НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики Кафедра прикладної математики

Звіт

про виконання лабораторної роботи №3

по дисципліні «Основи Нелінійного Аналізу » на тему

«Фрактали. Побудова фракталів»

Виконала студентка групи КМ-81

Верзун П.В.

Перевірив доцент кафедри ПМА

Балюнов О.О.

Київ — 2021

Зміст

1. [Вступ 3](#_gjdgxs)
2. [Постановка задачі 3](#_30j0zll)
3. Теоретичні відомості 4
4. Розв’язання задачі 6
5. Висновок 10
6. Додаток А 11

# 1 Вступ

Мета виконання цієї лабораторної роботи полягає у побудові зображень фракталів на комплексній площині. В рамках виконання лабораторної роботи було проведено ознайомлення з теорією фракталів.

# 2 Постановка Задачі

Побудувати зображення множини Мандельброта на комплексній площині.

# 3 Теоретичні відомості

В математиці **множина Мандельброта** - це фрактал , визначений як множина точок {\ Displaystyle c \!} на комплексній площині , для яких ітеративна послідовність

{\ Displaystyle z_ {0} = 0 \!}

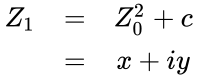
{\ Displaystyle z_ {n + 1} = {z_ {n}} ^ {2} + c \!}

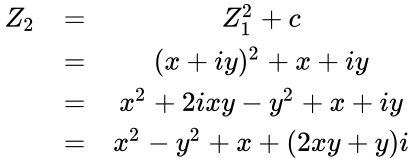
не йде в нескінченність.

Таким чином, вищевказана послідовність може бути розкрита для кожної точки {\ Displaystyle c \!} на комплексній площині таким чином:

{\ Displaystyle c = x + i \ cdot y \,}

{\ Displaystyle Z_ {0} = 0 \,}





{\ Displaystyle Z_ {3} = Z_ {2} ^ {2} + c = ... \,}

і тд.

Якщо переформулювати ці вирази у вигляді ітеративної послідовності значень координат комплексної площині {\ Displaystyle x \!}і {\ Displaystyle y \!}, Замінивши {\ Displaystyle z_ {n} \!}на {\ Displaystyle x_ {n} + i \ cdot y_ {n}}, а {\ Displaystyle c \!}на {\ Displaystyle p + i \ cdot q \!}, ми отримаємо:

{\ Displaystyle x_ {n + 1} = {x_ {n}} ^ {2} - {y_ {n}} ^ {2} + p \,}

{\ Displaystyle y_ {n + 1} = 2 {x_ {n}} {y_ {n}} + q \,}

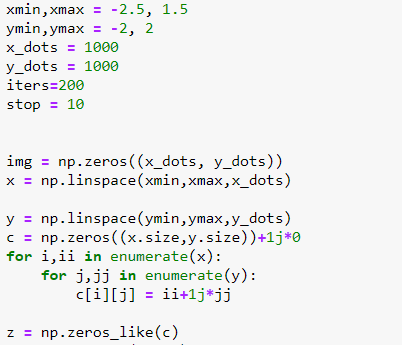
Візуально, всередині множини Мандельброта можна виділити нескінченну кількість елементарних фігур, причому найбільша в центрі являє собою кардиоиду . Також є набір кіл, що стосуються кардіоїди, розмір яких поступово зменшується, прямуючи до нуля. Кожен з цих кіл має свій набір менших кіл, діаметр яких також прямує до нуля і т. д. Цей процес триває нескінченно, утворюючи фрактал. Також важливо, що ці процеси розгалуження фігур не вичерпують повністю безліч Мандельброта: якщо розглянути зі збільшенням додаткові «гілки», то в них можна побачити свої кардіоїди і кола, не пов'язані з головною фігурою. Найбільша фігура (видима при розгляданні основного безлічі) з них знаходиться в області від -1,78 до -1,75 на негативній осі дійсних значень.

