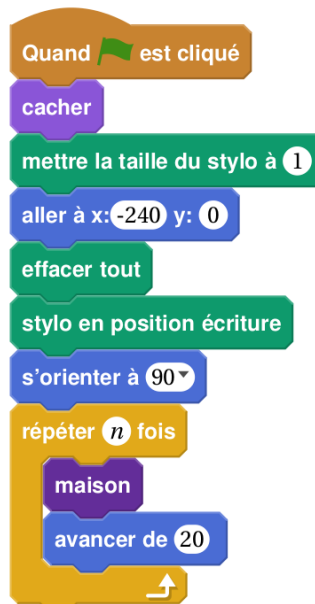




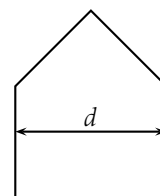
Voici des extraits du DNB 2017 : la version Scratch donnée au DNB, une proposition avec la tortue de Python *D'après les sujets de l'APMEP*

## 1. Centres Étrangers, juin 2017

Pour tracer une « rue », on a défini le tracé d'une « maison ».



- module turtle
- définition d'une fonction
- exitonclick

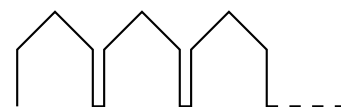


tracé de la « maison ».



Attention :

1. Vérifier que  $d$  est environ égal à 71 à l'unité près.
2. Un point dans une fenêtre d'exécution de votre programme a son abscisse qui peut varier de  $-240$  à  $240$  et son ordonnée qui peut varier de  $-180$  à  $180$ .  
Quel est le plus grand, nombre entier  $n$  que l'on peut utiliser dans le programme principal pour que le tracé de la « rue » tienne dans la fenêtre de votre ordinateur où s'exécute le programme ?  
*Vous pourrez tracer sur votre copie tous les schémas (à main levée ou non) qui auront permis de répondre à la question précédente et ajouter toutes les informations utiles (valeurs, codages, traits supplémentaires, noms de points...)*



```

1  # -*- coding:utf8 -*-
2  # python3
3
4  from turtle import *
5
6  def maison():
7      left(90)
8      forward(50)
9      right(45)
10     forward(50)
11     right(90)
12     forward(50)
13     right(45)
14     forward(50)
15     left(90)
16
17     hideturtle()
18     pensize(1)
19     setposition(-240,0)
20     clear()
21     pendown()
22     for k in range(7):
23         maison()
24         forward(20)
25
26     exitonclick()

```



## 2. Wallis et Futuma, décembre 2017

Notion de boucle et appel à une fonction créée dans le programme !

On souhaite tracer le motif en forme de losange.

Compléter le script du bloc Losange afin d'obtenir ce motif.

version Scratch

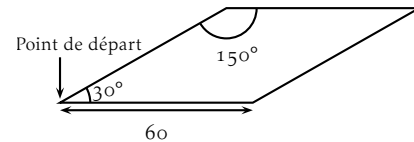
une version possible en Python

```

1  # -*- coding:utf8 -*-
2  # python 3
3
4  from turtle import *
5
6  def losange():
7      pendown()
8      forward(60)
9      left(30)
10     forward(60)
11     left(150)
12     forward(60)
13     left(30)
14     forward(60)
15     left(150)
16     penup()
    
```

- module turtle
- notion de fonction

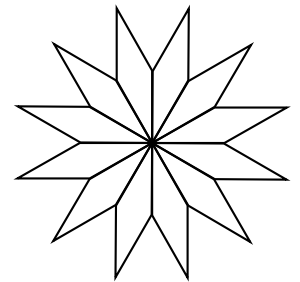
Le motif Losange



On souhaite réaliser la figure ci-dessous construite à partir du bloc Losange complété à la question 1.

On rappelle que l'instruction 's'orienter à 90° degrés' signifie que l'on se dirige vers la droite.

Parmi les instructions ci-dessous, indiquer sur votre copie, dans l'ordre, les deux instructions à placer dans la boucle ci-contre pour finir le script.



1.	tourner de 30 degrés
2.	Losange

3.	tourner de 150 degrés
4.	avancer de 600



```

1  # -*- coding:utf8 -*-
2  # python 3
3
4  from turtle import *
5
6  def losange():
7      pendown()
8      forward(60)
9      left(30)
10     forward(60)
11     left(150)
12     forward(60)
13     left(30)
14     forward(60)
15     left(150)
16     penup()
17
18     clear()
19     setposition(0,0)
20     right(90)
21     for i in range(12):
22         losange()
23         left(30)
24
25     mainloop()

