

# TP1 : boucles

## 1. Préambule

Téléchargez Processing <https://processing.org/download/>

Ouvrir un nouveau sketch, entrer les 2 lignes de code suivantes (et exécuter)

```
ellipse(80,70,20,30) ;
```

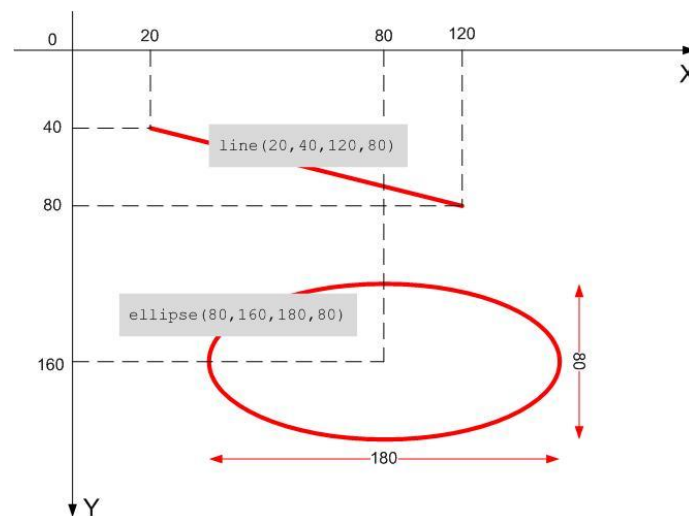
```
println(" hello ") ;
```

En plus du « hello » dans la console il est apparu dans une fenêtre de 100 sur 100 pixels, une ellipse blanche. Pour modifier la taille de la fenêtre, on utilise l'instruction `size(largeur, hauteur)` ;

- Essayez avec : `size(200,200)` ;
- Le fond de la fenêtre est par défaut gris. On va le changer en blanc avec l'instruction : `background(255,255,255)` ;

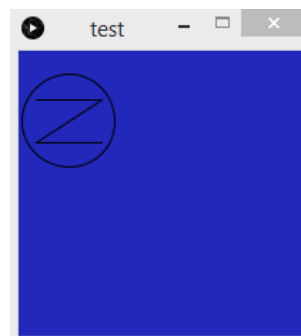
Couleur de trait : [https://www.processing.org/reference/stroke\\_.html](https://www.processing.org/reference/stroke_.html)

Couleur de remplissage : [https://www.processing.org/reference/fill\\_.html](https://www.processing.org/reference/fill_.html)



### Exercice 0 premier dessin

Dans une fenêtre de 200\*200 avec un fond bleu, tracer un Z entouré d'un cercle (peu importe la position dans la fenêtre). Par exemple :



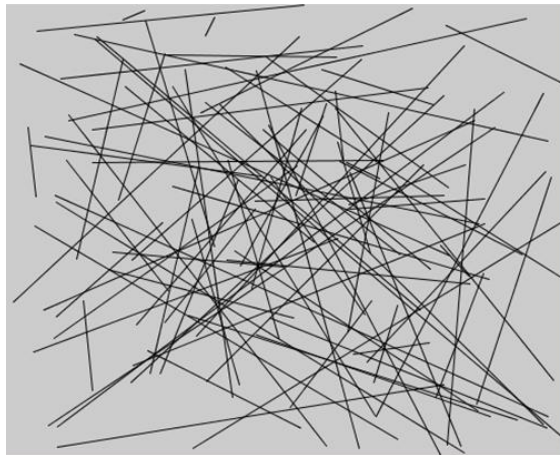
## 2. Exercices de TP

### Exercice 1 Segments aléatoires

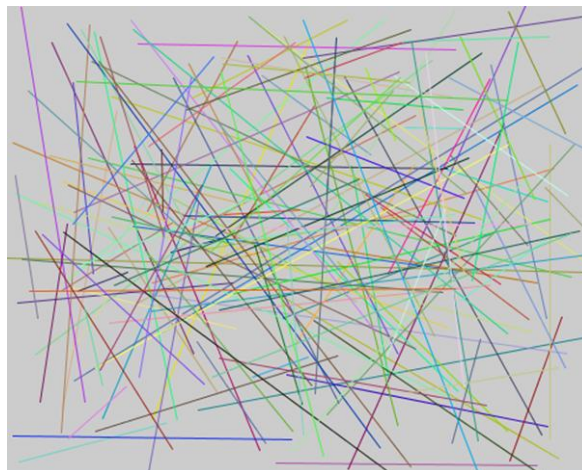
Commencer par exécuter le code ci-dessous :

```
int x, i, y;  
size(500, 400);  
for ( i=0; i<40; i++) {  
  x = (int)random(500);  
  y= (int)random(400);  
  line(0, 0, x, y);  
}
```

