Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»

Кафедра інформатики факультету інформатики

«Побудова фрактальних зображень та їх застосування у комп'ютерній графіці»

Текстова частина до курсової роботи за спеціальністю 122 «Комп’ютерні науки»

Керівник курсової роботи

Бучко О. А.

____

(підпис)

“___”_________2025 р.

Виконала студентка 3 р. н.

Косів Х. А.

“___”_________2025 р.

Київ 2025

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»

Кафедра інформатики факультету інформатики

ЗАТВЕРДЖУЮ

Бучко О. А.

(підпис)

„_______________________________________”____________2025 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу

студентці Косів Христини Андріївні факультету інформатики 3-го курсу

ТЕМА: Побудова фрактальних зображень та їх застосування у комп'ютерній графіці

Вихідні дані:

Зміст ТЧ до курсової роботи

Вступ

Анотація

1. Постановка задачі
2. Дослідження алгоритмів побудови фракталів
3. Практичний аналіз
4. Висновки
5. Список використаних джерел

Дата видачі ”___”__________2025 р. Керівник ________________

Завдання отримала ____

Календарний план виконання курсової роботи

Тема: Побудова фрактальних зображень та їх застосування у комп'ютерній графіці

Календарний план виконання роботи:

№ п/п	Назва етапу дипломного проекту (роботи)	Термін виконання етапу	Примітка
1.	Отримання завдання на курсову роботу.	Жовтень 2025 р.	
2.	Огляд технічної літератури за темою роботи.	Грудень 2025 р.	
3.	Створення практичної частини роботи	Січень-квітень 2025 р.	
4.	Написання текстової частини роботи	Березень-квітень 2025 р.	
5.	Надання роботи керівнику для перевірки	Кінець квітня 2025 р.	
6.	Коригування роботи за результатами перевірки кінець квітня-початок травня	Початок травня 2025 р.	
7.	Здача роботи для перевірки на плагіат.	Початок травня 2025 р.	
8.	Захист курсової роботи.	Середина травня 2025 р.	

Студент: Косів Х. А.

Керівник: Бучко О. А.

“______” ________________

ЗМІСТ

Анотація

Курсова робота присвячена дослідженню побудови фракталів та застосування їх у комп'ютерній графіці, проте також у мистецтві, літературі, медицині. Було проведено аналіз алгоритмів, їх детальний опис, та порівняння продуктивності різних підходів побудови.

У практичній частині використано бібліотеки matplotlib, pandas, numpy у мові програмування Пайтон, що дозволило наочно побачити побудову. Також розглянуто аналіз комп'ютерних ігор.

Вступ

Фрактали – всюди, у музиці, у мистецтві, у літературі, в іграх, у зображеннях, у медицині. У комп'ютерній графіці представлені найбільш характерно. Продемонстровано різноманітні приклади, де можна наочно побачити методи, структуру та власне самі зображення.

Спочатку описано побудову та алгоритми п'яти різних фракталів, а саме від найпростіших до найскладніших, від найлегших до найцікавіших. Таким чином можна більше дізнатися визначення фракталів, та як вони будуються.

Далі у роботі їх порівняння, з використанням формул. Виділено три характеристики, а саме швидкість побудови, де визначено практично, завдяки коду побудові, візуальна складність, де обгрунтована власна оцінка на основі характеристики та практичності, та найголовніше це фрактальна розмірність, де використано формулу для самоподібних фракталів. Та звісно ж загальний висновок з таблицею, для кращого ознайомлення.

Наступне, особисто для мене – найцікавіша частина, це виявлення ознак фрактальності, самоподібності, розмірності у реальному світі, а саме в медицині, мистецтві та літературі. Завдяки книзі Максима Кідрука обрана ця тема, бо прочитаний мною його технічний трилер "Бот" справив на мене неабияке враження. Таким чином, досліджено як у медицині використано фрактали та яке майбутнє у нього. Виявлено вірші та твори авторів, які використали самоподібність та навіть код та зображення. У скульптурах та музиці теж знайдено фрактальні алгоритми.

У курсовій роботі описано структуру різноманітних фракталів, зосереджено більше на теорії, також практична частина створена в Jupyter-notebook, де зображено візерунки та код до них за допомогою мови програмування – python. Та звісно ж проведений практичний аналіз комп’ютерних ігор, де виявлено описані мною фрактали.

1. Аналіз предметної області

Фрактали – це складні зображення, які створюються за допомогою математичних формул та ітерацій. Деякі явища природи, як-от гори, хмари не можна зобразити рівняннями чи формулами, бо їхня форма заскладна, тоді й використовують фрактали.

Мають широке застосування:

- Астрономія – аналіз кілець Сатурна, галактик.
- Біологія – зображення анатомії людини, рослин, бактерій.
- Хімія – ілюстрації хімічних реакцій.
- Географія – дослідження кордонів, берегових ліній.

Створення зображень: генерація точок безліч разів призводить до утворення фракталів. Для отримання потрібної форми та розміру, ми можемо ітерувати нескінченно, ще є назва для цього процесу – рекурсія, яка використовується в програмуванні.

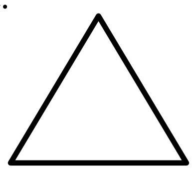
Типи фракталів:

1) Self-similar (самоподібні) – частини є зменшеними копіями цілого об’єкту. Масштабуємо одну частину, тоді отримуємо подібну до цілого структуру.
2) Self-affine (самоафінні) – масштабування відбувається з різними коєфіцієнтами у різних напрямках, а саме координатних.
3) Invariant (інваріантні) – створення відбувається за допомогою нелінійних перетворень

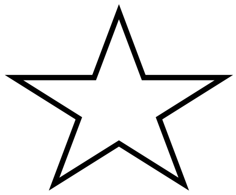
Розмірність фракталів - це поняття, яке характеризується складністю об’єкта чи зображення. Тобто чим вища розмірність, тим щільніша є картинка.