```

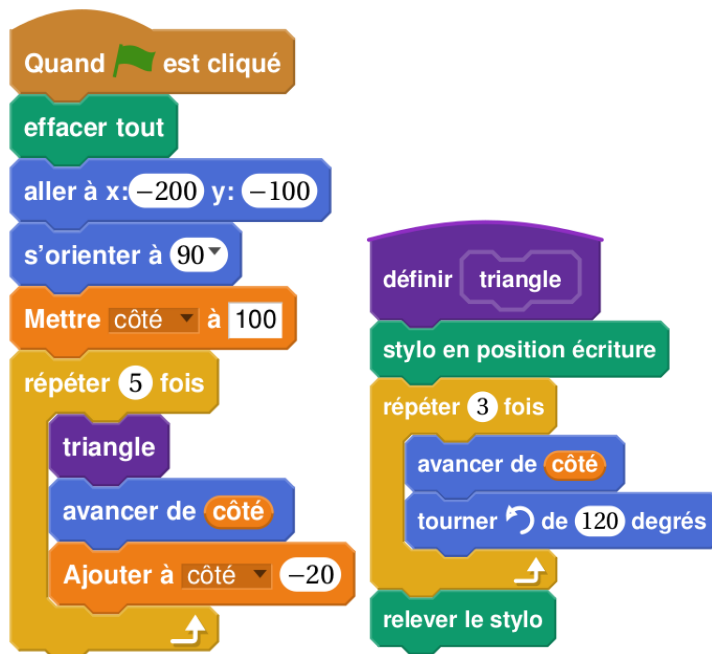
### 3. Métropole, juin 2017

On donne le programme suivant qui permet de tracer plusieurs triangles équilatéraux de tailles différentes.

Ce programme comporte une variable nommée « côté ». Les longueurs sont données en pixels.

- module turtle
- fonction avec une variable

On rappelle que l'instruction `s'orienter à 90` signifie que l'on se dirige vers la droite.



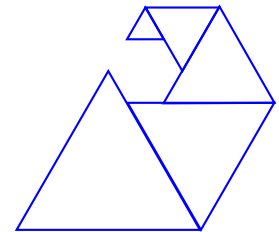


```

1  # -*- coding:utf8 -*-
2  # python3
3
4  from turtle import *
5
6  def triangle(cote):
7      """ construit un
8          ↪ triangle équilaté
9          ↪ ral
10         de côté <cote>
11         """
12         pendown()
13         for i in range(3):
14             forward(cote)
15             left(120)
16             penup()
17
18         clearscreen()
19         shape("turtle")
20         penup()
21         setposition(-200,-100)
22         cote=100
23         for i in range(8):
24             triangle(cote)
25             forward(cote)
26             cote = cote - 20
27
28     mainloop()

```

1. Quelles sont les coordonnées du point de départ du tracé ?
2. Combien de triangles sont dessinés par le script ?
3.
  - a) Quelle est la longueur (en pixels) du côté du deuxième triangle tracé ?
  - b) Tracer à main levée l'allure de la figure obtenue quand on exécute ce script.
4. On modifie le script initial pour obtenir la figure ci-contre. Indiquer le numéro d'une instruction du script **après laquelle** on peut placer l'instruction `tourner ↻ de 60 degrés` pour obtenir cette nouvelle figure.



#### 4. Metropole, septembre 2017

Voici trois figures différentes, aucune n'est à l'échelle indiquée dans l'exercice :

- module turtle
- variable globale

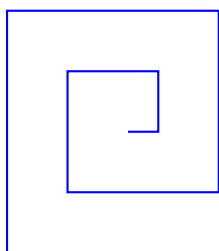


figure 1

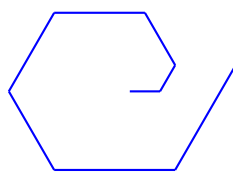


figure 2

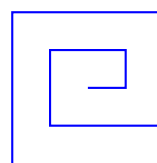
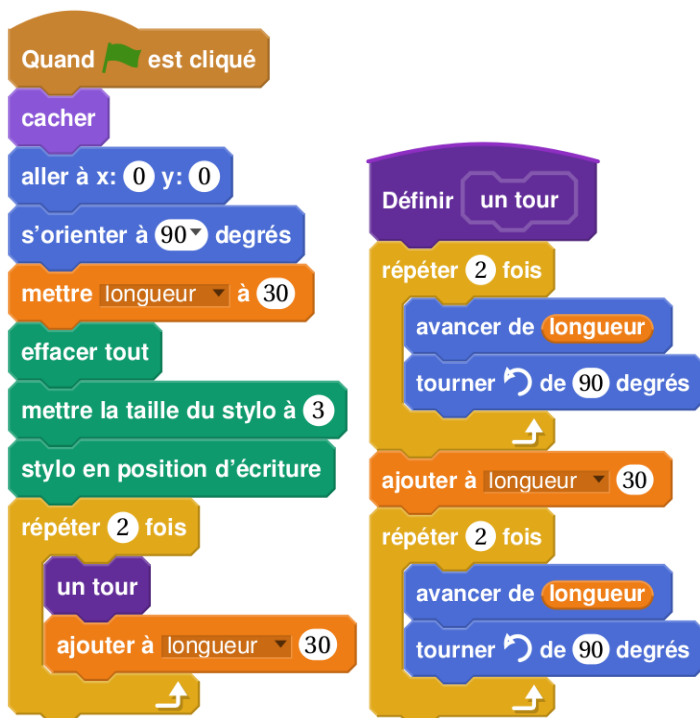