# 4 Розв’язання задачі

Для розв’язання задачі використовувалась мова програмування Python з такими модулями як matplotlib та numpy.

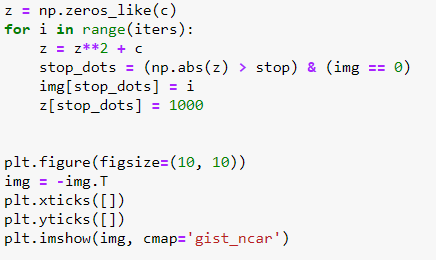


Спочатку ,використовуючи генератор linspace, визначаються межі в яких буде будуватися множина Мандельброта. Початкове зображення задається нульовою матрицею, матриця констант задається сіткою згідно з заданими межами. Матриця z - матриця розмірності матриці констант заповнена нулями.

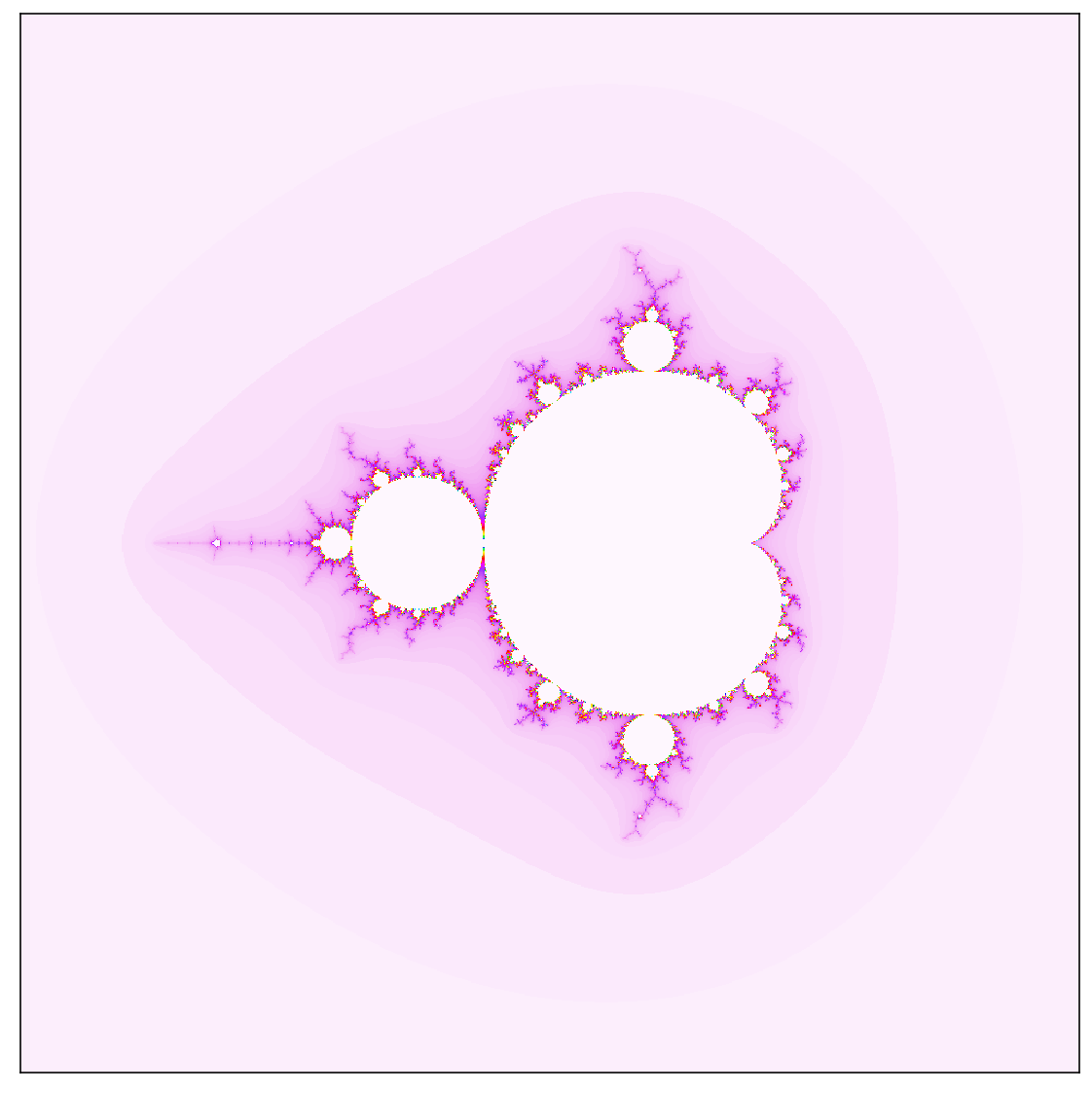


Далі згідно з заданою кількістю ітерацій відбувається зведення в квадрат матриці z та додавання матриці с. На кожній ітерації перевіряється чи не вийшла z за деяке заздалегідь встановлене значення. Якщо вийшла ,то елемент матриці ,що відповідає пікселю з відповідною координатою стає рівний значенню ітерації на якій відбулося переривання. Це дозволяє розмальовувати зображення в різні кольори в залежності від значення ітерації. Для відображення використовувалася палитра ‘gist\_ncar’, яка має вигляд:





Далі буде показана множина Мандельброта та деякі частини множини, які підтверджують ,що ця множина є фракталом, або іншими словами, в точності або приблизно збігається з частиною себе самого.



*рис.1*

Дані введені для рисунку 1:

