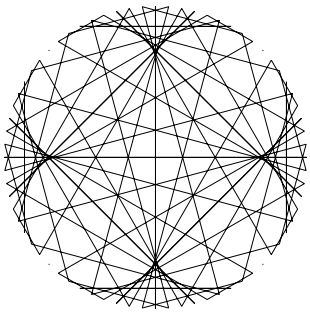
```
% Epicycloid of factor 10
\PstPolygon[unit=2,linewidth=0.003,
        PolyEpicycloid,PolyNbSides=72,PolyOffset=11]
```



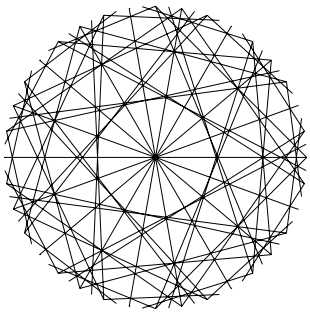
```
% Epicycloid of factor 22
\PstPolygon[unit=2,linewidth=0.003,
        PolyEpicycloid,PolyNbSides=72,PolyOffset=23]
```



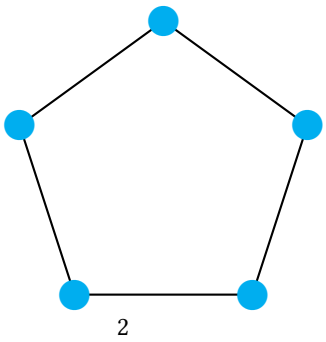
```
\psset{unit=1.9,linewidth=0.001,PolyNbSides=72,PolyEpicycloid}
\multido{\i=71+1}{3}{%
  \PstPolygon[PolyOffset=\i]\hspace{5mm}}
```



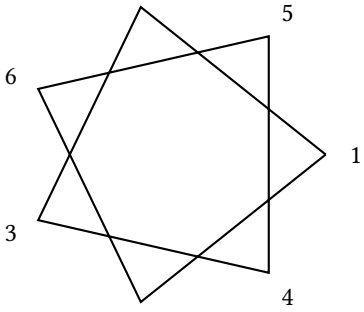
```
% Epicycloid of factor 100
\PstPolygon[unit=2,linewidth=0.003,
        PolyEpicycloid,PolyNbSides=72,PolyOffset=101]
```



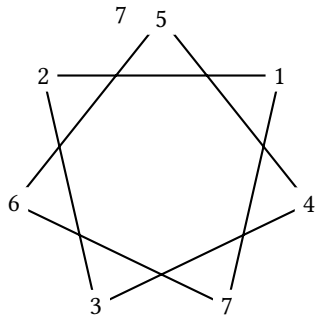
```
% Epicycloid of factor 153
\PstPolygon[unit=2,linewidth=0.003,
        PolyEpicycloid,PolyNbSides=72,PolyOffset=154]
```



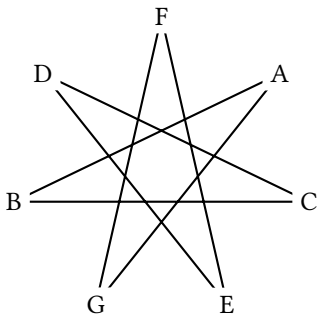

```
\providecommand{\PstPolygonNode}{%
  \psdots[dotsize=0.2,linecolor=cyan](1;\INode)}
\PstPentagon[unit=2]
```




```
\providecommand{\PstPolygonNode}{%
  \rput{*0}(1.2;\INode){\small\the\multidocount}}
\PstPolygon[unit=2,PolyNbSides=7,PolyOffset=2]
```

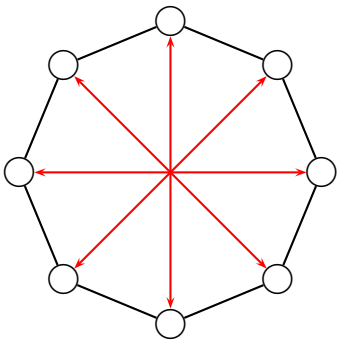



```
\providecommand{\PstPolygonNode}{%
  \rput*{*0}(1;\INode){\small\the\multidocount}}
\PstHeptagon[unit=2,PolyOffset=2]
```

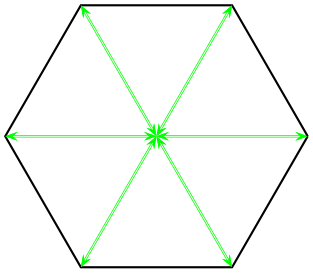



```
% \newcounter{Letter}
\providecommand{\PstPolygonNode}{%
  \setcounter{Letter}{\the\multidocount}%
  \rput*{*0}(1;\INode){\small\Alph{Letter}}}
\PstHeptagon[unit=2,PolyOffset=3]
```

