



# 数据结构与算法（Python版）

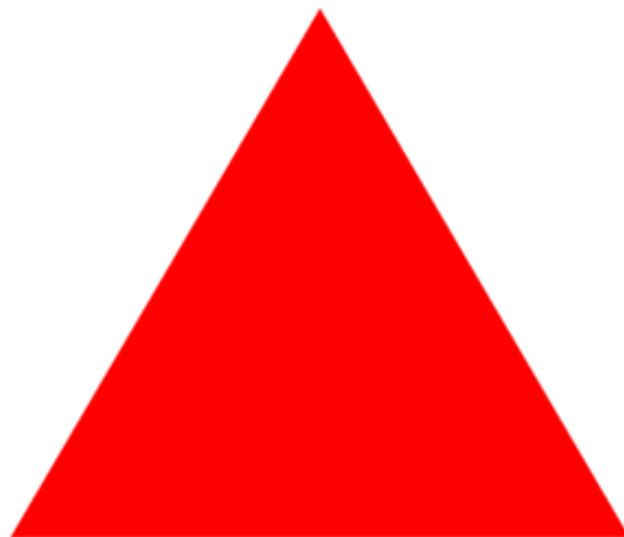
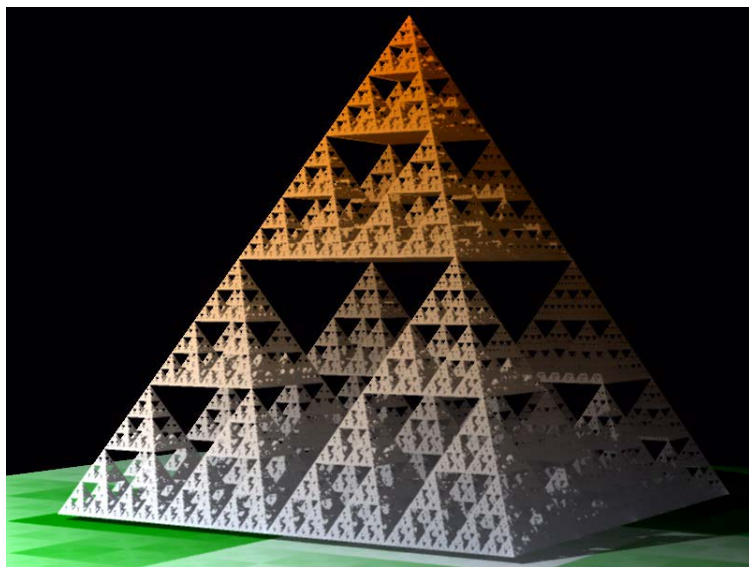
## 递归可视化：谢尔宾斯基三角形

陈斌 北京大学 [gischen@pku.edu.cn](mailto:gischen@pku.edu.cn)

# 谢尔宾斯基Sierpinski三角形

❖ **分形构造**，平面称谢尔宾斯基三角形，立体称谢尔宾斯基金字塔

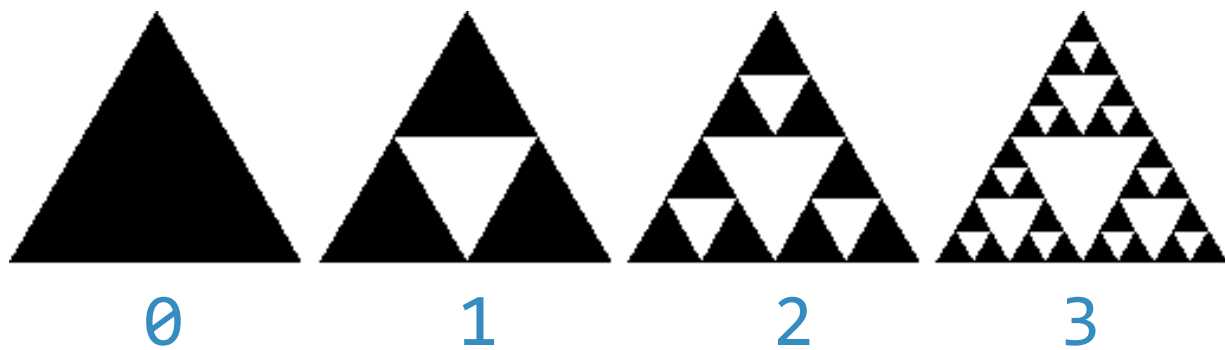
实际上，真正的谢尔宾斯基三角形是完全不可见的，其面积为0，但周长无穷，是介于一维和二维之间的分数维（约1.585维）构造。



# 谢尔宾斯基三角形：作图思路

❖ 根据**自相似特性**，谢尔宾斯基三角形是由**3个尺寸减半的谢尔宾斯基三角形按照品字形拼叠而成**

由于我们**无法真正做出谢尔宾斯基三角形**（ $\text{degree} \rightarrow \infty$ ），只能**做degree有限的近似图形**。

