Fractal Assignment Report

Bijita Bose (22PHPH10), Subhadip Santra (22PHPH12), Abhishek Thakur (22PHPH05)

1 Introduction

1.1 What is Fractal?

Fractals are complex geometric shapes or patterns that repeat themselves at different scales and are characterized by their self-similarity. They are created by applying simple mathematical rules or equations repeatedly to create complex and intricate designs.
Fractals are found in nature, such as in snowflakes, tree branches, coastlines, and even in the structure of the human lungs. They have also been used in computer graphics, art, and architecture, among other fields. Fractals are a fascinating subject in mathematics and have practical applications in fields like computer science, physics, and biology.

1.2 Why do we need fractals?

Studying fractals can be beneficial for various reasons, including:

1. Understanding natural phenomena: Fractals are found in many natural phenomena, such as the branching patterns of trees and blood vessels, the shapes of coastlines, and the structure of snowflakes. By studying fractals, we can gain a deeper understanding of these natural phenomena and the underlying mathematical principles that govern them.

2. Developing mathematical skills: Fractals provide a rich area of study in mathematics that involves geometry, topology, algebra, and calculus. Studying fractals can help develop mathematical skills such as problem-solving, logical reasoning, and mathematical modeling.

3. Computer graphics and visualization: Fractals are often used in computer graphics and visualization to create complex and intricate designs. Studying fractals can help in the development of algorithms and software tools for generating and manipulating fractal images.

4. Engineering and physics: Fractal geometry has practical applications in various fields, including engineering and physics. For example, fractal analysis can be used to study the behavior of materials and structures at different scales, and to model complex systems such as fluids and turbulence.

Overall, the study of fractals provides a fascinating and interdisciplinary approach to understanding the natural world and developing mathematical, computational, and scientific skills.

2 The Barnsley Fern

The Barnsley Fern is a fractal named after mathematician Michael Barnsley, who introduced it in his book "Fractals Everywhere" in 1988. It is one of the most famous examples of a fractal generated by an iterated function system (IFS).
The Barnsley Fern is based on four affine transformations that each take a point in two-dimensional space and transform it into a new point. These transformations are applied iteratively to a starting point, resulting in a complex and intricate fractal pattern.

Here are the programing code for Barnsley Fern that our group have done together:

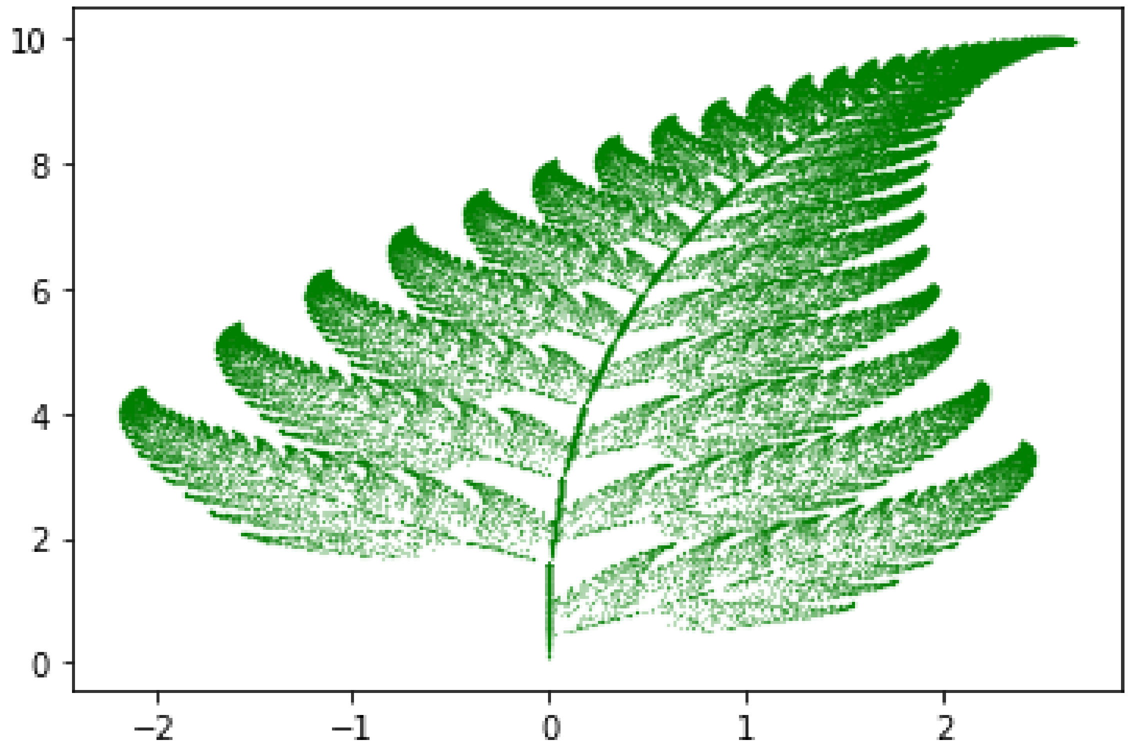
In [48]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [49]:
```
def function1(x,y):
    return(0.0,0.16*y)
def function2(x,y):
    return(0.85*x + 0.04*y, -0.04*x + 0.85*y +1.6)
def function3(x,y):
    return(0.2*x - 0.26*y , 0.23*x +0.22*y + 1.6)
def function4(x,y):
    return(-0.15*x + 0.28*y , 0.26*x + 0.24*y + 0.44)
functions = [function1,function2,function3,function4]
```