Чому це важливо? Він кількісно вимірює самоподібність та нерегулярність візерунків, не лише використовує довжину, площу, об’єм. Завдяки врахуванням розмірності можна досягти покращень у стисненні зображень, генерації комп’ютерної графіки, обробки мереж, моделюванні, аналізі даних, що зробить ці процеси набагато ефективнішими та реалістичнішими.

Геометричні фрактали: вони описують форми, що тряпляються в природі. Для побудови починаємо з певної початкової фігури – ініціатора. Далі підчастини ініціатора замінюються за допомогою візерунка, а саме генератора.



- Ініціатор



- Генератор

Класифікація фракталів:

- Exact self-similarity (найпотужніший тип, сильна особливість у масштабуванні, тобто однаково виглядатиме зображення при різних екранах. Демонструють самоподібність завдяки системі ітераційних функцій).
- Quasi-self-similarity (фрактал виглядатиме приблизно, проте не повністю однаково при різних масштабів. Містять малі копії всього фракталу в деформованій формі).
- Statistical self-similarity (найслабший тип у самоподібності, зберігає числові характеристики або статистичні при різних масштабів).

Як з’явились фрактали? Термін “фрактал” ввели у 1975 році. Розробка першого програмного забезпечення для створення фракталів розпочалась з ідей Бенуа Мандельброта, та саме він та ще декілька програмістів створили перші роздруківки. Вивчення самоподібних об’єктів розпочалось з Лейбніца в 17 столітті і продовжилось наприкінці 19 століття (та на початку 20 століття).

2. Практична реалізація та аналіз фрактальних зображень

2.1. Множина Канта

Множина Канта є однією з найпростіших та найлегших фракталів для початку, щоб розібратися з усіма деталями їх побудови та алгоритму.

Щодо історії, то Георг Кантор – німецький математик, найбільш відомий своєю роботою з теорії множин, також він довів, що множина дійсних чисел є незліченною. Проте множина Канта було винайдена Генрі Смітом в 1874 році, а Георг був саме тим, хто зробив це популярним, демонструючи усьому світу в 1883 році. Множина є цікавою та простою тому, що складається з двох множин чисел, а саме із зліченною та незліченною.

Створення множини Канта: сам алгоритм є дуже простим, тобто спочатку беремо відрізок з довжиною 1 одиниця, далі ділимо відрізок на три рівні частини і видаляємо середину, таким чином рекурсивно повторюємо отримане зображення. Нижче наведено приклад:

Отже, це можна робити безкінечно. Якщо продовжувати нескінченну кількість кроків, то залишаться точки, яких буде безліч.

Властивості множини Канта:

1)Перша властивість полягає в тому, що загальна довжина всіх інтервалів у множині буде дорівнювати нулю. Тобто це означає, що точки не створюють єдиний інтервал, а є окремими точками. Доведення: підсумуємо довжину проміжків. Таким чином, з попереднього зображення випливає, що на кроці 2 ми отримали проміжок довжиною 1/3, на кроці 3 ми отримали 2 проміжка довжиною 1/9 і так далі. Отримали формулу – 1/3+2/9+4/27+…

За допомогою геометричної прогресії маємо, що A+AR+AR^2+AR^3+…=A/(R-1). A=1/3, R=2/3, (R-1)=1/3.

Сума вийшла 1/3 % 1/3 = 1 – це загальна довжина проміжків. Звідси випливає, що довжина інтервалів, що залишається після видалення всіх проміжків дорівнює нулю.

2)Друга властивість полягає в тому, що множина не має ізольованих точок. Тобто для будь-якої точки, яка є в множині, можна знайти іншу точку, яка буде близько до першої.

3)Третя властивість полягає в тому, що кількість елементів у множині Кантора дорівнює кількості дійсних чисел, тобто є незліченною.

2.2 Крива Дракона (Dragon Curve)

Крива Дракона – це крива, яка є рекурсивною та яка не перетинається між собою. Тісно пов’язана з математикою, та навіть можна апроксимувати за допомогою L-системи.

L-система – це система переписування рядків, яка є дуже корисною в побудові фракталів. Починається усе з рядка сивола, яка називається аксіомою і до нього застосовується набір правил, які використовуються для переписання. Рекурсивні L-система - це кожен символ змінюється його копією і плюс шось додаткове відбувається.

Правила:

1) Символи, які замінюються з правил набір правил трансформації – набір змінних
2) Символи, які не замінюються (+, -, !) – набір констант
3) Початковий рядок – аксіома
4) Правила, за допомогою, яких визначають як саме змінні можуть бути переписані – набір правил трансформації.

Алгоритм фрактала:

Будується як границя багатокутних наближень D(n).

- D(1) – є звичайним відрізком.
- D(n+1) отримуємо з D ось так:

1) Перемістимо D(n) так, щоб його кінцева точка опинилась в початку координат.
2) Копіюємо та множимо на √(1/2) .
3) Результат повертаємо на 45° і називаємо C(n) наш результат.
4) Копіюємо C(n) та повертаємо на 90°, приєднуємо копію до кінця C(n).

Тобто границі послідовностей багатокутників можуть формувати двовимірну множину, таким чином Драконова крива є неперервною кривою з образом, що має додатну площу.

