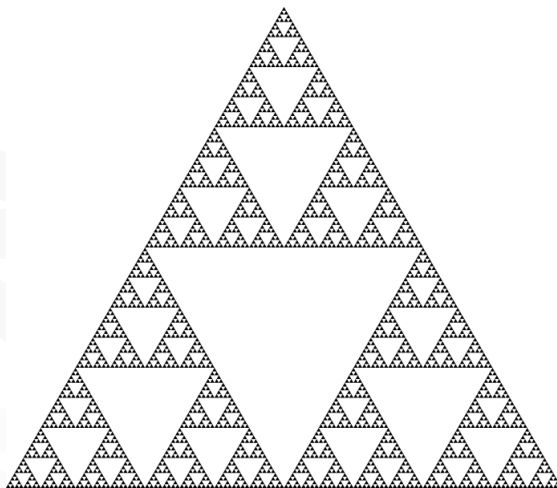


Rekurencja

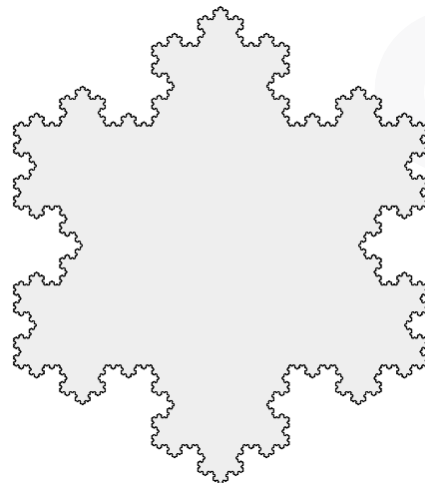
W przyrodzie spotykamy obiekty, które są podobne same do siebie – jeśli popatrzymy na cząstkę kalafiora, to wygląda ona jak cały kalafior. Rysunki powstające w podobny sposób nazywamy fraktalami. Jak je rysować? Najczęściej wykorzystujemy do tego rekurencję.



Rekurencja polega na rozwiązywaniu problemu w oparciu o rozwiązania tego samego problemu dla danych o mniejszych rozmiarach. Czyli jak nauczę się rysować kawałek kalafiora, to narysuję go całego. Istnieje wiele pięknych fraktali takich jak płatek Kocha czy trójkąt Sierpińskiego. Jednak rekurencję możemy wykorzystać także do tworzenia prostszych rysunków.



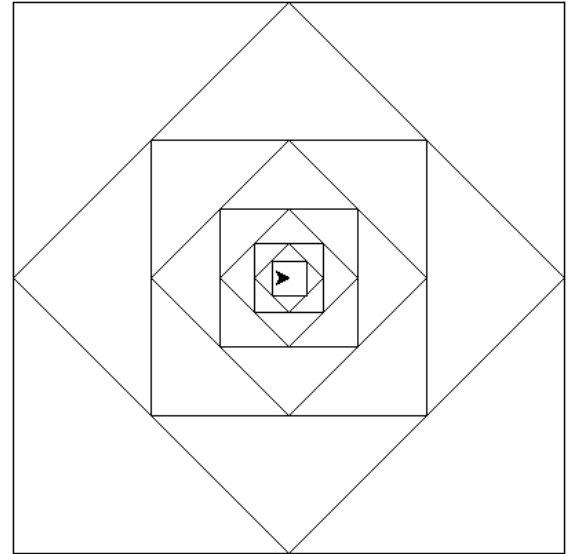
Trójkąt Sierpińskiego



Płatek Kocha

Kwadraty

Wyobraźmy sobie, że mamy narysować kwadraty takie jak na rysunku obok i wykorzystać do tego rekurencję. Długość boku największego kwadratu wynosi 400, a długość boku najmniejszego kwadratu wynosi co najmniej 20. Rozwiązanie tego problemu polega na narysowaniu największego kwadratu. Dalsza część rozwiązania to to samo zadanie (rysowanie kwadratu) tylko bok kwadratu jest mniejszy i wynosi $\frac{bok}{\sqrt{2}}$.



