

Chronique 13

Polygones et autres dessins

Dans cette chronique, on va dessiner des polygones réguliers puis des frises circulaires, et on reviendra un peu sur les rosaces.

On va commencer par détailler la construction d'un polygone régulier dans un cas plus intéressant que le triangle ou le carré.

13.1 Pentagone

Si on veut dessiner un pentagone régulier inscrit dans un cercle par exemple de rayon 2, ce n'est pas très facile d'utiliser les coordonnées cartésiennes pour déterminer les sommets du pentagone ! On va donc utiliser un repérage polaire et prendre pour sommets les points repérés par $(2;0)$, $(2;72)$, $\dots(2;288)$. Je rappelle que les angles sont exprimés en degrés (voir page 55).

Que faut-il faire pour tracer un pentagone régulier ?

Se placer au premier sommet avec l'instruction `\moveto`, puis tracer un segment de ce premier sommet au deuxième avec l'instruction `\lineto` et ainsi de suite jusqu'au dernier sommet.

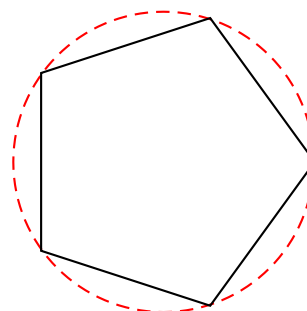
Enfin, il faut refermer la figure et revenir au premier sommet.

Naturellement on a besoin d'une boucle définie par `\multido` qui prendra autant de valeurs que le nombre de côtés du polygone, c'est-à-dire 5. La variable servant de compteur peut être de type `integer`.

Enfin, on a déjà vu qu'utiliser l'instruction `\moveto` nécessite d'être dans un environnement `\pscustom`. Et dans cet environnement, on revient au point de départ en fermant le chemin par l'instruction `\closepath`.

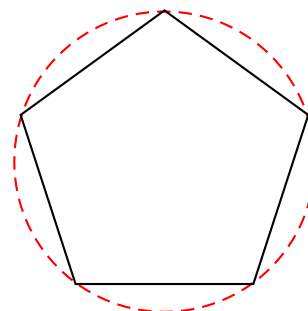
Voici le code du polygone et le résultat obtenu :

```
\psset{unit=1cm}
\def\xmin{-2} \def\xmax{2} \def\ymin{-2} \def\ymax{2}
\begin{pspicture}(\xmin,\ymin)(\xmax,\ymax)
\pscircle[linecolor=red,linestyle=dashed](0,0){2}
\pscustom{
  \moveto(2;0)%      premier sommet
  \multido{\i=0+72}% variable
    {5}%             nombre de côtés
    {\lineto(2;\i)}% tracé du côté
  \closepath%        retour au départ
}%                   fin du \pscustom
\end{pspicture}
```



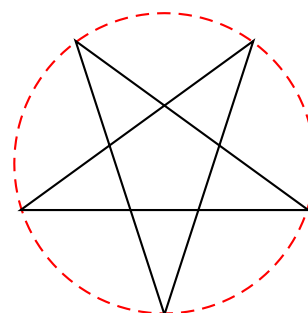
Le sommet de départ est en $(2;0)$; on pourrait le mettre à n'importe quel endroit sur le cercle. On va pour cela créer une variable `\deb` qui va désigner l'angle repérant le sommet initial :

```
\psset{unit=1cm}
\def\xmin{-2} \def\xmax{2} \def\ymin{-2} \def\ymax{2}
\begin{pspicture}(\xmin,\ymin)(\xmax,\ymax)
\pscircle[linecolor=red,linestyle=dashed](0,0){2}
\def\deb{18}
\pscustom{\moveto(2;\deb)
\multido{\i=\deb+72}{5}
{\lineto(2;\i)}}
\closepath}
\end{pspicture}
```