Приклад кривої Дракона через L-систему:

Крива Драковна L-системи	
змінні:	f, h
константи:	+ –
аксіома:	f
правила:	f->f-h h -> f+h
кут:	90 °
крок 1:	f-h
крок 2:	**f-h** – **f+h**
крок 3:	**f-h** – **f+h** – **f-h** + **f+h**
крок 4:	**f-h** – **f+h** – **f-h** + **f+h** – **f-h** – **f+h** + **f-h** + **f+h**

Це є послідовність для побудови фрактала. Символи додавання та віднімання керують поворотами на 90°, а f та h – є командами руху вперед з малюванням відрізка.

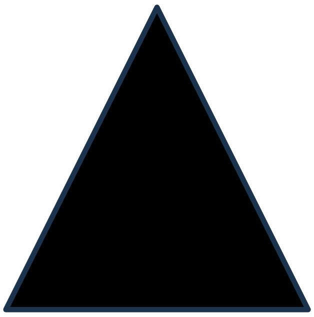
Отже, завдяки L-системам можна створити складні геометричні фігури, які забезпечують зрозумілий підхід до побудови, який не лише адаптивно використовується, а й дуже цікавий та наочно демонстративний. Математична суть криві Дракона дозволяє побачити ці правила системи. Вона не лише не перетинається, а й заповнює площу, що робить її ще більш популярною з графічної точки зору. Завдяки повторення звичайних правил саме зображення виглядає дуже красиво.

2.3 Сніжинка Коха (Koch snowflake)

Фрактал є варіантом кривої Коха

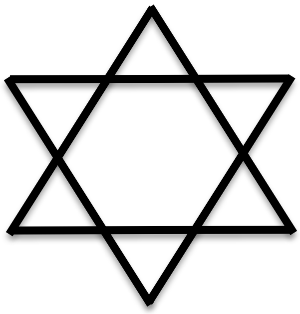
Алгоритм побудови:

1) Почнемо з рівностороннього трикутника:



2) Наступне ділимо кожну сторону на 4 відрізки і змінюємо як показано:

Центральний виступ формуватиме основне зображення, утворюючи кут 60 градусів. Застосуймо до всіх сторін трикутника:



Форма називається гексаграмою, тобто зірка з 6 вершинами, де всі сторони рівні.

Отже, узагальнемо три етапа побудови та знайдемо площу даного фрактала:

- Ітерація 0 – рівносторонній трикутник, 4 сторони. N(0)=3
- Ітерація 1 – малюємо трикутник на кожній стороні, таким чином кожна

сторона стає 4 сторонами, тоді взагальному буде 12 сторін. N(1)=12

- Ітерація 2 – знову малюємо трикутник на кожній стороні, тоді буде 48 сторін. N(2)=48

Тоді, кількість сторін починається з 3 і множиться на 4 з кожною ітерацією. N(n)=3*(4^n).

Обчислимо загальну площу фрактала: кожна ітерація додає додаткові трикутники, кількість додаткових трикутників на ітерація n, дорівнює попередній кількості. T(n)= 3*(4^n)-1. Для кожної ітерації довжина трикутника також зменшується, а саме в 3 рази, тоді площа зменшується в 9 разів.

Можемо сказати, що площа початкового трикутника дорівнює a0, тоді площа кожного трикутника буде:

a(1)=a(0)9, a(2)=a(0)/9^2, a(n)=a(0)/9^n

Тоді додаткова площа Sn – це кількість трикутників, помножена на площу кожного трикутника для цієї ітерації.

Sn = T(n) * a(n) = ¾ *4^n * a(0)/9^n

Загальна площа сніжинки дорівнює площі початкового трикутника та сума всіх додаткових площ до цієї ітерації.

A(n)=a(0)+Σ(k=1)*S(n)

Периметр: на кожній ітерації кожна сторонра ділить на 4 сторони, таким чином на кожній ітерації периметр збільшується в ¾ рази. Периметр буде нескінченний, якщо кількість сторін збільшуватиметься.

2.4. Фрактальний вогонь (Flame Fractals)

Цей фрактал особливий тим, що відноситься до орбітальних, де формула є набором перетворень, одне з яких обирається випадково на кожній ітерації. Інша назва – гра хаосу. Можна згенерувати за допомогою ітераційної системи функцій. Ітераційна система функцій – набір рівнянь, який використовується для визначення точок для перетворення на площині. Фрактальний вогонь відкрив Скотт Дрейвс у 1992 році. Завдяки ітерацій простого набору функцій, створюються надзвичайно унікальні зображення. Цей фрактал є відносно новим, проте уже є популярним, бо має широкий спектр можливостей та простоті генерації. Можна припустити, що їх популярність лише зростатиме у майбутньому.

Флеймові фрактали мають три компонента:

1.Affine transform (афінне перетворення) – зберігає паралельність, створює прямі, виконує операції такі як, обертання, масштабування, нахил, зсув, дзеркальне відображення. Наприклад, квадрат може перетворитись на паралелограм.

2.Non-linear transform (нелінійне перетворення) – масштабування координат, тобто означає дія через одну або кілька варіацій.

3.Post-affine transform (післяафінне перетворення) – застосовується після нелінійного перетворення таке саме, як афінне.

Алгоритм побудови:

1)Перший крок – визначити набір функцій, кожна з яких матиме три компоненти описані вище, тому використання афінних перетворень та складання нелінійних перетворень.

Афінне перетворення:

x1=a(i)*x+b(i)*y+c(i)

y1= d(i)*x+e(i)*y+f(i)

Кожна така функція є варіацією:

- V(x,y) = (x,y) – лінійна

- V(x,y) = (sin x, sin y) – синусоїдальна

- $V(x,y) = (x,y) / r^2$ – сферична

- $V(x,y) = 1/r * ((x-y)(x+y), 2xy)$ – підкова

2)Другий крок – визначення точку початку, а саме з $x2=0$ $y2=0$.