Tak może wyglądać rekurencyjne rozwiązanie tego problemu:

```

7% *kwadrat.py - D:\Python\programy1\kwa... - □ ×
File Edit Format Run Options Windows Help
from turtle import *
from math import sqrt

def kwad(a):
    for i in range(4):
        fd(a); lt(90)

def kwady(a):
    if a<20:
        return
    kwad(a)
    fd(a/2); lt(45)
    kwady(a/sqrt(2))

def kwadrat():
    a=400
    pu()
    bk(a/2); lt(90)
    bk(a/2); rt(90)
    pd()
    kwady(a)
    pu(); home(); pd()

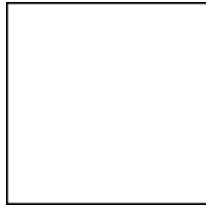
kwadrat()
Ln: 24 Col: 10

```

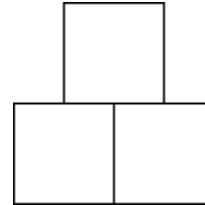
Dziwne schody

Wyobraźmy sobie, że rysujemy rekurencyjnie schody w następujący sposób:

- Schody stopnia 1 to kwadrat.
- Schody stopnia 2 to trzy kwadraty o połowę mniejsze.


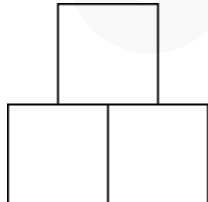
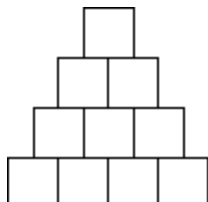


schody 1



schody 2

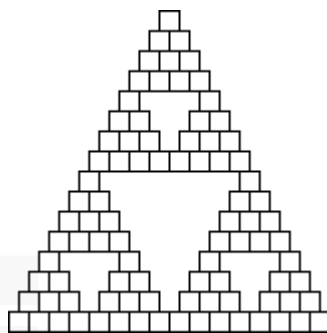
Postępując zgodnie z powyższym opisem schodów stopnia 2 i 1 moglibyśmy napisać następujące polecenia:

Zapis w Pythonie	Efekt na ekranie
<pre>#rysujemy kwadrat - schody stopnia 1 for i in range(4): fd(bok); rt(90)</pre>	
<pre>#rysujemy 3 kwadraty - schody stopnia 2 kwadrat(bok/2) pu(); fd(bok/2); rt(90); fd(bok/4); lt(90); pd() kwadrat(bok/2) pu(); bk(bok/2); rt(90); fd(bok/4); lt(90); pd() kwadrat(bok/2) pu(); lt(90); fd(bok/2); rt(90); pd()</pre>	
<pre>#rysujemy schody stopnia 3 #rysujemy schody stopnia 2 o wielkości bok/2 schody(bok/2, 2) pu(); fd(bok/2); rt(90); fd(bok/4); lt(90); pd() schody(bok/2, 2) pu(); bk(bok/2); rt(90); fd(bok/4); lt(90); pd() schody(bok/2, 2) pu(); lt(90); fd(bok/2); rt(90); pd()</pre>	

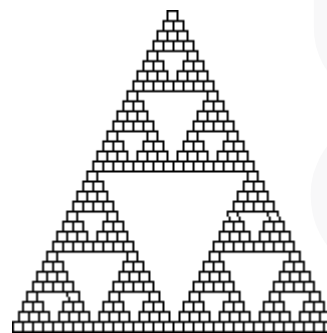
A tak wyglądałaby ogólna procedura rysująca schody:

```
def schody(bok, n):
    if (n==1):
        kwadrat(bok)
        return
    else:
        schody(bok/2, n-1)
        pu(); fd(bok/2); rt(90); fd(bok/4); lt(90); pd()
        schody(bok/2, n-1)
        pu(); bk(bok/2); rt(90); fd(bok/4); lt(90); pd()
        schody(bok/2, n-1)
        pu(); lt(90); fd(bok/2);rt(90); pd()
```

Rysunki otrzymane tą metodą dają bardzo ciekawe, choć może nieco inne niż oczekiwaliśmy, rezultaty:



schody 5



schody 6