



```
1 import math
2 import matplotlib.pyplot as plt
3 import matplotlib.patches as pts
4 import matplotlib.animation as animation
5 from matplotlib import rc
```

```
1 #一つの三角形を塗りつぶす
2 def drawTriangle(center, radius):
3     return pts.RegularPolygon(center, 3, radius, fill=True, color='blue')
```

```
1 #再帰的に三角形を生成
2 def oneStep(center, radius, iter, ax):
3     if iter == 0:
4         ax.add_patch(drawTriangle(center, radius))
5         return
6     #サイズが1/2の三角形へ分割
7     r = radius/2.
8     p0 = (center[0]-math.sqrt(3)*r/2, center[1]-r/2)
9     oneStep(p0, r, iter-1, ax)
10    p1 = (center[0]+math.sqrt(3)*r/2, center[1]-r/2)
11    oneStep(p1, r, iter-1, ax)
12    p2 = (center[0], center[1]+r)
13    oneStep(p2, r, iter-1, ax)
```

```
1 fig = plt.figure(figsize=(10, 10))
2 plt.xlim(-10, 10)
3 plt.ylim(-10, 10)
4 r = 10
5 center = (0, -1)
6 imgs=[]
7 tm = 8
8 for i in range(tm):
9     ax = fig.subplots()
10    ax.set_xlim(-10, 10)
11    ax.set_ylim(-10, 10)
12    oneStep(center, r, i, ax)
13    im = ax.get_children()
14    imgs.append(im)
15 ani = animation.ArtistAnimation(fig, imgs, interval=1000)
16 rc('animation', html='jshtml')
17 ani
```