In [50]:
```
points = 100000
x, y = 0, 0
x_list = []
y_list = []
for i in range(points):
    function = np.random.choice(functions, p=[0.01, 0.85, 0.07, 0.07])
    x, y = function(x,y)
    x_list.append(x)
    y_list.append(y)
```

In [51]:
```
plt.scatter(x_list,y_list, s= 0.01, color = 'green')
```

And the outcome of this code gives us the Barnsley Fern



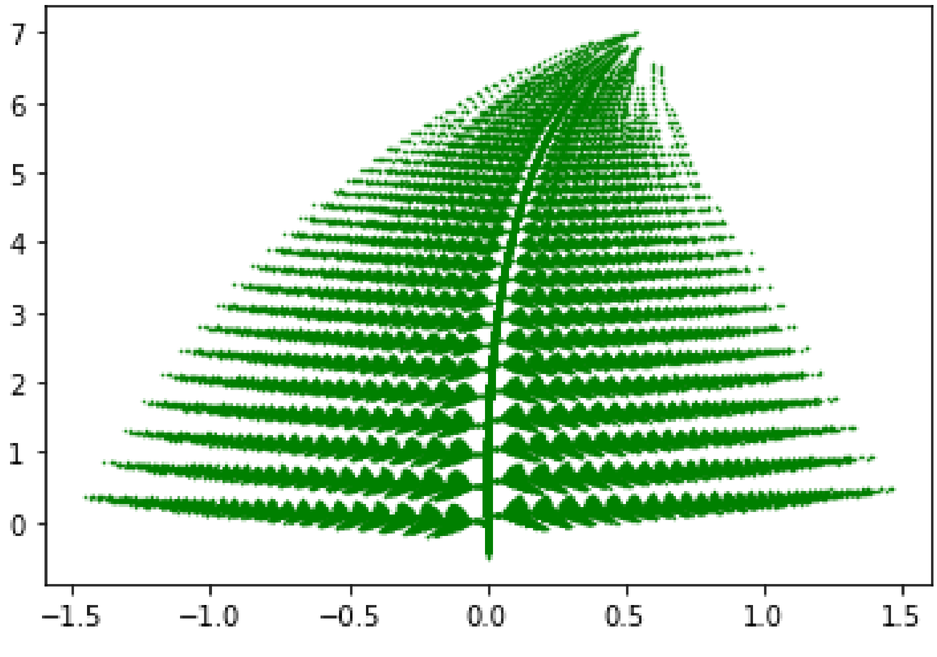
3 What happens if we change the constant values and probabilities?

We have found that different coefficients and different probability of applying each transformation can get different types of fern in our natural world; hence, by studying such mathematical function we can know more about our surrounding world. For example, if we change the constants and probability as

```
In [97]: def function1(x,y):
             return(0.0,0.25*y - 0.4)
         def function2(x,y):
             return(0.95*x + 0.005*y - 0.002, -0.005*x + 0.93*y +0.5)
         def function3(x,y):
             return(0.035*x - 0.2*y - 0.09, 0.16*x +0.04*y + 0.02)
         def function4(x,y):
             return(-0.04*x + 0.2*y + 0.083 , 0.16*x + 0.04*y + 0.12)
         functions = [function1,function2,function3,function4]
```

```
In [98]: points = 100000
         x, y = 0, 0
         x_list = []
         y_list = []
         for i in range(points):
             function = np.random.choice(functions, p=[0.02, 0.84, 0.07, 0.07])
             x, y = function(x,y)
             x_list.append(x)
             y_list.append(y)
         plt.scatter(x_list,y_list, s= 0.2, color = 'green')
```

