

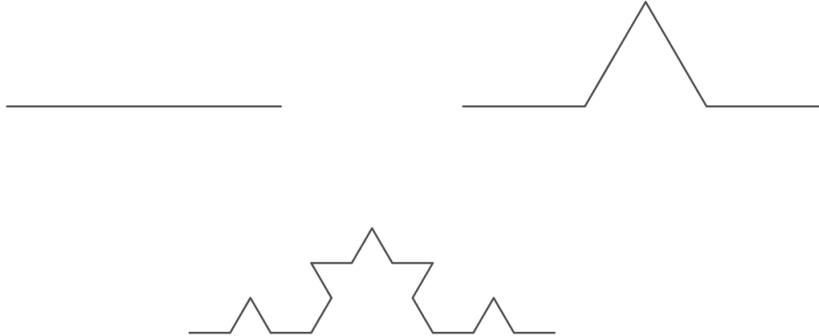


Région académique
PAYS DE LA LOIRE



Flocon de Koch :

On se propose d'effectuer la transformation suivante de façon répétée : Chaque segment de la figure est coupé en 3 puis la partie centrale est remplacée par deux segments de la même longueur. On obtient par exemple :



Le flocon de Koch est la figure obtenue en partant d'un triangle équilatéral et en répétant l'opération un grand nombre de fois. En voici, une programmation sous Python :

```
from math import*
from matplotlib.pyplot import*

def Koch(x1,x2,y1,y2): #Effectue la transformation sur un segment
    a1=x1+(x2-x1)/3 #Calcul des coordonnées des deux points de cassure
    a3=x1+2*(x2-x1)/3
    b1=y1+(y2-y1)/3
    b3=y1+2*(y2-y1)/3
    a2=a1+(a3-a1)/2-(b3-b1)*sqrt(3)/2 #Calcul des coordonnées de la pointe
    b2=b1+(a3-a1)*sqrt(3)/2+(b3-b1)/2
    return [a1,a2,a3],[b1,b2,b3] #renvoie les trois nouveaux points à insérer dans la ligne brisée

def TraceKoch(n,X,Y): # X,Y correspondent à la liste des coordonnées des points de la ligne brisée

    Xt=X #initialisation de la liste des points
    Yt=Y

    for i in range(n): #Passage d'une étape dans la création du flocon
        l=len(Xt)
        for j in range(0,4*(l-1),4):
            l1,l2=Koch(Xt[j],Xt[j+1],Yt[j],Yt[j+1]) #pour chaque point, création des nouveaux points

            Xt.insert(j+1,l1[0]) # insertion des nouveaux points dans la liste existante
            Xt.insert(j+2,l1[1])
            Xt.insert(j+3,l1[2])
            Yt.insert(j+1,l2[0])
            Yt.insert(j+2,l2[1])
            Yt.insert(j+3,l2[2])

    plot(Xt,Yt,"r-") # tracer du flocon
    axis([0.5,2.5,0.5,2.5]) # réglage de la zone d'affichage
    show()
    return

for n in range(7): # affichage des 6 premières étapes de construction du flocon
    TraceKoch(n,[1,3/2,2,1],[1,1+sqrt(3)/2,1,1])
```