- a. A l'aide de la fonction random, tracer dans une fenêtre 500\*400
- 40 segments aux coordonnées aléatoires
  - qui restent inclus dans la fenêtre



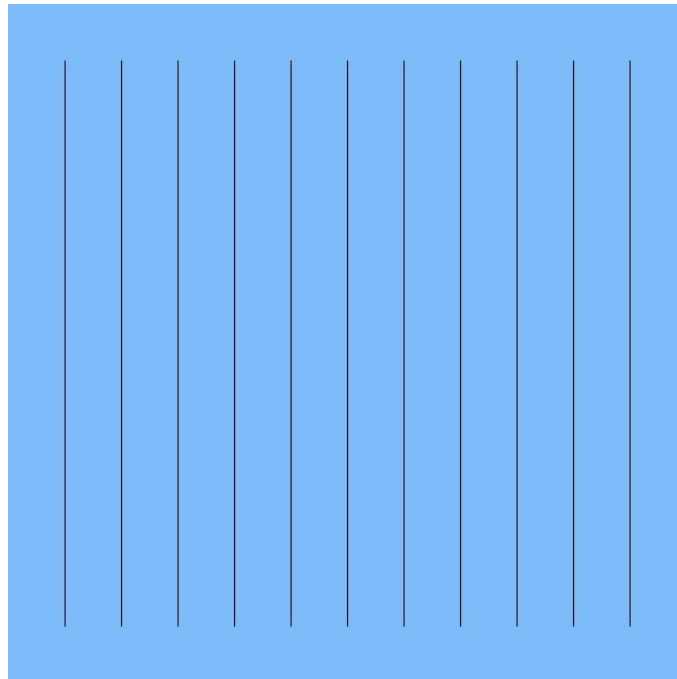
- b. Idem en choisissant au hasard la couleur des traits. Voir la fonction stroke : [https://processing.org/reference/stroke\\_.html](https://processing.org/reference/stroke_.html)



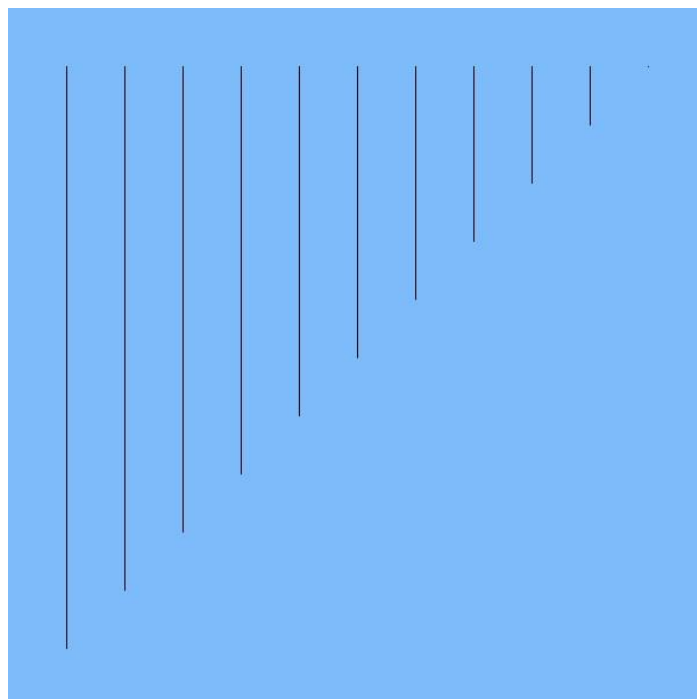
Pour une couleur aléatoire : stroke(random(255), random(255), random(255)) ;

## Exercice 2 : Lignes verticales

- a. Avec une boucle for, tracer 11 lignes verticales espacées de 50 pixels dans une fenêtre de 600 par 600



