```
0  # -*- coding: utf-8 -*-
1  """
2  Created on Tue Apr 14 20:10:51 2015
3
4  @authors: Group 10
5  """
6
7  import numpy as np
8  import matplotlib.pyplot as plt
9  from math import sqrt
10 import scitools
11 def Heighway_dragon(z=0, n=15):
12     z = np.vstack((0,1)) # generate vertical array
13
14     for x in range(n):
15         w = z[::-1] # finds the reverse of the array
16
17         z = np.vstack((((z+(z*1j))/2.),1-((w-(w*1j))/2.)))# generates complex
               number thus real and imaginary aspect
18
19        # print w # prints the value of w
20     return z
21
22 def plotH_dragon(z=0, n=5):
23     for ei in range(0,n):
24         plt.figure()
25         z = Heighway_dragon(z, ei)# stores the Heighway_dragon values in z
26
27         plt.plot(z.real,z.imag)# plots value of real against its corresponding
               imaginary value
28
29         plt.axis('equal')
30         plt.title('Heighway Dragon Curve  (iteration = '+ str(ei+1)+')')
31         if(ei>9):
32             plt.savefig("dragon_"+"9"+str(ei)+".png")
33         else:
34             plt.savefig("dragon_"+str(ei)+".png")
35         plt.show()
36         #putting all the figures together for animation which will open with a
               browser
37     scitools.easyviz.movie("dragon_*.png", encoder = "html", output_file = "
           Heighway_dragon.html", fps = 2)
38     #call the function to plot the graphs, save them as images and animate them
39 plotH_dragon()
```

```
Found 5 files of the format dragon_*.png.

Making HTML code for displaying dragon_0.png, dragon_1.png, dragon_2.png,
    dragon_3.png, dragon_4.png


movie in output file Heighway_dragon.html
```

```
37
38 #question 2
39 def twin_dragon(z=0, n=15):
40     z = np.vstack((0, 1,1-1j))
```

```
    for x in range(n):
        w = z[::-1]
        z = np.vstack((((w+(w*1j))/2.),1-((z+(z*1j))/2.)))
    return z

def plot_twin_dragon(z=0, n=5):
    for ei in range(n):
        plt.figure()
        z = twin_dragon(z, ei)
        #finding the length of the array z
        r = len(z)
        """using the length to slice the array to plot certain indexes against
           each other on
        one side and the others at the back of the first graph"""
        plt.plot(z.real[:r/2],z.imag[:r/2],z.real[r/2:], z.imag[r/2:])
        plt.axis('equal')
        plt.title('Twin Dragon   (iteration = '+str(ei+1)+')')
        if(ei>9):
            plt.savefig("t_dragon_"+"9"+str(ei)+".png")
        else:
            plt.savefig("t_dragon_"+str(ei)+".png")
        plt.show()
    scitools.easyviz.movie("t_dragon_*.png", encoder = "html", output_file = "
       twin_dragon.html", fps = 2)
    #call the function to plot the graphs, save them as images and animate the
       twin dragon
plot_twin_dragon()
```

```
Found 5 files of the format t_dragon_*.png.

Making HTML code for displaying t_dragon_0.png, t_dragon_1.png, t_dragon_2.png
    , t_dragon_3.png, t_dragon_4.png


movie in output file twin_dragon.html
```

```

#
#
def terdragon(z=0, n=15):
    l1= (1/2.)-(1j/(2*(sqrt(3))))
    l2= (1/2.)+(1j/(2*(sqrt(3))))
    z = np.vstack((0, 1))
    for x in range(n):
        z=np.vstack(((l1*z),((z*1j)/(sqrt(3)))+l1,(l1*z)+l2))
    return z

def plot_terdragon(z=0, n=5):
    for ei in range(n):
        plt.figure()
        z = terdragon(z, ei)
        #finding the length of the array z
        r = len(z)
        """using the length to slice the array into three groups of points to
           plot certain indexes against each other on
        one side,the middle and the last third against the middle graph"""
        plt.plot(z.real[:r/3],z.imag[:r/3],z.real[r/3:2*r/3],z.imag[r/3:2*r/3],z
           .real[2*r/3:],z.imag[2*r/3:])
```

```
85          plt.axis('equal')
86          plt.title('Terdragon Curve  (iteration = '+str(ei+1)+')')
87          if(ei>9):
88              plt.savefig("terdragon_"+"9"+str(ei)+".png")
89          else:
90              plt.savefig("terdragon_"+str(ei)+".png")
91          plt.show()
92      scitools.easyviz.movie("terdragon_*.png", encoder = "html", output_file = "
            terdragon.html", fps = 2)
93      #call the function to plot the graphs, save them as images and animate the
            terdragon
94  plot_terdragon()
```

```
Found 5 files of the format terdragon_*.png.

Making HTML code for displaying terdragon_0.png, terdragon_1.png, terdragon_2.
    png, terdragon_3.png, terdragon_4.png


movie in output file terdragon.html
```