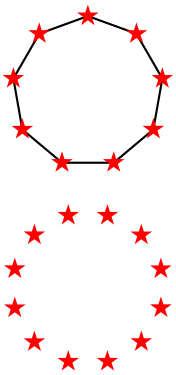
```
\providecommand{\PstPolygonNode}{%
  \SpecialCoor
  \degrees[3]
  \rput{0.5}(0.5;\INode){%
    \pspolygon*(0.5;0.5)(0.5;1.5)(0.5;2.5)}}
\PstTriangle
```



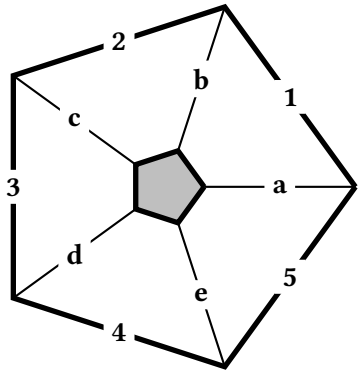
```
\providecommand{\PstPolygonNode}{%
  \psdots[dotstyle=o,dotsize=0.2](1;\INode)
  \psline[linecolor=red]{->}(0.9;\INode)}
\PstPolygon[unit=2,PolyNbSides=8]
```



```
\providecommand{\PstPolygonNode}{%
  \psline[linewidth=0.1mm,doubleline=true,
        linecolor=green]{<->}(0;0)(1;\INode)}
\PstHexagon[unit=2]
```



```
\newbox{\Star}
\savebox{\Star}{%
  \PstStarFive*[unit=0.15,linecolor=red]}
\providecommand{\PstPolygonNode}{%
  \rput{*0}(1;\INode){\usebox{\Star}}}
\shortstack{%
  \PstNonagon\\[5mm]
  \PstDodecagon[linestyle=none]}
```




```
%\newcounter{Letter}% if not defined
\setcounter{Letter}{\the\multidocount}%
\begin{pspicture}(-2.5,-2.5)(2.5,2.5)
\PstPolygon[PolyName=A,fillstyle=solid,fillcolor=lightgray,linewidth=2pt,
    PolyNbSides=5,unit=0.5,PstPicture=false]
\PstPolygon[PolyName=B,PolyNbSides=5,linewidth=2pt,PstPicture=false,unit=2.5]
\multido{\iA=1+1,\iB=2+1}{5}{%
  \setcounter{Letter}{\the\multidocount}%
  \ncline{A\iA}{B\iA}\ncput*{\bfseries\alph{Letter}}%
  \ifnum\iB>5 \def\iB{1}\fi%
  \pcline[linestyle=none](B\iA)(B\iB)\ncput*{\bfseries\iA}}
\end{pspicture}
```

8 List of all optional arguments for pst-poly

Key	Type	Default
PstPicture	boolean	true
PolyRotation	ordinary	0
PolyNbSides	ordinary	5
PolyOffset	ordinary	1
PolyIntermediatePoint	ordinary	
PolyName	ordinary	
PolyCurves	boolean	true
PolyEpicycloid	boolean	true

References

[1] Victor Eijkhout. *TEX by Topic – A TEXnician Reference*. 1 **edition**. Heidelberg/Berlin: DANTE – lehmanns media, 2014.

[2] Denis Girou. “Présentation de PSTricks”. **in***Cahier GUTenberg*: 16 (**april** 1994), **pages** 21–70.

[3] Michel Goosens **andothers**. *The LATEX Graphics Companion*. second. Boston, Mass.: Addison-Wesley Publishing Company, 2007.

[4] Nikolai G. Kollock. *PostScript richtig eingesetzt: vom Konzept zum praktischen Einsatz*. Vaterstetten: IWT, 1989.

[5] Herbert Voß. *Presentations with LATEX*. 1. Heidelberg/Berlin: DANTE – Lehmanns Media, 2012.

[6] Herbert Voß. *PSTricks – Grafik für TEX und LATEX*. 6. Heidelberg/Hamburg: DANTE – Lehmanns, 2010.

[7] Herbert Voß. *PSTricks – Graphics and PostScript for LATEX*. 1. Cambridge – UK: UIT, 2011.

[8] Herbert Voß. *LATEX quick reference*. 1. Cambridge – UK: UIT, 2012.

[9] Timothy Van Zandt. *multido.tex - a loop macro, that supports fixed-point addition*. CTAN:/macros/generic/multido.tex, 1997.

[10] Timothy Van Zandt **and** Denis Girou. “Inside PSTricks”. **in***TUGboat*: 15 (**september** 1994), **pages** 239–246.

Index