- a. Idem en faisant diminuer la hauteur des lignes

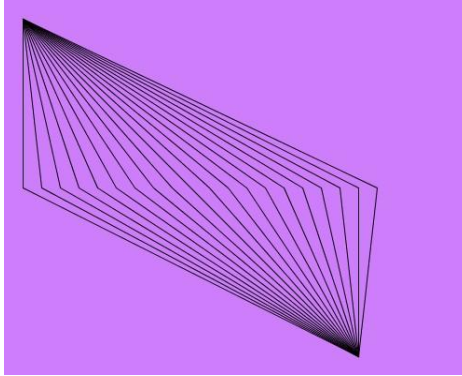


**Correction :**

Commencer par tracer les premiers motifs du dessin sans boucle	On peut réécrire ces 4 lignes avec une variable x
<pre>size(600, 600) ; line(50, 50, 50, 550) ; line(100, 50, 100, 550) ; line(150, 50, 150, 550) ; line(200, 50, 200, 550) ;</pre>	<pre>size(600, 600) ; int x=50 ; line(x, 50, x, 550) ; x=x+50 ; line(x, 50, x, 550) ; x=x+50 ; line(x, 50, x, 550) ; x=x+50 ; line(x, 50, x, 550) ; x=x+50 ;</pre>
On place désormais la partie du code à répéter dans une boucle avec un compteur i	Ou avec x comme compteur
<pre>size(600, 600) ; int x=50 ; for(int i = 0 ; i&lt;4 ; i=i+1){     line(x, 50, x, 550) ;     x=x+50 ; }</pre>	<pre>size(600, 600) ; for(int x=50 ; x&lt;600 ; x=x+50){     line(x, 50, x, 550) ; }</pre>

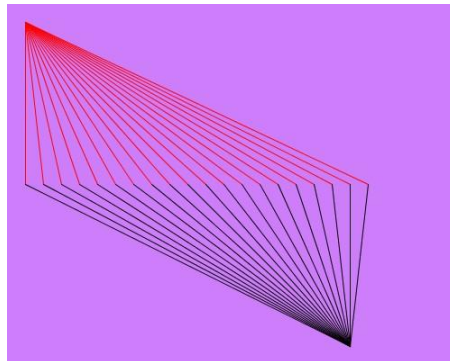
### Exercice 3 : Parallélogramme

- a. Reproduire le dessin ci-dessous en utilisant des boucles. La fenêtre fait 500\*400, le point en haut à gauche est à (20,20), celui en bas à droite à (380,380). La ligne des points médians est à  $y = 200$

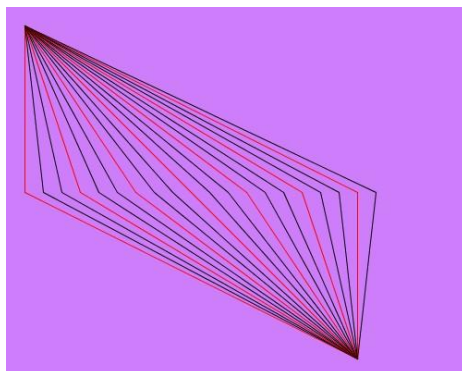


- b. Reproduire le dessin suivant (n'utiliser qu'une seule boucle)

Couleurs : les lignes de la partie supérieure sont rouges, celles de la partie inférieure sont bleues.



- c. Idem mais avec une ligne sur 3 en rouge (n'utiliser qu'une unique boucle) Contrainte : utiliser le modulo %



### Exercice 4 mini snake

Dans une fenêtre 500 par 400, reproduire le dessin.

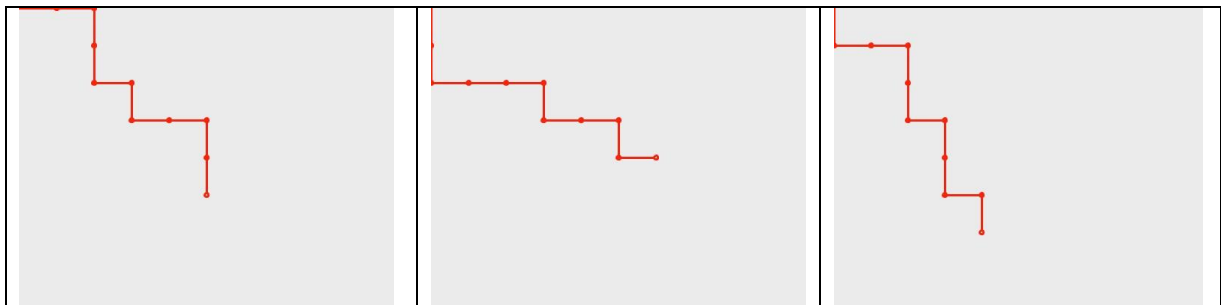
Principe de construction :