3)Третій крок – післяафінне перетворення, а саме кути

- $x3=a(i)*x2+b(i)*y+y(i)$

- $y3= d(i)*x2+e(i)*y+f(i)$

4)Четвертий крок – ітерації, повторення багато разів.

Отже, усі точки перетворюються в координати зображення – пікселі.

Особливості:

- Можна використати логарифмічне освітлення
- За індексом функції поєднати зображення з кольором
- Використання нелінійних варіацій

Тому, фрактальний вогонь – це потужний розвиток ітераційних фракталів, яке також поєднує нелінійні варіацій, афінні та постафінні перетворення, кольрову генерацію для створення художніх візуалізацій.

2.5. Листок Барнслі (Barnsley Fern)

Листок є найпопулярнішим зображенням фрактала. Завдяки математичним формулам створюється цікаве зображення папороті. Якщо заглибитись в історію, то моделювання рослин та інших природних явищ було спеціалізацією Майклом Барнслі. Він надихнув інших, хто намагався імітувати природу, завдяки математиці.

Особливості зображення:

- Стохастичний підхід: при кожній ітерації обирається одне з перетворень, тому можна створити фігуру з різних точок.

- Подібність: кожен листочок схожий на загальну картину.

- Детальність: чим більше ітерацій, тим чіткіше зображення.

Тому для створення фрактала використано 4 математичних рівнянь, які Барнслі представив у вигляді системи ітераційних функцій, що генерують точки для побудови на графіку, у наслідку багаторазових обчислень.

Результат одного обчислення використовується як вхід для наступного. Застосовуємо спеціальні формули, генеруємо нові точки та підставляємо їх знову. У висновку з'являється чудовий малюнок, який і справді дуже нагадує папороть. Чим довше триває процес, тим чіткіший та детальніший створюється фрактал.

Формули перетворення, які використовує автор, з коефіцієнтами:

a	b	c	d	e	f	p	-
0	0	0	0.16	0	0	0.01	Стебло
0.85	0.04	-0.04	0.85	0	1.60	0.85	Малий листок
0.20	-0.26	0.23	0.22	0	1.60	0.07	Великий листок(ліворуч)
0.15	0.28	0.26	0.24	0	0.44	0.07	Великий листок(праворуч)

Що означають коефіцієнти? Вони для чотирьох різних перетворень. Тобто, кожна формула описує, як з координат x, y отримати нові координати x`, y`. Тоді a, b, c, d – стосуються перетворення, а саме матриця обертання, нахил, маштабування; e, f – зсув; p – ймовірність вибору перетворення.

- Перша формула: x(n+1)=0, y(n+1)=0.16*y(n) – відбувається стискання по вертикалі, а саме формування стебла.
- Друга формула: x(n+1)=0.85*x(n)+0.04* y(n), y(n+1)=-0.04* x(n)+0.85*y(n)+1.6 - найголовніше перетворення, відбувається обертання, стискання та зсув точок вгору.
- Третя формула: x(n+1)=0.2*x(n)+0.26* y(n), y(n+1)=0.23* x(n)+0.22*y(n)+1.6 – відбувається створення лівої частини папороті.
- Четверта формула: x(n+1)=-0.15*x(n)+0.28* y(n), y(n+1)=0.26* x(n)+0.24*y(n)+0.44 – відбувається створення правої частини папороті.

Отже, алгоритм побудови:

1.Починаємо з точки, у якій x=0, y=0.

2.Обираємо одне з чотирьох перетворень відповідно до ймовірностей.

3.Після застосування формули отримуємо нову точку.

4.Малюємо цю точку та повторюємо безліч разів.

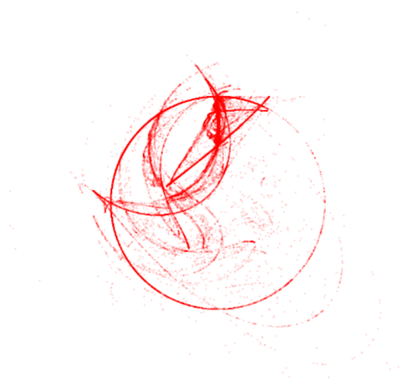
Знаючи перетворення для папороті та алгоритм побудови можна розробити код для їх візуалізації.

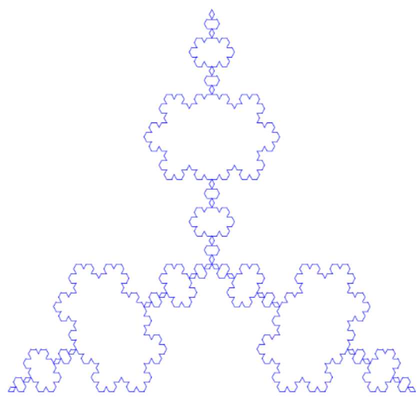
Складність створення листка Барнслі у тому, що необхідно велика кількість ітерацій, тому її складно побудувати самостійно, не використовуючи мову програмування.

Отже, фрактал демонструє, що завдяки формулам можна отримати надзвичайно красивий, хоча й простий малюнок, який демонструє як за допомогою математики отримати імітацію природи. При побудові використовуються ітераційні формули, рекурсію, ітерації, випадкових процесів, що є чудовим інструментом у науці.