**xmin,xmax = -2.5, 1.5**

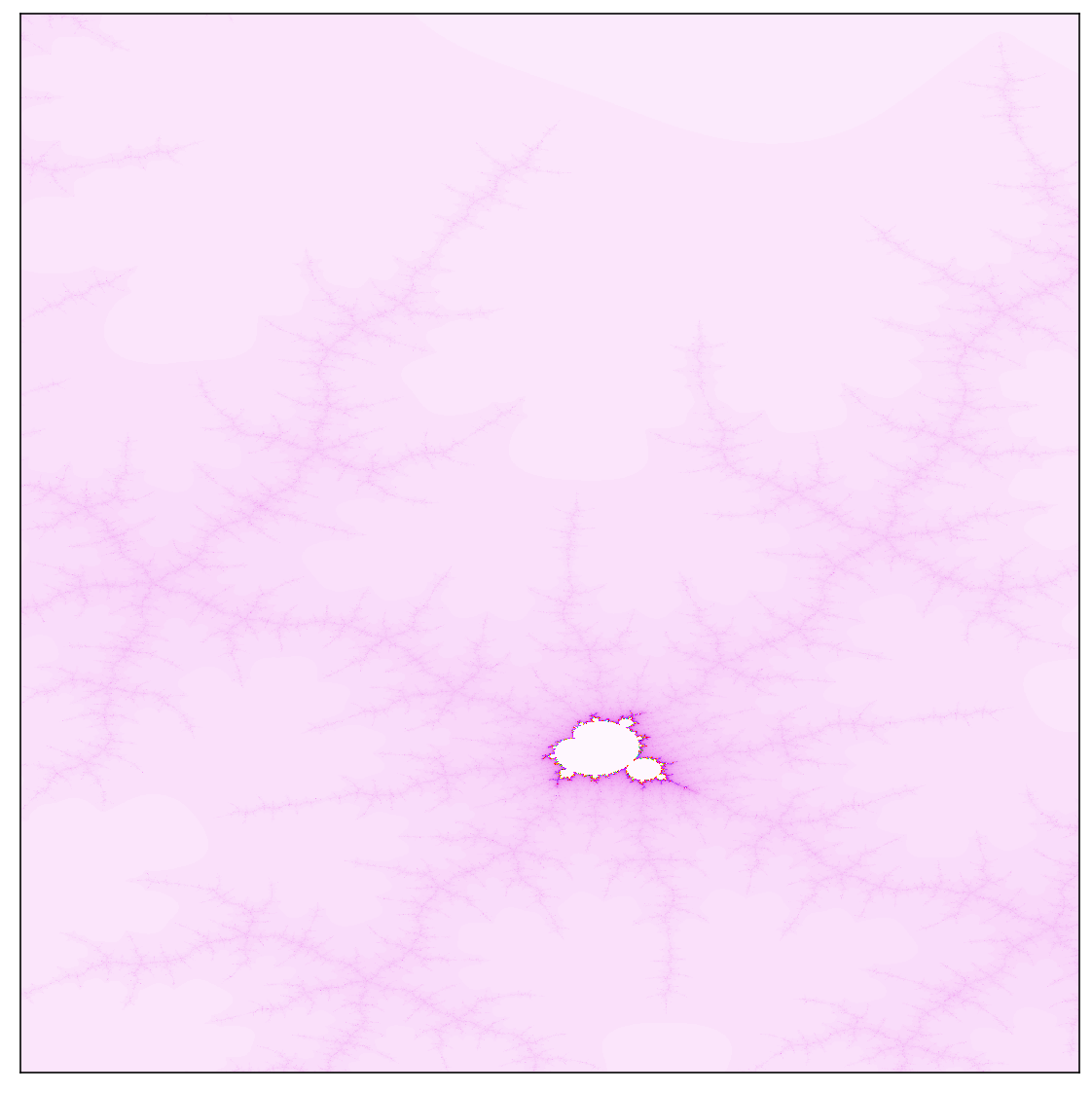
**ymin,ymax = -2, 2**

**x\_dots = 1000**

**y\_dots = 1000**

**iters=1000**

**stop = 100**



*рис.2*