- le dessin commence dans le coin haut gauche de la fenêtre.
- A chaque iteration : le crayon dessine un petit cercle puis se déplace soit vers la droite, soit vers le bas de 50 pixels
- il y a 10 itérations au total

Remarque : pour le tirage aléatoire, on pourra s'inspirer du code suivant :

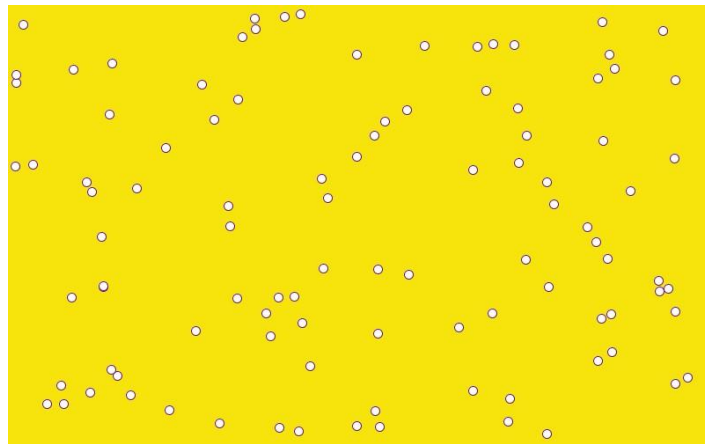
```
if (random(1)<0.5) {  
    println("a gauche");  
} else {  
    println("a droite");  
}
```

Exemple de sorties possibles :

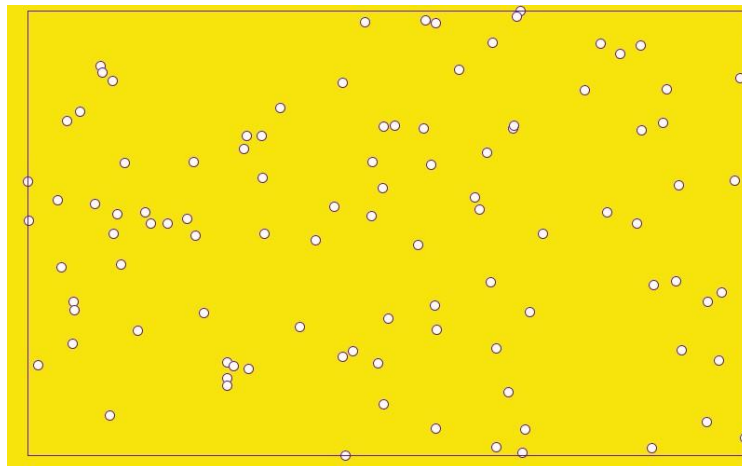


### Exercice 5 nuage de points

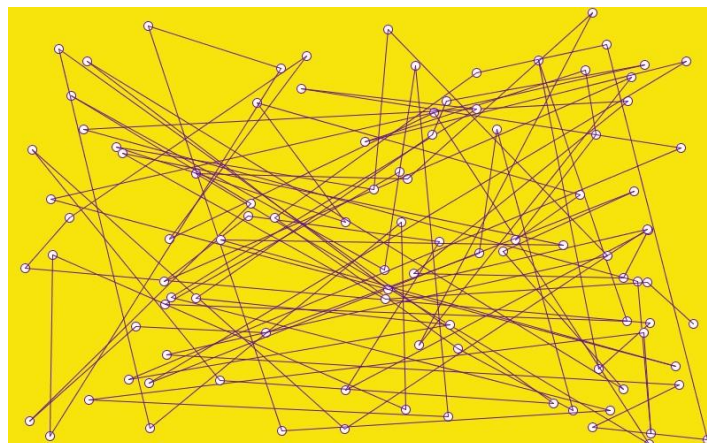
1. Dans une fenêtre 800 par 500, tracer 100 points de coordonnées aléatoires.