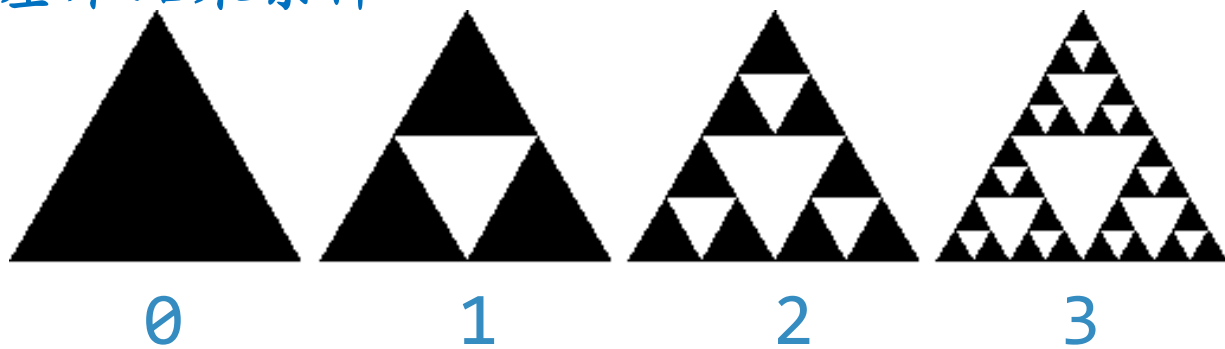


# 谢尔宾斯基三角形：作图思路

❖ 在degree有限的情况下， $\text{degree}=n$ 的三角形，是由3个 $\text{degree}=n-1$ 的三角形按照品字形拼叠而成

同时，这3个 $\text{degree}=n-1$ 的三角形边长均为 $\text{degree}=n$ 的三角形的一半（规模减小）。

当 $\text{degree}=0$ ，则就是一个等边三角形，这是递归基本结束条件



# 谢尔宾斯基三角形：代码

等边三角形

最小规模, 0直接退出

减小规模:  
getMid边长减半

调用自身, 左上右次序

```
1 import turtle
2
3 def sierpinski(degree, points):
4     colormap = ['blue', 'red', 'green', 'white', 'yellow', 'orange']
5     drawTriangle(points, colormap[degree])
6     if degree > 0:
7         sierpinski(degree - 1,
8                     {'left':points['left'],
9                      'top':getMid(points['left'], points['top']),
10                     'right':getMid(points['left'], points['right'])})
11        sierpinski(degree - 1,
12                    {'left':getMid(points['left'], points['top']),
13                     'top':points['top'],
14                     'right':getMid(points['top'], points['right'])})
15        sierpinski(degree - 1,
16                    {'left':getMid(points['left'], points['right']),
17                     'top':getMid(points['top'], points['right']),
18                     'right':points['right']})
```

# 谢尔宾斯基三角形：代码

```
20 def drawTriangle(points, color):
21     t.fillcolor(color)
22     t.penup()
23     t.goto(points['top'])
24     t.pendown()
25     t.begin_fill()
26     t.goto(points['left'])
27     t.goto(points['right'])
28     t.goto(points['top'])
29     t.end_fill()
30
31 def getMid(p1, p2):
32     return ( (p1[0] + p2[0]) / 2, (p1[1] + p2[1]) / 2 )
33
34 t = turtle.Turtle()
35
36 points = {'left':(-200, -100),
37           'top':(0, 200),
38           'right':(200, -100)}
39 sierpinski(5, points)
40
41 turtle.done()
```

绘制等边三角形

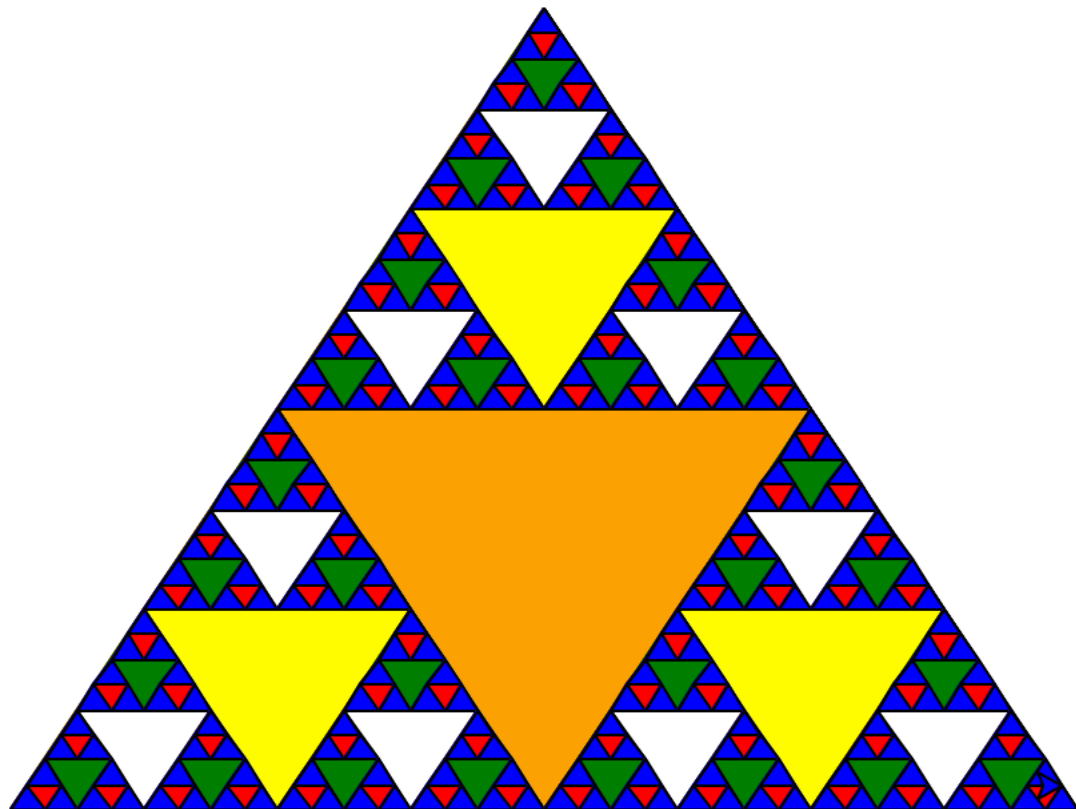
取两个点的中点

外轮廓三个顶点

画degree=5的三角形



# degree=5的三角形



# 数据结构与算法（Python版）