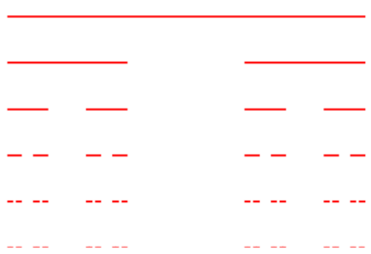
3. Порівняння фракталів

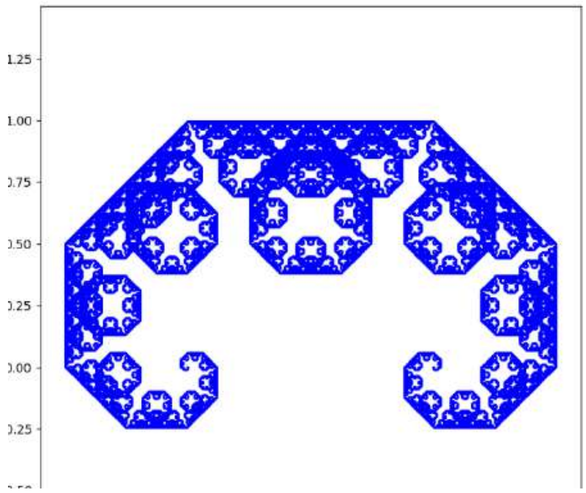
Використовуючи практичний код у побудові фракталів можна зробити деякі висновки у порівнянні. Тому візуальні зображення:





Фрактальний вогонь Сніжинка Коха





Множина Кантора   Крива Дракона

Листок Барнслі

3.1. За швидкістю та ефективністю

- Фрактальний вогонь. За допомогою випадкового вибору варіацій точки оброблятимуться окремо та постійно нові обчислення синуса та косинуса. Швидкість буде середня через це, бо доволі обширні обчислення. Щодо мінусів, то через scatter кожні 500 точок на великій кількості уповільнується процес.

- Сніжинка Коха. Завдяки принципу побудови, а саме, що кожен відрізок ділиться на чотири нові відрізка, то кількість ліній у чотири рази збільшується. Щодо швидкості, то вона повільна, адже, чим більше ітерацій ти складніший процес, тому швидкість є нестабільна.

- Множина Кантора. Найпростіший фрактал, тому очевидно, що швидкість висока, бо всього лише лінії та координати.
- Крива Дракона. При кожній ітерації відбувається подвоєння точок, тому швидкість є середньою.
- Листок Барнслі. Швидкі точки та обчислення, тому швидкість висока.

3.2 За візуальною складністю

Назва	Оцінка	Опис
Фрактальний вогонь	4.5/5	Химерні малюнки
Сніжинка Коха	4/5	Симетрична фігура з зрозумілою структурою
Множина Кантора	1/5	Дуже просте зображення
Крива Дракона	4.5/5	Цікава форма
Листок Барнслі	5/5	Складний, проте чіткий візерунок

Пояснення:

1) Фрактальний вогонь (4.5/5) – Складна побудова, нечітка структура варіацій. Проте з недоліків, то важкий аналіз.
2) Сніжинка Коха (4/5) - Гарна симетрія і нескінченна деталізація, але фігура досить "механічна", без природної нерівності.
3) Множина Кантора (1/5) – Проста, але й мінус, що виглядає нецікаво, немає живої структури.
4) Крива Дракона (4.5/5) – Чудова самоподібність, цікавий, різноманітний візерунок.
5) Листок Барнслі (5/5) - Найближча до справжніх природних форм.

3.3 За фрактальною розмірністю

Назва	Приблизна розмірність	Опис
Фрактальний вогонь	1.7–2.0	Залежить від варіацій
Сніжинка Коха	~1.26	Класичний приклад фракталу
Множина Кантора	~0.63	Дуже простий фрактал
Крива Дракона	~2.0	Майже повністю заповнює площину
Папороть Барнслі	~1.7	Наближено до природних форм листя

Формула для самоподібних фракталів:

$D=\log(N)/(\log(1/r))$

Для сніжинки Коха: $D=\log(4)/\log(3)\approx 1.26186$

- $N=4$— кількість фрагментів

- $r=1/3$— кожен відрізок втричі коротший.

Для множини Канта: $D=\log(2)/\log(3)\approx 0.63093$

- $N=2$— кількість відрізків

- $r=1/3$— кожен відрізок втричі коротший.

Для фрактального вогню: від 1.7 до 2, тому що вони зазвичай заповнюють площину повністю.

Для кривої Дракона: D = 2, бо заповнює площину.

Для листка Барнслі: розмірність буде 1.7, на жаль, теж немає конкретної формули.

Висновок досліджень:

Назва	Швидкість	Складність	Фрактальна розмірність
Фрактальний вогонь	середня	4.5/5	1.7–2.0
Сніжинка Коха	повільна при багатьох ітерацій	4/5	~1.26
Множина Кантора	дуже швидка	1/5	~0.63
Крива Дракона	середня	4.5/5	~2.0
Листок Барнслі	дуже швидка	5/5	~1.7

Отже, при порівнянні можна прийти до висновку, що Листок Барнслі, у звя’зку з подібністю до природи має найбільшу складність з обраних мною фракталів, велика фрактальна розмірність та чудова швидкість тому, що хоч й має велику кількість ітерацій, проте чіткий алгоритм побудови. Множина Кантора – чудовий фрактал для початківців, бо швидкість висока, проте складність найлегша. Сніжинка Коха велика складність, середня розмірність, проте повільна при великій кількості ітерацій. Щодо кривої Дракона, то висока складність, у зв’язку з різноманітністю варіацій та велика фрактальна розмірність, проте середня швидкість у порівнянні з іншими. Фрактальний вогонь матиме найбільшу фрактальну розмірність, високу складність, але середню швидкість.

4. Фрактальні зображення у моделюванні реального світу

4.1. Фрактали в медицині

У медицині фрактали є дуже поширеними тому, що допомагають в біологічних процесах. Вони всюди – від складних нейронних мереж у мозку до дихальних шляхів у легенях.