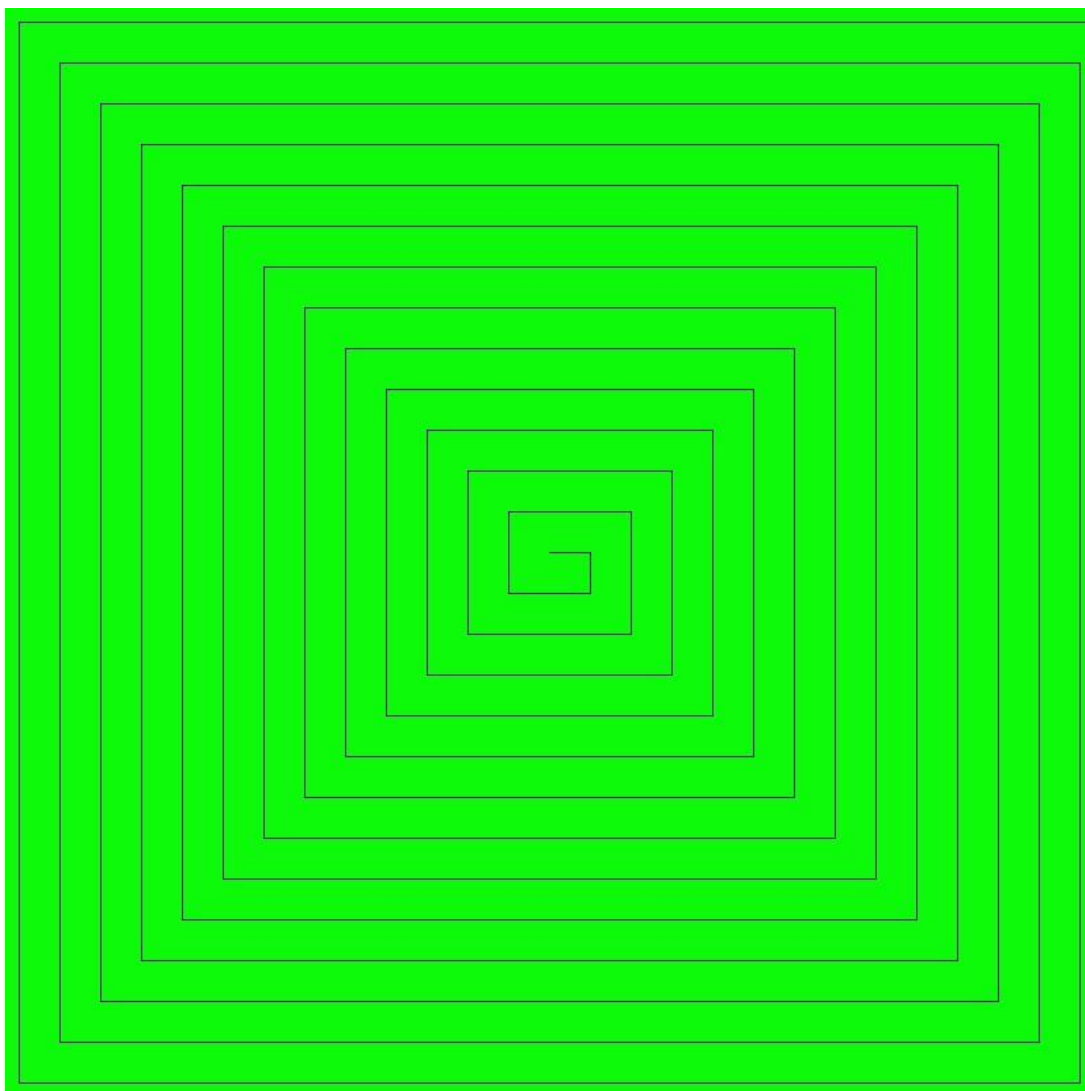
2. Encadrer ces points avec 4 lines



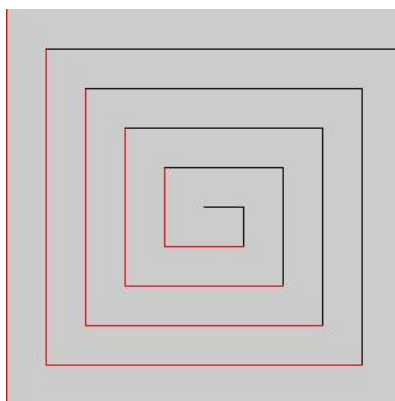
3. Relier ces points avec des lines



## Exercice 6 spirale carrée (partiel Janvier 2018)



Indice :