Дані введені для рисунку 2:

**xmin,xmax = -1.3757446762278472, -1.3759046762278472**

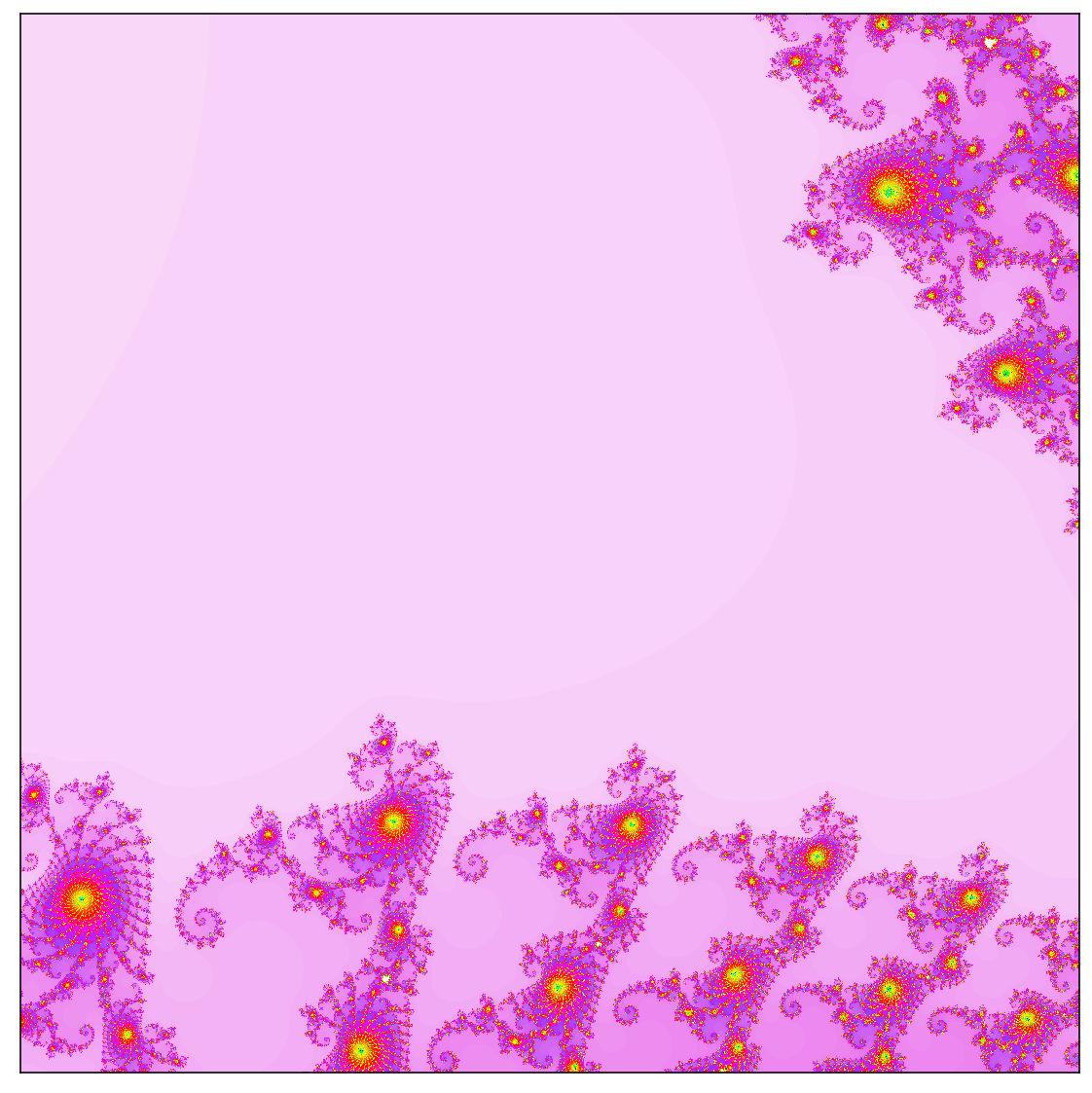
**ymin,ymax = 0.04302156600924987, 0.04328156600924987**

