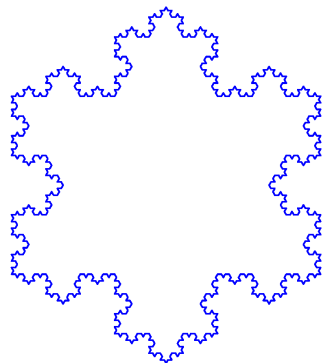


Helge Von-Koch (1870-1924) est un mathématicien suédois qui a donné son nom en 1904 à l'une des premières fractales, le flocon de Von-Koch. Il a décrit ce flocon dans un article intitulé "Sur une courbe continue sans tangente, obtenue par une construction géométrique élémentaire".



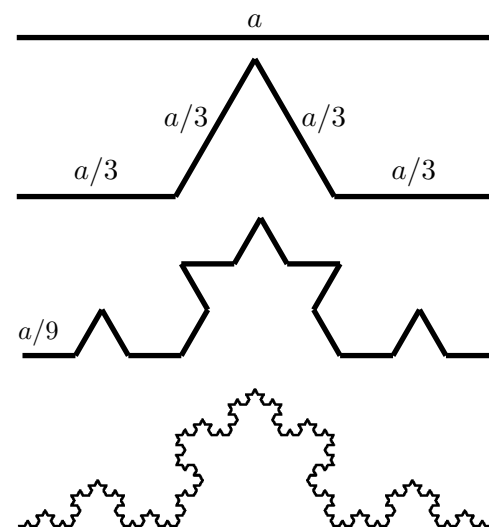
Ce travail dans un environnement de programmation Python avec par exemple l'éditeur Spyder.

## I - Méthode de construction :

- On commence par un segment de longueur  $a$  ;
- On coupe ce segment en 3 parties égales : le segment central est remplacé par un triangle équilatéral de côté  $a/3$  ;
- Chaque segment de longueur  $a/3$  est lui même découpé en trois parties égales (donc de longueur  $a/9$ ) : on remplace la partie centrale par un triangle équilatéral de côté  $a/9$  ;

• etc...

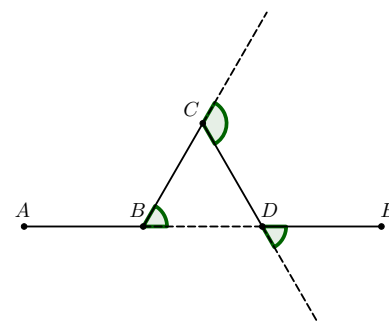
On décide à l'avance quand on doit s'arrêter.



## II - Les angles

### QUESTION 1

Compléter la figure ci-dessous en indiquant les mesures des angles en degré :



### III - Le module turtle

En Python le module `turtle` permet de réaliser des figures.

Voici l'instruction d'import du module `turtle`, et notre tortue se nommera `t` :

```
import turtle as t
```

#### QUESTION 2

Observer le code ci-dessous. Avant de le coder sur une machine, dessiner la figure qui doit être affichée.

```
import turtle as t

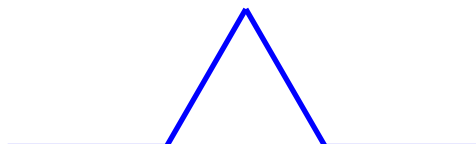
# déplace la tortue aux coordonnées
t.penup()
t.goto(-200, 200)
t.pendown()

# orientation initiale de la tête :
# vers la droite de l'écran
t.setheading(0)
# on cache la tortue
t.hideturtle()
# on accélère la tortue
t.speed(0)
# définition des caractéristiques du tracé
t.color('blue')
t.pensize(1)

# on trace la figure
t.forward(100)
t.left(90)
t.forward(100)
t.right(90)
t.forward(100)
t.right(90)
t.forward(100)
t.left(90)
t.forward(100)
```

#### QUESTION 3

Modifier le code ci-contre afin d'obtenir la figure ci-dessous :



#### QUESTION 4

Recopier et compléter l'algorithme récursif suivant afin de réaliser une des figure décrite dans la partie I :

##### Variables :

*n* : entier indiquant le nombre d'étape à réaliser

*taille* : la longueur du segment initial

##### Définition *flocon*(*n*, *taille*):

##### Si *n* = 0 alors

| Tracer le segment de longueur *cote*

##### Sinon

| Appeler la fonction *flocon* avec les paramètres ...

| Tourner la tortue ....

| ...

##### FinSi

##### FinDéfinition

#### QUESTION 5

Implémenter en Python la fonction *flocon*.

#### QUESTION 6

Écrire un programme Python utilisant la fonction *flocon* afin de construire les figures ci-dessous :