Чим допомагають фрактали?

- Проаналізовані через фрактальну геометрію варіації серцевого ритму можуть сигналізувати про ранні ознаки проблем з серцем.
- Фрактальний аналіз допомагає визначити зміни на знімках, коли ракові тканини ростуть в організмі у вигляді нерегулярних візерунках.
- Фрактальні симуляції допомагають передбачити, як ліки взаємодіятимуть з складними тканинами, роблячи процес створення ліків чіткішим та швидшим.
- Фрактальні дизайни вдосконалюють каркаси для загоєння ран, також регенерації тканин.

Потрібно згадати про морфометрію, що є важливою та невід’ємною частиною досліджень, а саме морфологічних. Існує багато анатомічних структур з складними формами і буває важко охарактеризувати їх традиційними способами, тому на допомогу приходять саме фрактальний аналіз. В останні десятиліття принципи, методи фрактальної геометрії використовується частіше.

Розмірність є корисним параметром також в медицині, бо дозволяє оцінити ступінь складності нерегулярних анатомічних структур та доповнює показники патологій.

Таким чином, вони покращують розуміння біології, створюючи інструменти для змін медицини. Наприклад, органи та протези, що імітують природу вже є реальністю з назвою – регенеративна медицина.

Звісно ж можливості обмежені тому, що є потреба в потужних обчислювальних ресурсів, складній техніці. Проте алгоритми здатні передбачити навіть захворювання, завдяки прогнозам на основі штучного інтелекту. Отже, фрактали – початок для інновацій.

4.2. Фрактали в літературі

Насправді знайти фрактальні структури в літературі нелегко, особливо чітко визначену. Проте перший натяк на це можна знайти у висловлюваннях Поле Остера в інтерв'ю про його ж роман "Mr. Vertigo", де перше речення містить суть усього твору. Таке робив і Шенкер у музиці.

Очевидно, що автори несвідомо торкались фрактальних тем, однак все ж таки знайти їх реально. Оскільки вони оточують нас всюди, а саме у природі, а література буває зображує природу у своїх рядках, тому не дивно, що у вони існують й там.

Почати хочу з оповідання Хорхе Луїса Борхеса "The Aleph", у якому описується чудове місце, з якого можна побачити усі місця з усіх ракурсів. Ось приклад:

"…On the back part of the step, toward the right, I saw a small iridescent sphere of almost unbearable brilliance. At first I thought it was revolving; then I realized that this movement was an illusion created by the dizzying world it bounded. The Aleph's diameter was probably little more than an inch, but all space was there, actual and undiminished. Each thing was infinite things, since I distinctly saw it from every angle of the universe. I saw the teeming sea; I saw daybreak and nightfall; I saw the multitudes of America; I saw a silvery cobweb in the center of a black pyramid;" and on and on, until "I saw the Aleph from every point and angle, and in the Aleph I saw the Earth and in the earth the Aleph and in the Aleph the earth…"

Алеф нагадує монади Лейбніца, тобто кожна частина Всесвіту містить у собі весь Всесвіт, тому нескінченну кількість копій Всесвіту, кожна з яких знову містить нескінченну кількість копій, і так продовжується безкінечно. Лейбніц запропонував фрактальну модель Всесвіту, а Борхес художньо втілив цю ідею.

Наступний приклад це неймовірна паліндромна поема, де кожна літера зчитується в обидві сторони, тобто перший та останній рядок є ідентичним, структура – дзеркальна вісь симетрії навколо центрального рядка. Центр фрактальної самоподібності є рядок (“Be still if I fill its ebb”) – що є майже паліндромом. До речі, цей вірш був проєктом студента на курсі фрактальної геометрії, ось і він:

“…Dammit I'm mad
Evil is a deed as I live.
God, am I reviled?
I rise, my bed on a sun, I melt.
To be not one man emanating is sad. I piss.
Alas it is so late. Who stops to help?
Man, it is hot.

I'm in it.
I tell.
I am not a devil.
I level "Mad Dog".

Ah, say burning is as a deified gulp
in my halo of a mired rum tin.
I erase many men. Oh, to be man, a sin.
Is evil in a clam? In a trap?
No. It is open.
On it I was stuck.

Rats peed on hope.
Elsewhere dips a web.
Be still if I fill its ebb.
Ew, a spider ... eh?
We sleep.

Oh no!
Deep, stark cuts saw it in one position.
Part animal, can I live? Sin is a name.
Both, one ... my names are in it. Murder?
I'm a fool. A hymn I plug,
Deified as a sign in ruby ash - a Goddam level I lived at.

On mail let it in. I'm it.
Oh, sit in ample hot spots.
Oh, wet!
A loss it is alas (sip). I'd assign it a name.
Name not one bottle minus an ode by me:
"Sir, I deliver. I'm a dog."
Evil is a deed as I live.
Dammit I'm mad…”

Не могла не згадати Максима Кідрука, який є моїм земляком з Рівненської області. Його твір “Бот”, що є першим технічним трилером в Україні. Саме фракталами привернули увагу злодія та його помічників – ботів. Тимур – головний герой спочатку показував елементарні зображення, проте психістоті потрібно було більш складніші кожного разу, тому книга містить багато фрактальних візерунків від найпростіших до складніших. У другій частині боти були одержимі малюванням фракталів. Кідрук чудово поєднав художній твір та програмування з графікою.