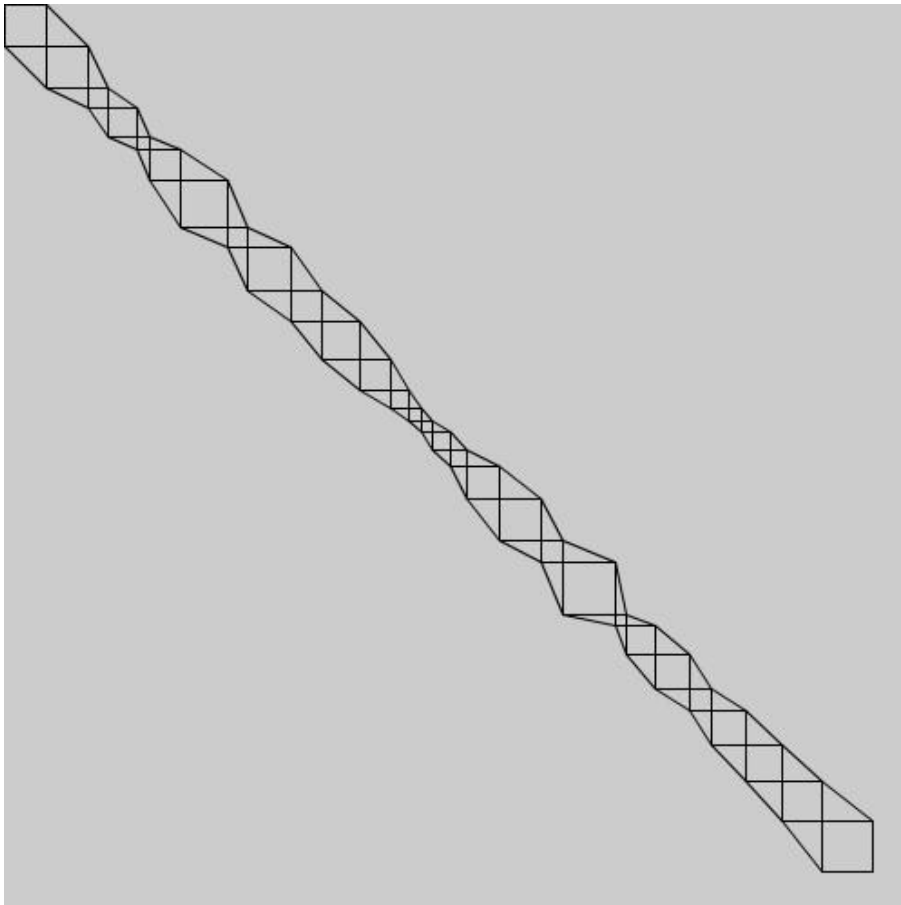




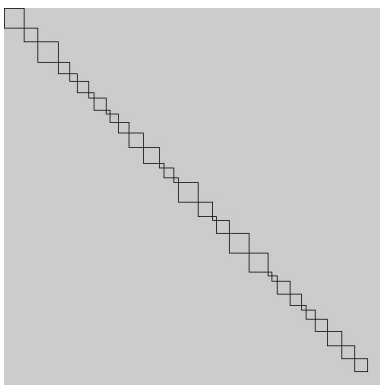
## Exercice 7

Reproduire le dessin ci-dessous dans une fenêtre 500\*500

- chaque carré a un côté entre 5 et 30 pixels
- le dessin doit rester dans la fenêtre



Indice :

