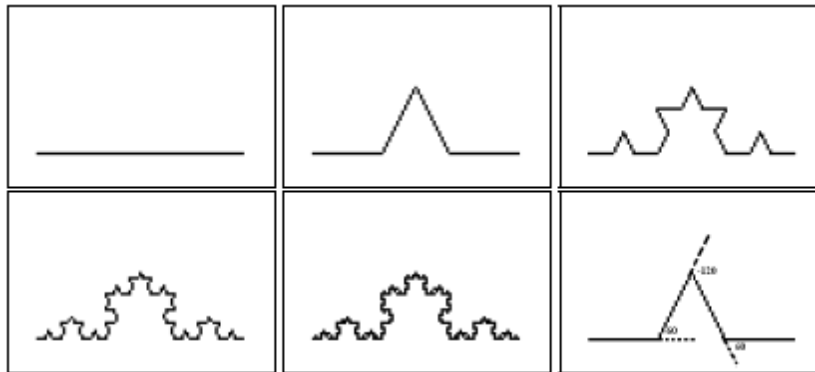


Die Kochkurve

turtle1.py, koch.py, koch2.py, koch3.py

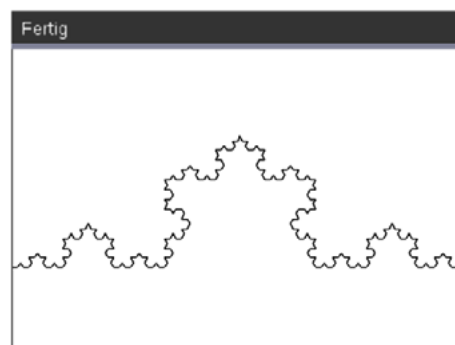
Wir wollen nun mit unserer Turtle-Programmiersprache Fraktale erzeugen. Dazu gehen wir von einer Strecke aus. Diese wird durch ein bestimmtes Muster ersetzt. Jede Strecke in diesem Muster wird wieder durch eine verkleinerte Form dieses Musters ersetzt. Diese Prozedur wird wiederholt. Das Ergebnis ist ein verwinkeltes Fraktal.



```

1.6 1.7 1.8 *PyKurz RAD 10/10
koch.py
from turtle import*
p,n=[0],4
r=[0,60,0,-120,0,60,0]
for i in range(n):
  np=[]
  for c in p:
    if c==0:np.extend(r)
    else:np.append(c)
  p=np[:]
turtle(p,-10,-3,s=.25)

```

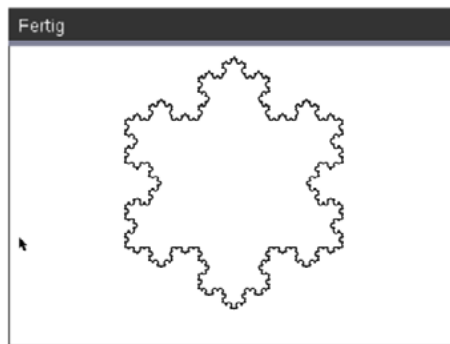


Hinweis: `liste1.extend(liste2)` ist ein – nicht dokumentierter – Listenbefehl, der die Liste `liste1` um die Elemente der `liste2` erweitert, während `liste1.append(element)` das element in die `liste1` aufnimmt. (Nicht `liste1.append([element])`)

- Zuerst wird unser eben erzeugtes Modul turtle (2.2.29) aufgerufen und wir geben den Beginn p als Strecke vor. Das Muster r wird viermal ersetzt. Daher führen wir in einer Schleife $n = 4$ Iterationen durch. Jedes Mal ist die neue Figur np anfänglich leer. Für jedes c der jeweils letzten Figur gibt es zwei Alternativen:
- Wenn $c = 0$ (eine Strecke), wird das Muster r der neuen Figur angefügt, anderenfalls nur die Drehung um den ausgeführt.
- Im nächsten Iterationsschritt wird der Inhalt der neuen Figur np nach p kopiert.
- Zum Schluss wird `turtle()` aufgerufen, wobei wir die Figur p beginnend im Punkt $(-10,-3)$ mit einer Schrittgröße $s = 0,25$ zeichnen.

Wenn du nicht mit einer Strecke beginnst, sondern mit der nächsten Figur, erhältst du die Koch-Schneeflocke oder auch Koch-Insel.

$120,0,-120,0,-120,0$

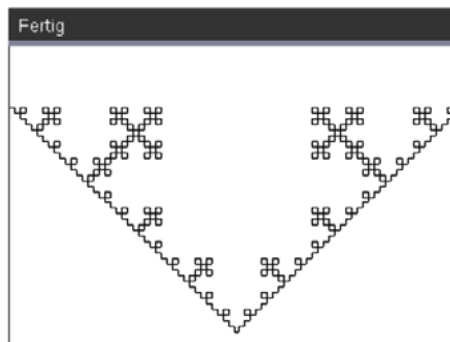


Welches Fraktal ergibt sich, wenn du mit einer Strecke beginnst und das Muster

$0,120,0,-120,0,-120,0,120,0$

anwendest? (Startpunkt und s müssen angepasst werden!)

Auch dieses Muster ist sehr bekannt:



Versuche auch, eigene Grundmuster zu entwerfen. Vielleicht gelingen dir schöne Fraktale!