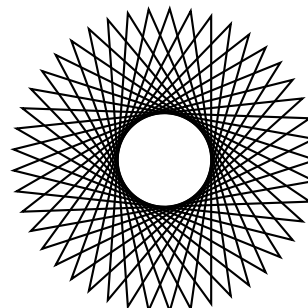
En donnant à `\deb` la valeur -18 on aura un pentagone avec la pointe en bas. Un pentagone régulier peut aussi être croisé ; il suffit de relier les sommets de 2 en 2 (puisque 5 et 2 sont premiers entre eux) :

```
\psset{unit=1cm}
\def\xmin{-2} \def\xmax{2} \def\ymin{-2} \def\ymax{2}
\begin{pspicture}(\xmin,\ymin)(\xmax,\ymax)
\pscircle[linecolor=red,linestyle=dashed](0,0){2}
\def\deb{-18}
\pscustom{\moveto(2;\deb)
\multido{\i=\deb+144}{5}
{\lineto(2;\i)}}
\closepath}
\end{pspicture}
```



Enfin on va faire tourner cette étoile pour créer un joli dessin : on va donc insérer le code du tracé du pentagone croisé dans une boucle `\multido` qui prendra 9 valeurs par pas de 8. On se passera du cercle et la variable `\deb` sera remplacée par le compteur `\I` de cette nouvelle boucle :

```
\psset{unit=1cm}
\def\xmin{-2} \def\xmax{2} \def\ymin{-2} \def\ymax{2}
\begin{pspicture}(\xmin,\ymin)(\xmax,\ymax)
\multido{\I=0+8}{9}
{
\pscustom{\moveto(2;\I)
\multido{\i=\I+144}{5}
{\lineto(2;\i)}}
\closepath}
}
\end{pspicture}
```

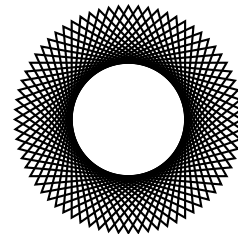
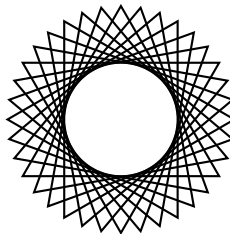
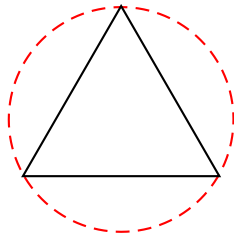
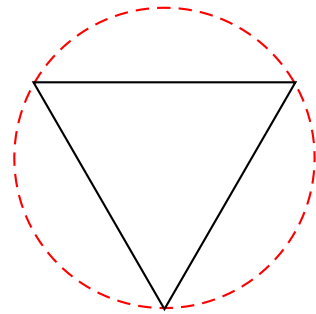


Joli, non ?

13.2 Triangle

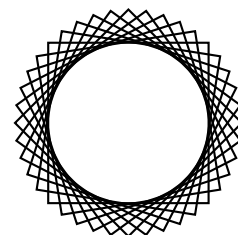
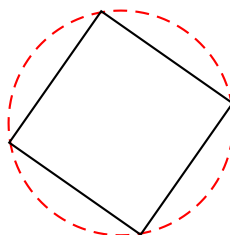
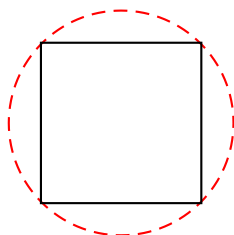
Ce que l'on a fait pour le pentagone régulier, on peut le faire pour le triangle équilatéral :

```
\psset{unit=1cm}
\def\xmin{-2} \def\xmax{2} \def\ymin{-2} \def\ymax{2}
\begin{pspicture}(\xmin,\ymin)(\xmax,\ymax)
\pscircle[linecolor=red,linestyle=dashed](0,0){2}
\def\deb{30}
\pscustom{
\moveto(2;\deb)
\multido{\i=\deb+120}{3}
{\lineto(2;\i)}
\closepath
}
\end{pspicture}
```



13.3 Carré

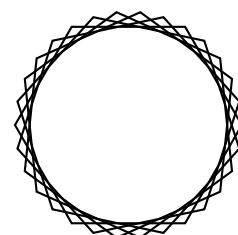
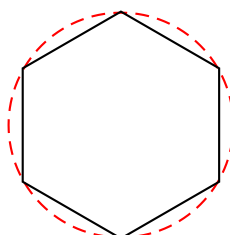
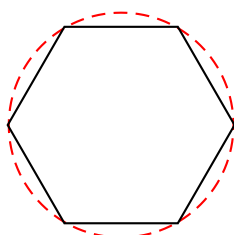
Pour le carré, on modifie le nombre de côtés et l'angle de décalage :