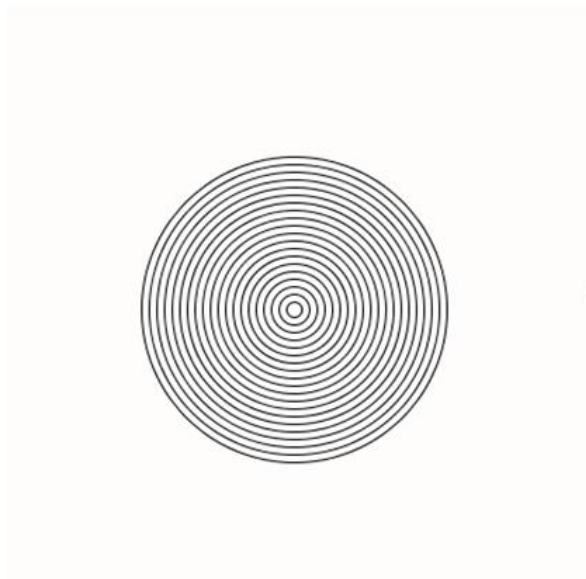


## Exercices supplémentaires

### Exercice 8

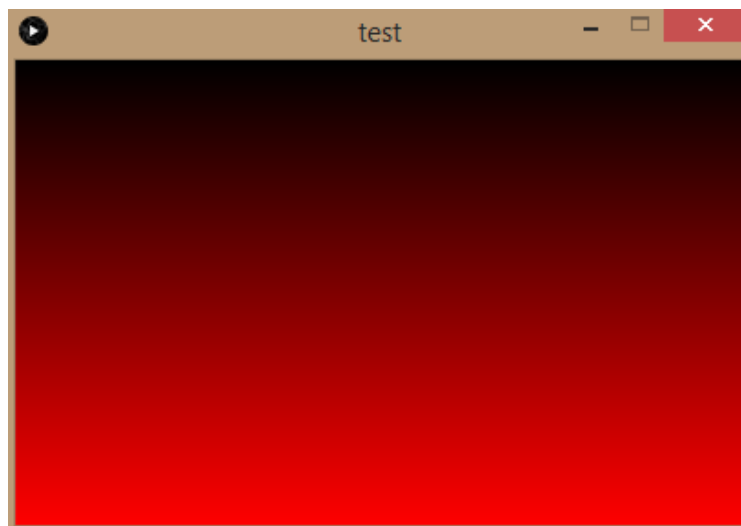
Reproduire le dessin ci-dessous en utilisant des boucles (20 cercles centrés en (200,200))

Utiliser la fonction ellipse ou circle ([https://processing.org/reference/circle\\_.html](https://processing.org/reference/circle_.html)) et (éventuellement) la fonction noFill() qui rend les fonds transparents (à placer avant )

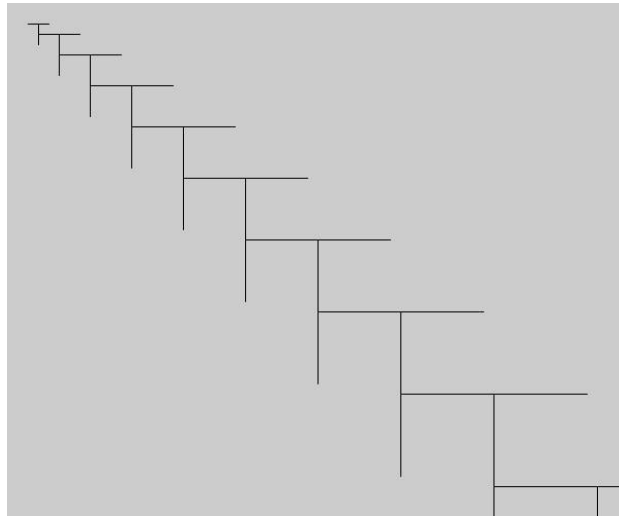


### Exercice 9 Dégradé

Dans une fenêtre de 400\*255, dessiner un dégradé de rouge comme une succession de lignes horizontales de couleurs différentes : noir en haut (0, 0, 0) et rouge en bas (255,0,0)



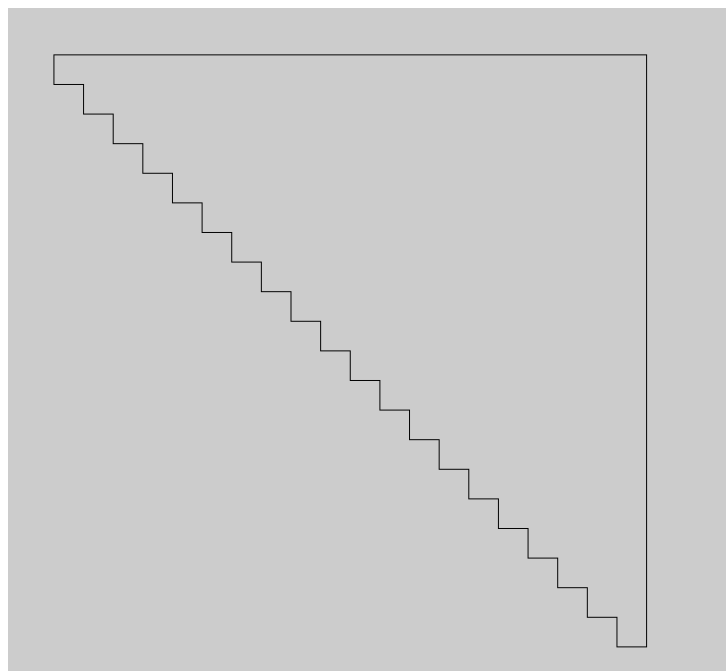
### Exercice 10 : Pas du patineur



Le dessin commence en haut à gauche en (20, 20). Le principe de construction est le suivant : on avance vers la droite sur 20 pixels, on revient au milieu et on repart vers le bas, toujours sur 20 pixels. On itère le processus en augmentant la distance de déplacement de 20 pixels à chaque iteration.