Сніжинка Коха чітко розписана та продемонстрована:

“…Спершу програміст перетворив трикутник на шестикутну зірку, як це раніше зробив бот. Потім кожну зі сторін отриманого багатокутника поділив на три частини. До кожної середньої ланки приліпив рівносторонній трикутник з довжиною сторони, втричі меншою за сторону зірки Давида. Нутро заповнив чорним кольором. У результаті вийшла фрактальна сніжинка Коха на третій ітерації…”

Також і простіші візерунки:

“… — Але що це?

— Самоподібна крива Пеано. «Виродок» у класичній геометрії. Я міг би продовжувати побудову, до нескінченності дроблячи сторони. При цьому, скільки б я не наближав криву, вона завжди виглядатиме однаково.

— А чому «виродок»?

— Після перетворення площа фігури, обмежена кривою, не змінюється: один квадрат «вирізається», але тут же такий самий додається. Зате довжина ламаної на кожній ітерації зростає в 1,8 раза. Якщо продовжувати перетворення, крива безмежно зростатиме. Через це виникає парадокс: лінія нескінченної довжини обмежує фігуру з цілком конечною площею…”

Отже, у літературі фрактальні зображення поширені та популярні, чи то структурою, чи візерунки, чи згадки принципу. Дуже рекомендую для прочитання книгу “Бот” українського автора, у якому детально описано побудову фрактальних зображень.

4.3. Фрактали в мистецтві

Художники, скульптори та композитори задовго використовували фрактальні візерунки як художній інструмент. У кінці 1940-х Джексон Поллок створив декілька картин, де використав дві методики створення хаотичних малюнків, а саме:

a. Для широкого діапазону масштабу руху в процесі малювання він задіяв усе своє тіло.
b. Він наносив фарбу, просто даючи її капати на полотно, таким чином кожен окремий шар складався з фрактального візерунка.

Ешер створював свої тесселяції, які популярні у всьому світі, завдяки нескінченним повторенням та великою масштабністю. Сальвадор Далі у "Обличчі війни" представив фрактальний прогрес смертельних масок, що зменшуються. Якщо не знати, що таке фрактали, можна бути здивованим їх популярністю навколо.

Навіть у музиці вони наявні. Наприклад, канони та фуги. Канон – це композиція, у якій мелодія повторюється одним або кількома голосами після певного інтервалу. Перша мелодія називається лідером, а її імітація – фоловером, що може бути точною копією або його видом. Існує декілька типів канонів. Фуга – складніша форма композиції, де основна тема послідовно звучить в кожного голосі, ніби імітація. Таким чином, дослідження канонів і фуг Баха підтверджує використання самоподібності та масштабування. Навіть музика Бетховена та Моцарта містить ці елементи.

Отже, знаючи методи та структуру фракталів можна побачити їх всюди, бо протягом історії різноманітні діячі, не знаючи навіть, що це фрактали, використовували їх. Очевидно, що цей неймовірно важливий хаос у мистецтві є особливістю завдяки своїй ієрархічній природі. Вона присутня в різних формах у всі історичні періоди. До речі, фрактали стали частиною математики, проте задовго до того вони панували у мистецтві, завдяки чому є такими популярними.

5. Моделювання рослинності в комп'ютерних іграх

5.1. Застосування L-системи в програмі SpeedTree

SpeedTree – це комерційна програма, що дозволяє генерувати моделі рослин, які можна використати в різноманітних іграх.

Чому це цікаво та корисно?

- Швидке редагування (За допомогою інструментів можна згинати, редагувати базову модель)
- Наявна велика платна бібліотека (Рослини мають безмежну випадковість, повний спектр змін і налаштовану анімацію вітру)
- Просте інтегрування та висока продуктивність (Складна упаковка текстур, рендеринг для великих масштабів)

Створення фрактальних алгоритмів відбувається через L-систему. Завдяки інтерфейсу програми можна зробити дерева, рослини унікальними, проте зберегти реалістичну структуру, а саме кути розгалуження, довжини, вигин гілок, товщини стовбура.

5.2. Використання фракталів у грі Days Gone та Dying Light 2

Для прикладу взято гру Days Gone, у якій програма SpeedTree використана для створення рослинності – кущів, рослин, дерев, які реагують на навколишнє середовище, дозволяючи гравцяв зосередитись на реальності в грі. Це підтверджено в технічних інтерв'ю та документації розробників Bend Studio. У Days Gone це особливо помітно завдяки густим лісам і плавній анімації листя на вітрі.

Аналіз гри Days Gone: обрано під час проходження декілька картинок природи. Серед обраних мною фракталів можна побачити на зображеннях, що є скриншотами з гри, фрактальний вогонь та листок Барнслі, інші види, такі як множина Канта, сніжинка Коха важко зообразити в комп'ютерній графіці, бо це математичні об'єкти. Щодо кривої Дракона, то на зображеннях не спостерігається, вона відображається в явно вигнутих кривих.

Скриншот 1. Природна реалізація листка Барнслі навколо.

Скриншот 2. Ефекти освітлення на мотоциклі, туману створені за допомогою фрактального вогню.

Наступний приклад – гра Dying Light 2. Деякі елементи мають фрактальну структуру, але не таку явну.

Аналіз гри Dying Light 2: також використано SpeedTree через L-систему. Тому можна проаналізувати схожість з попередньою грою, а саме фрактальних зображень листка Барнслі та фрактального вогню, проте не такі чіткі.

Скриншот 1. Листок Барнслі, представлений у вигляді папороті в нижній частині картинки.

Скриншот 2. Праворуч у центрі видно жовті іскри, які є схожими на фрактальний вогонь.

Отже, на зображеннях з гри Days Gone та Dying Light 2 можна побачити використання фрактальних візерунків, а саме в елементах природного середовища. Найчіткішим прикладом є листя папороті, що відтворює листок Барнслі, який створений за допомогою ітеративних систем. Також можна припустити застосування фрактального шуму, які візуально нагадують фрактальний вогонь у зонах туману на фоні чи іскр.