figure 3

Le programme ci-dessous contient une variable nommée « longueur ».



```

1  # -*- coding:utf8 -*-
2  # python3
3
4  from turtle import *
5
6  def tour():
7      global longueur
8      for k in range(2):
9          forward(longueur)
10         left(90)
11         longueur = longueur + 30
12     for k in range(2):
13         forward(longueur)
14         left(90)
15
16 hideturtle()
17 setposition(0,0)
18 longueur=30
19 clear()
20 pensize(3)
21 pendown()
22 for k in range(2):
23     tour()
24     longueur = longueur + 30
25
26 mainloop()

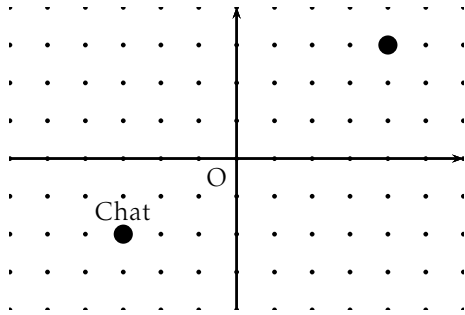
```

- Dessiner la figure obtenue avec le bloc « un tour » donné dans le cadre de droite ci-dessus, pour une longueur de départ égale à 30, étant orienté vers la droite avec le stylo, en début de tracé. On prendra 1 cm pour 30 unités de longueur, c'est-à-dire 30 pixels.
  - Comment est-on orienté avec le stylo après ce tracé? (aucune justification n'est demandée)
- Laquelle des figures 1 ou 3 le programme ci-dessus permet-il d'obtenir? Justifier votre réponse.
- Quelle modification faut-il apporter au bloc « un tour » pour obtenir la figure 2?

## 5. Amérique du Nord, juin 2017

L'image représente la position obtenue au déclenchement du bloc départ d'un programme de jeu.

- module turtle
- Notions de scripts se déroulant en parallèles!
- événements contrôlés au clavier



L'arrière-plan est constitué de points espacés de 40 unités.

Dans cette position, le chat a pour coordonnées  $(-120 \ -80)$ .

**Le but du jeu est de positionner le chat sur la balle.**

1. Quelles sont les coordonnées du centre de la balle représentée dans cette position ?
2. Dans cette question, le chat est dans la position obtenue au déclenchement du bloc départ.

Le script du lutin « chat » qui se déplace est donné ci-après.

- a) Expliquez pourquoi le chat ne revient pas à sa position de départ si le joueur appuie sur la touche  $\rightarrow$  puis sur la touche  $\leftarrow$ .
- b) Le joueur appuie sur la succession de touches suivante :  $\rightarrow \rightarrow \uparrow \leftarrow \downarrow$ .  
Quelles sont les coordonnées  $x$  et  $y$  du chat après ce déplacement ?
- c) Parmi les propositions de succession de touches ci-dessous, laquelle permet au chat d'atteindre la balle ?

Déplacement 1	Déplacement 2	Déplacement 3
$\rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow \uparrow \uparrow \uparrow \uparrow$	$\rightarrow \rightarrow \rightarrow \uparrow \uparrow \rightarrow \downarrow \leftarrow$	$\uparrow \rightarrow \uparrow \rightarrow \uparrow \rightarrow \rightarrow \downarrow \downarrow$

3. Que se passe-t-il quand le chat atteint la balle ?




---

```

1  # -*- coding:utf8 -*-
2  # python3
3
4  from turtle import *
5
6  def flecheG():
7      setx(xcor()-40)
8      appuiNimporteQuoi()
9
10 def flecheD():
11     setx(xcor()+80)
12     appuiNimporteQuoi()
13
14 def flecheH():
15     sety(ycor()+80)
16     appuiNimporteQuoi()
17
18 def flecheB():
19     sety(ycor()-40)
20     appuiNimporteQuoi()
21
22 def appuiNimporteQuoi():
23     if position() == (160,120):
24         write("je t'ai attrapé",
25             ↪False)
26         ontimer(depart,2000)
27
28 def depart():
29     clear()
30     hideturtle()
31     penup()
32     setposition(160,120)
33     dot(10,"red")
34     setposition(-120,-80)
35     showturtle()
36     pendown()
37
38
39 depart()
40 listen() #Pour "écouter"
41 onkeypress(flecheG, "Left") #
42     ↪Touche gauche
43 onkeypress(flecheD, "Right") #
44     ↪Touche droite
45 onkeypress(flecheH, "Up") #
46     ↪Touche haut
47 onkeypress(flecheB, "Down") #
48     ↪Touche bas
49 onkeypress(appuiNimporteQuoi) #
50     ↪Toutes les touches libres
51     ↪sont
52
53     #
54     ↪associées à appui_quelconque
55 mainloop()

```

---