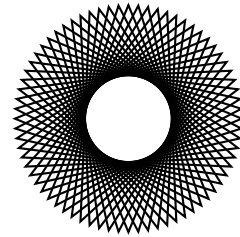
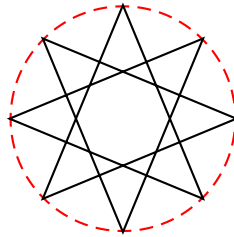
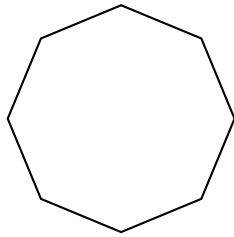
13.4 Autres polygones réguliers

On peut décliner le procédé en changeant le nombre de côtés.

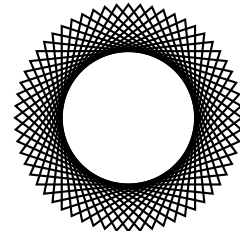
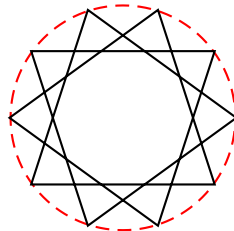
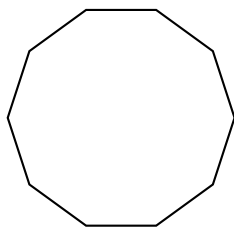
13.4.1 Hexagone



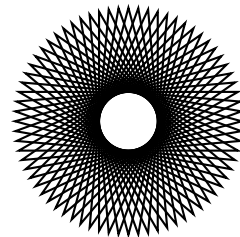
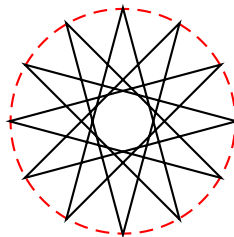
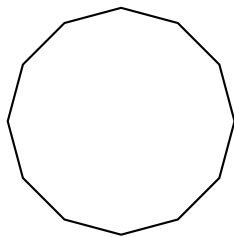
13.4.2 Octogone



13.4.3 Décagone

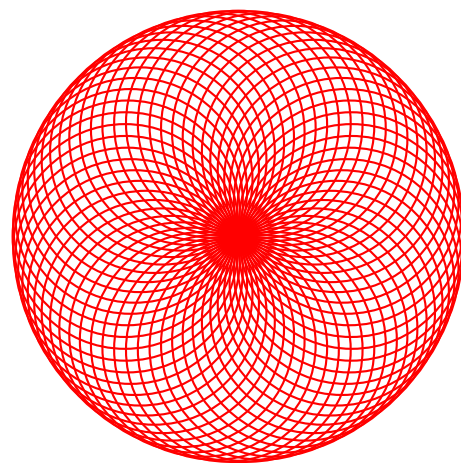
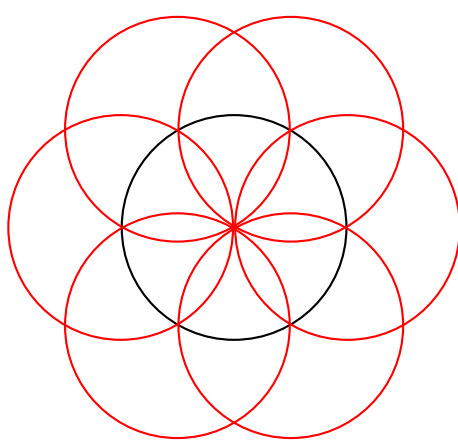


13.4.4 Dodécagone



13.5 Rosaces

On a déjà vu la figure de gauche à la page 55, et il y a peu de changements à effectuer pour obtenir la figure de droite :



