Lab 2

Ivan Annicchiarico

October 3, 2020

Abstract

In this lab report I presented the code that the Professor Hahn gave.

1 Introduction

Fractals are very complex patterns that present similar structures across different scales

2 Code part 1

```
!pip install git+https://github.com/williamedwardhahn/mpcr
from mpcr import *
def midpoint(P, Q):
    return (0.5*(P[0] + Q[0]), 0.5*(P[1] + Q[1]))
vertices = [(0, 0), (2, 2*np.sqrt(3)), (4, 0)]
n = 25 # Change this value and see what happens
When n is big enough, the result is a graph similar Sierpinski fractal
```

3 Code part 2

```
def pick(p):
    c = np.cumsum(p)
    return bisect(c, np.random.random() * c[-1])
p = np.array([0.01,0.07,0.07,0.85])
eq = [np.array([[0,0,0],[0,0.16,0]]),
      np.array([[0.2,-0.26,0],[0.23,0.22,1.6]]),
      np.array([[-0.15, 0.28, 0],[0.26,0.24,0.44]]),
      np.array([[0.85, 0.04, 0],[-0.04, 0.85, 1.6]])]
n = 100000 # Change this value and see what happens
x = np.zeros((n,3))
x[:,2] = 1
When n increases, a Barnsley ferns is generated
plt.figure(figsize=(10,10))
```

```
plt.scatter(x[:,0], x[:, 1], s=3, c="g", marker="s", linewidths=0)
plt.axis("equal"),plt.axis("off");
```