ВИСНОВОК

У цій роботі досліджено математичні алгоритми та побудову фрактальних зображень за допомогою мови програмування python та різними бібліотеками, адаптовані для роботи в середовищі Colab та виявлено їх у комп'ютерних іграх, літературі, мистецтві та медицині.

Цілі наукової роботи:

- проаналізовано предметну область та обґрунтовано алгоритми;
- реалізовано покрокову побудову фракталів з графічною візуалізацією;
- порівняно фрактали за критеріями швидкості виконання, візуальної складності та фрактальної розмірності;
- розглянуто приклади застосування фрактальних структур у моделюванні реального світу, зокрема в медицині, літературі, мистецтві та іграх;
- знайдено у комп'ютерних іграх.

Реалізовано зображення:

1.Множини Канта;

2.Кривої Дракона;

3.Сніжинки Коха;

4.Фрактального вогню;

5.Листка Барнслі;

Особливу увагу присвячено літературі, виявлено фрактальні алгоритми у віршах, книгах. Також на прикладах досліджено комп'ютерні ігри на побудову фрактальних зображень, детально визначено щодо рослиності та обгрунтовано використання їх на реалістичних пейзажах.

Щодо практичної частини, то не лише розгорнуто та детально описано фрактальні візерунки, а й досліджено їх у справжній прикладах комп'ютерної графіки – грі.

Таким чином, заплановані цілі досягнути у цій роботі. Наукова новизна виявлена, за допомогою аналізу комп'ютерних ігр. Робота має практичну та теоретичну цінність, завдяки якій можна отримати знання щодо фракталів та їх застосування та використати їх в подальшому аналізі в комп'ютерній графіці, у моделюванні в медицині, для загального розвитку. Проаналізовано їх відмінність в різноманітних аспектів, як-от: швидкість, фрактальна розмірність, складність зображення.

Практична частина -
https://colab.research.google.com/drive/1Bbl8nJsK0goryuiqNDqDDuD25XgIpRRJ?usp=sharing

Список використаної літератури:

1. Fractals in Computer Graphics [Електронний ресурс]. – URL: https://www.geeksforgeeks.org/fractals-in-computer-graphics/
2. *Fractal Literature*. Yale University. [Електронний ресурс]. – URL: https://gauss.math.yale.edu/fractals/Panorama/Literature/Lit/Literature.html
3. Smith T. G. Fractals in Biology and Medicine // *Journal of the Royal Society Interface*. – 2020. – № 17(172). – URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC7673317/
4. Cantor Set [Електронний ресурс] // GraphicMaths. – URL: https://graphicmaths.com/fractals/fractals/cantor-set/
5. Basics of Flame Fractals [Електронний ресурс] // Fractal Formulas. – 01.05.2017. – URL: https://fractalformulas.wordpress.com/2017/05/01/basics-of-flame-fractals/
6. Draves S. The Fractal Flame Algorithm [Електронний ресурс]. – URL: https://flam3.com/flame_draves.pdf
7. Halder A. Fractals in Medicine: Unlocking Nature's Patterns [Електронний ресурс] // LinkedIn. – URL: https://www.linkedin.com/pulse/fractals-medicine-unlocking-

natures-patterns-arindam-halder-jyp8f

8. Кідрук М. А. Бот : роман / Макс Кідрук. – Харків : Клуб сімейного дозвілля, 2013. – 496 с.
9. Mandelbrot B. The Fractalist: Memoir of a Scientific Maverick. – New York : Knopf Doubleday Publishing Group, 2012.
10. Richardson L. F., Ashford O. M., Drazin P. G. The Collected Papers of Lewis Fry Richardson. – Cambridge : Cambridge University Press, 2009.
11. NOVA. Hunting the Hidden Dimension [Електронний ресурс]. – PBS, 2008. – URL: http://www.pbs.org/wgbh/nova/physics/hunting-hidden-dimension.html
12. NVIDIA. GPU Gems [Електронний ресурс]. – NVIDIA Corporation, 2007. – URL: https://developer.nvidia.com/gpugems/GPUGems/
13. Tessendorf J. Simulating Ocean Water. – Computer Graphics Laboratory, 2005.
14. Hu Y., Velho L., Tong X., Guo B., Shum H. Realistic, Real–Time Rendering of Ocean Waves // *Computer Animation and Virtual Worlds*. – 2006. – Vol. 17, № 1. – С. 59–67. – DOI: 10.1002.
15. Bird K., Dickerson T., George J. Techniques for Fractal Terrain Generation // *HRUMC*. – 2013.
16. Feldman D. P. Chaos and Fractals: An Elementary Introduction. – Oxford : Oxford University Press, 2012.
17. Worboys M. F., Duckham M. GIS: A Computing Perspective. – 2nd ed. – Boca Raton : CRC Press, 2004.
18. Ebert D. S. Texturing & Modeling: A Procedural Approach. – 1994. – ISBN 978-0122287602.

19. Boiangiu C.-A., Raducanu B. 3D Mesh Simplification Techniques for Image-Page Clusters Detection // *WSEAS Transactions on Information Science and Applications*. – 2008. – Vol. 5, № 7. – C. 1200–1209.
20. Saunders R. L. Terrainosaurus: Realistic Terrain Synthesis Using Genetic Algorithms. – December 2006.
21. Belhadj F., Audibert P. Modeling Landscapes with Ridges and Rivers // *Proceedings of the 3rd International Conference on Computer Graphics and Interactive Techniques, Australasia and South East Asia*. – November 2005. – C. 447–450.