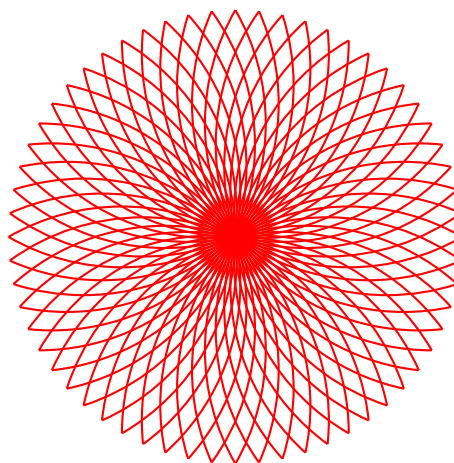
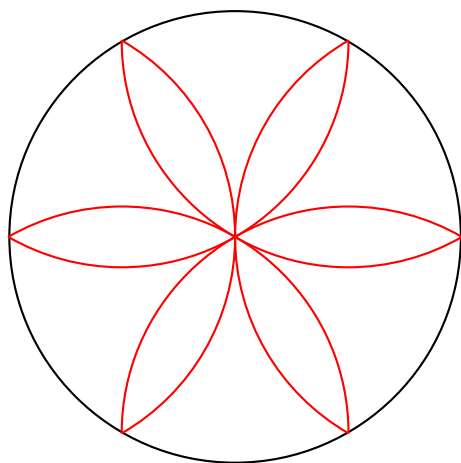
Voici les deux codes pour vous en persuader :

```
\psset{unit=0.5cm}
\def\xmin{-6.5} \def\xmax{6.5}
\def\ymin{-6.1} \def\ymax{6.1}
\begin{pspicture}
  (\xmin,\ymin)(\xmax,\ymax)
\pscircle(0,0){3}
\multido{\i=0+60}
  {6}
  {\pscircle[linecolor=red](3;\i){3}}
\end{pspicture}
```

```
\psset{unit=0.5cm}
\def\xmin{-6.5} \def\xmax{6.5}
\def\ymin{-6.5} \def\ymax{6.1}
\begin{pspicture}
  (\xmin,\ymin)(\xmax,\ymax)
%\pscircle(0,0){3}% pas de cercle noir
\multido{\i=0+6}
  {60}
  {\pscircle[linecolor=red](3;\i){3}}
\end{pspicture}
```

Au lieu de tracer 6 cercles dont les centres sont décalés de 60° , on en trace 60 dont les centres sont décalés de 6° .

Il faut un peu plus de travail pour obtenir ces deux dessins :



Dans la figure de gauche, on ne trace plus des cercles (`\pscircle`) mais des arcs de cercles (`\psarc`) ; mais pour cette dernière instruction, il faut donner l'angle de départ de l'arc, et l'angle d'arrivée. Et ces deux angles varient en fonction de la position du centre de l'arc sur le cercle noir.

Il faut également savoir que les arcs de cercles se tracent dans le sens trigonométrique : l'instruction `\psarc(abs,ord){rayon}{ang début}{ang fin}` va tracer un arc de cercle de rayon `rayon`, centré sur le point de coordonnées `(abs,ord)` et allant dans le sens trigonométrique de `ang début` à `ang fin`.

Au lieu des coordonnées cartésiennes, on utilise ici aussi un repérage polaire pour déterminer le centre ; le centre sera donc repéré par le couple `(rayon ; angle du centre)`.

Le tableau suivant donne l'angle de départ et l'angle d'arrivée pour chaque position du centre :

angle du centre	angle début	angle fin
0°	120°	240°
60°	180°	300°
120°	240°	360°
180°	300°	420°
240°	360°	480°
300°	420°	540°

Il faut donc trois variables entières, une pour l'angle du centre, une pour l'angle de départ de l'arc, une pour l'angle d'arrivée; ces trois variables prenant le même nombre de valeurs, une seule boucle suffit.

Le code du dessin de gauche est :

```
\psset{unit=1cm}
\def\r{3}%                                rayon
\def\xmin{-\r} \def\xmax{\r}
\def\ymin{-\r} \def\ymax{\r}
\begin{pspicture}(\xmin,\ymin)(\xmax,\ymax)
\pscircle(0,0){\r}%                       cercle noir
\multido{\i=0+60,\ix=120+60,\iy=240+60}%   variables
    {6}%                                   nombre de cercles
    {\psarc[linecolor=red](\r;\i){\r}{\ix}{\iy}}% arc
\end{pspicture}
```

Pour obtenir le dessin de droite, il suffit de remplacer la boucle `\multido` par celle-ci :

```
\multido{\i=0+6,\ix=120+6,\iy=240+6}
    {60}
    {\psarc[linecolor=red](\r;\i){\r}{\ix}{\iy}}
```

Au lieu de 6 arcs de cercle décalés de 60°, on dessine 60 arcs de cercle décalés de 6°.