**x\_dots = 2000**

**y\_dots = 2000**

**iters=3000**

**stop = 4**



*рис.3*

Дані введені для рисунку 3:

**xmin,xmax = -0.748 , -0.746**

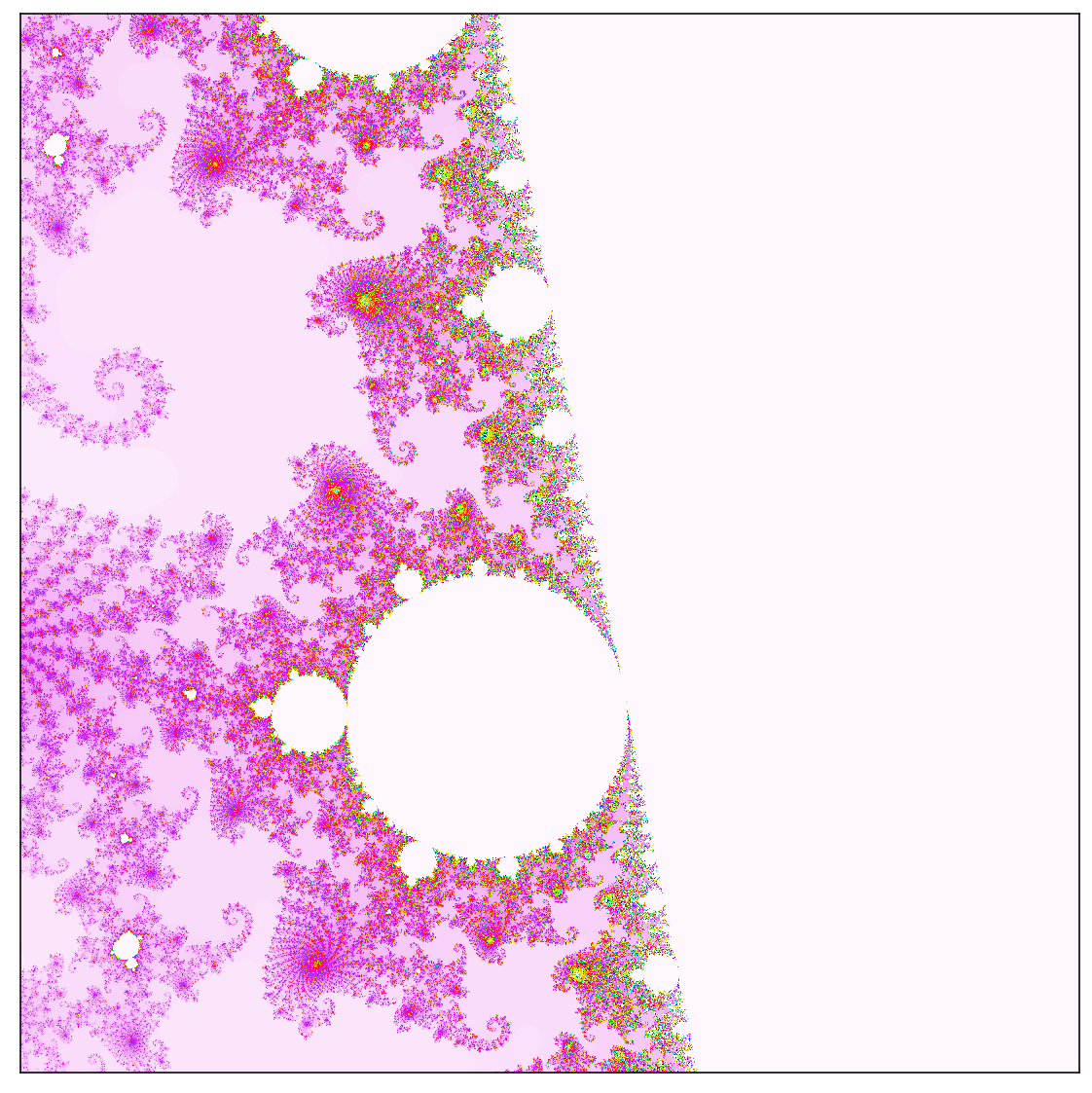
**ymin,ymax = 0.1119, 0.1139**

**x\_dots = 2000**

**y\_dots = 2000**

**iters=1000**

**stop = 4**



*рис.4*

Дані введені для рисунку 4:

**xmin,xmax = -0.74548 , -0.74548+0.01276**

**ymin,ymax = 0.11669, 0.11669 + 0.01276**

**x\_dots = 2000**

**y\_dots = 2000**

**iters=3000**

**stop = 4**

# Висновок

В результаті лабораторної роботи було досліджено множину Мандельброта з точки зору візуалізації, були отримані різноманітні форми та фігури на зображення, але всіх їх об’єднує одна спільна риса - вони самоподібні(адже являються частиною множини Мандельброта).

# Додаток А

**import numpy as np**

**import matplotlib.pyplot as plt**

**%config InlineBackend.figure\_format='retina'**

**xmin,xmax = -2.5, 1.5**

**ymin,ymax = -2, 2**

**x\_dots = 1000**

**y\_dots = 1000**

**iters=1000**

**stop = 100**

**img = np.zeros((x\_dots, y\_dots))**

**x = np.linspace(xmin,xmax,x\_dots)**

**y = np.linspace(ymin,ymax,y\_dots)**

**c = np.zeros((x.size,y.size))+1j\*0**

**for i,ii in enumerate(x):**

**for j,jj in enumerate(y):**

**c[i][j] = ii+1j\*jj**

**z = np.zeros\_like(c)**

**for i in range(iters):**

**z = z\*\*2 + c**

**stop\_dots = (np.abs(z) > stop) & (img == 0)**

**img[stop\_dots] = i**

**z[stop\_dots] = 1000**

**plt.figure(figsize=(10, 10))**

**img = -img.T**

**plt.xticks([])**

**plt.yticks([])**

**plt.imshow(img, cmap='gist\_ncar')**