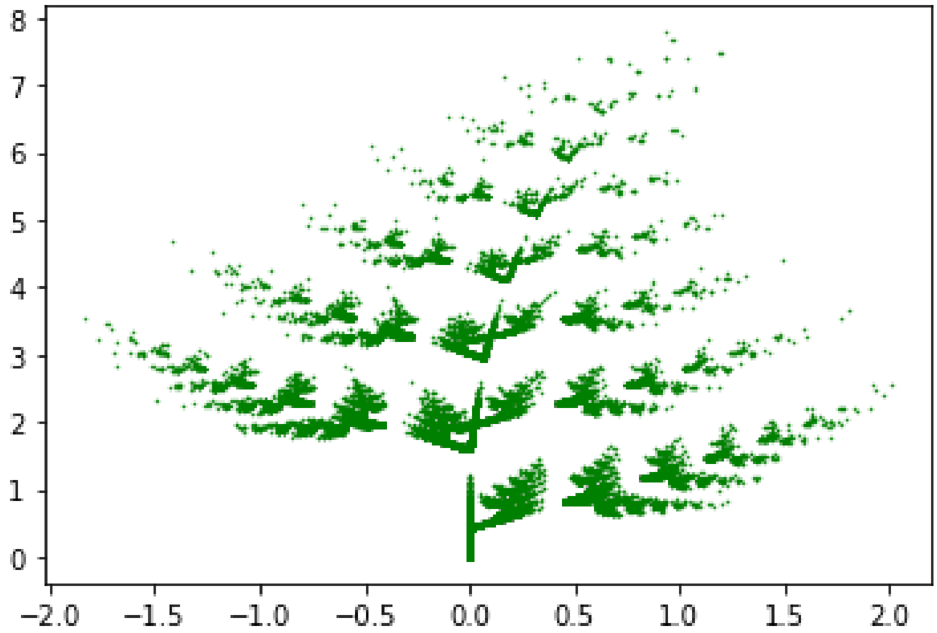
And the plot we get for this is



Now let's see what happens when we have used the same transformation but keeping the probabilities of choosing each function equal i.e; 25 percent

```
In [100]: points = 100000
x, y = 0, 0
x_list = []
y_list = []
for i in range(points):
    function = np.random.choice(functions, p=[0.25, 0.25, 0.25, 0.25])
    x, y = function(x,y)
    x_list.append(x)
    y_list.append(y)
plt.scatter(x_list,y_list, s=0.2, color = 'green')
```

And output of this would give a very broken image of the Barnsley Fern



4 Appendex

Barnsley's fern uses four affine transformation. The transformation formula is the following:

$$f(x,y) = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \tag{1}$$

$$f_1(x,y) = \begin{pmatrix} 0 & 0 \\ 0 & 0.16 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{2}$$

$$f_2(x,y) = \begin{pmatrix} 0.85 & 0.04 \\ -0.04 & 0.05 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 1.60 \end{pmatrix} \tag{3}$$

$$f_3(x,y) = \begin{pmatrix} 0.20 & -0.26 \\ 0.23 & 0.22 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0.00 \\ 1.60 \end{pmatrix} \tag{4}$$

$$f_4(x,y) = \begin{pmatrix} -0.15 & 0.20 \\ 0.26 & 0.24 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0.44 \end{pmatrix} \tag{5}$$

.

$$f_1 \rightarrow x_{n+1} = 0 \tag{6}$$
$$y_{n+1} = 0.16y_n \tag{7}$$
$$f_2 \rightarrow x_{n+1} = 0.85x_n \tag{8}$$
$$y_{n+1} = -0.04x_n + 0.85y_n + 1.60 \tag{9}$$
$$f_3 \rightarrow x_{n+1} = 0.20x_n - 0.26y_n \tag{10}$$
$$y_{n+1} = 0.23x_n + 0.22y_n + 1.6 \tag{11}$$
$$f_4 \rightarrow x_{n+1} = 0.15x_n + 0.20y_n \tag{12}$$
$$y_{n+1} = 0.26x_n + 0.24y_n + 0.44 \tag{13}$$

For the later part we would like to refer to our coding→