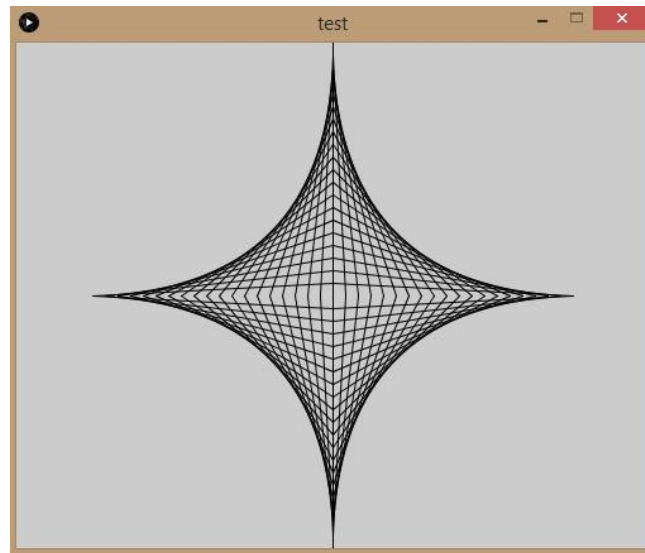
### Exercice 11 Escalier

La fenêtre fait 700 par 700. Le point en haut à gauche est en (50,50). Les marches font 30 pixels de côté. Le nombre de marches est un nombre aléatoire inférieur à 20 et supérieur à 3 = (int) random(3, 21).



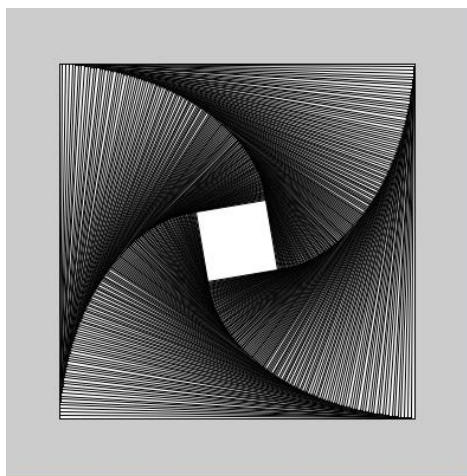
## Exercice 12

On pourra utiliser l'instruction `translate(width/2, height/2)` ; (à placer après le `size`) pour décaler l'origine du repère au centre de la fenêtre. Tracer le dessin ci-dessous (Noter que le dessin n'est composé que des lignes droites).



## Exercice 13 Tournicarré

Pour ce dessin, on pourra utiliser la fonction `rotate`<sup>1</sup>, `translate` et `rectMode(CENTER)`<sup>2</sup>



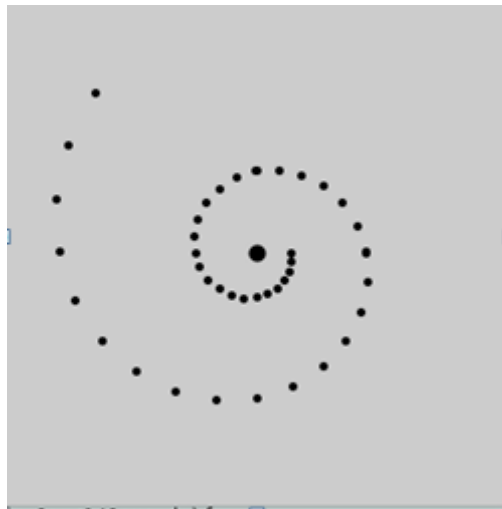
<sup>1</sup> [https://processing.org/reference/rotate\\_.html](https://processing.org/reference/rotate_.html) (ici on choisira `rotate(radians(0.4))`);

<sup>2</sup> [https://processing.org/reference/rectMode\\_.html](https://processing.org/reference/rectMode_.html)

## Exercice 14 Spirale

Tracer le dessin suivant dans une fenêtre de 300 par 300 :

- Le centre du repère est marqué par un point de 10 pixels d'épaisseur. Les points de la spirale font 5 pixels d'épaisseur.
- Le point le plus proche est à 20 pixels
- L'angle entre 2 points consécutifs est de  $15^\circ$
- L'augmentation de la distance entre 2 points est de 5%
- Le point le plus loin du centre est à 140 pixels.



Contrainte :

- On déplacera le centre du repère au milieu de la fenêtre
- Utiliser la fonction `point (...)`, `cos()` et `sin ()`
- Pour l'épaisseur des point, utiliser `strokeWeight(...)`