

## 2. Фрактальна 3D-графіка

### 2.1. Фрактали

Слово "фрактал" з'явилося завдяки геніальному вченому Бенуа Мандельброту. Він увів цей термін у сімдесятих роках минулого століття, запозичивши слово *fractus* із латини, де воно буквально означає ламаний або роздрібнюваний. Сьогодні під словом "фрактал" найчастіше прийнято мати на увазі графічне зображення структури, яка в більшому масштабі подібна сама до себе.

Французький математик Лейбніц побудував фрактал на основі простої формули, проітерованої циклом зворотного зв'язку. Це означає, що конкретне значення числа знаходиться на основі попереднього.

Мандельброт був першим, хто використовував комп'ютер для прорахунку класичного фрактала. Обробивши послідовність, що складається з великої кількості значень, Бенуа переніс результати на графік (рис. 2.1).

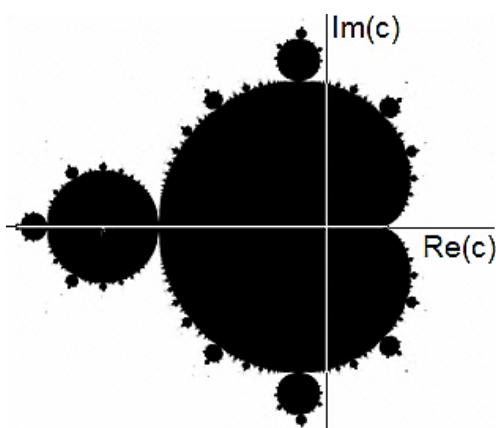


Рис. 2.1. Фрактал Мандельброта

Згодом це зображення було розфарбоване і стало одним із найпопулярніших зображень, які тільки були створені людиною (рис. 2.2).

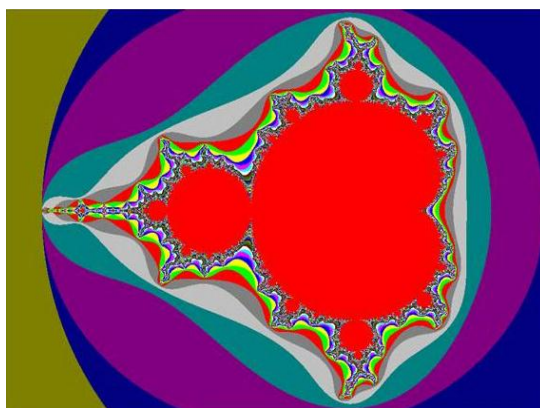


Рис. 2.2. Зображення фрактала Мандельброта

## 2.2. Сфера застосування фракталів

Теорія фракталів знайшла практичне застосування в художників.

Засновник студії "Pixar" аніматор Лорен Карпентер вивчив принципи фрактальної геометрії і знайшов спосіб реалізувати її в комп'ютерній графіці. Принцип, який використовував Лорен для досягнення мети, полягав у тому, щоб поділяти більшу геометричну фігуру на дрібні елементи, а ті, у свою чергу, ділити на аналогічні фігури меншого розміру (рис. 2.3).

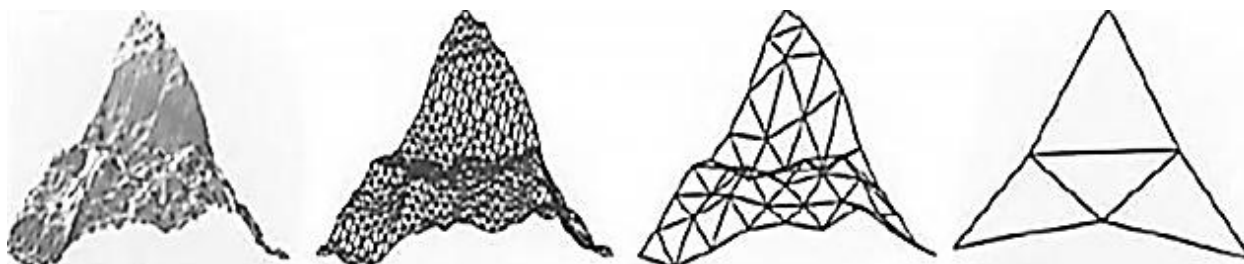


Рис. 2.3. Принцип поділу великої геометричної фігури

Використовуючи більші трикутники, Карпентер дрібнив їх на чотири дрібніших і потім повторював цю процедуру знову, поки в нього не виходив реалістичний гірський ландшафт. Таким чином, йому вдалося стати першим художником, який застосував у комп'ютерній графіці фрактальний алгоритм для побудови зображень.

Нині всі популярні додатки для створення 3D-зображень використовують аналогічний принцип генерування природних об'єктів.

Фрактальні зображення застосовуються для дизайну інтер'єру, прикрашання одягу, аерографії та в інших галузях.

Для дизайну інтер'єру використовуються різні картини і постери, які кардинально перетворюють приміщення (рис. 2.4).



Рис. 2.4. Дизайн інтер'єру

Картини, створені на основі фрактальної графіки, дозволяють візуально збільшити простір кімнати, інші надають відтінки затишку, треті містять певну інформацію.

Друк фрактальної графіки на одязі дозволяє якісно й ефектно виразити свою індивідуальність, реалізувати свій смак і своє бачення світу. Фрактальну графіку друкують на футболках, на толстовках, на бейсболках, сумках, навіть на шкарпетках і парасольках (рис. 2.5).



**Рис. 2.5. Фрактальна графіка на футболці**

Фрактальні зображення, нанесені за допомогою аерографії, прикрашають автомобілі (рис. 2.6).



**Рис. 2.6. Фрактальне зображення на кузові автомобіля**

Аерографія – це не просто модний тюнінг, але і свого роду сучасне мистецтво. Виконання рисунка відбувається порошковими барвниками за допомогою стиснутого повітря.

### 2.3. Математичний опис фракталів

Фрактали дозволяють описувати цілі класи зображень, для детального опису яких потрібно відносно мало пам'яті. Програмні засоби для роботи з фрактальною графікою призначені для автоматичної генерації зображень шляхом математичних розрахунків. Створення фрактальної художньої композиції полягає не в малюванні або оформленні, а в програмуванні. Таким чином, для створення конкурентоспроможних і комерційних фракталів потрібне знання чисельних методів.

Математичною основою фрактальної графіки є фрактальна геометрія. У сучасній науково-технічній літературі розглядається велика кількість різних методів синтезу фрактальних зображень. Для того щоб розібратися в них, треба їх класифікувати. Існують класифікації, наприклад, рукотворні та природні. До рукотворних належать ті фрактали, які були придумані вченими, вони за будь-якого масштабу мають фрактальні властивості. На природні фрактали накладається обмеження на область існування, тобто максимальний і мінімальний розміри, за яких в об'єкта спостерігаються фрактальні властивості. Уся рукотворна множина фракталів поділяється на геометричні, алгебраїчні і стохастичні.

Геометричні фрактали – найнаочніші, тому що в них відразу видно самоподібність. Прикладом таких кривих слугує трикутник Серпінського. Побудова трикутника Серпінського заснована на методі Iterated Function System (IFS) [1, 3]. IFS – це сукупність стискальних афінних перетворень. Як відомо, афінне перетворення включає масштабування, поворот і паралельне перенесення. Афінне перетворення вважається стискальним, якщо коефіцієнт масштабування менший від одиниці.

Сутність методу IFS можна пояснити за допомогою перетворень. Помістимо прямокутний трикутник з довгим катетом, що дорівнює одиниці, на комплексну площину  $Z$  (рис. 2.7).

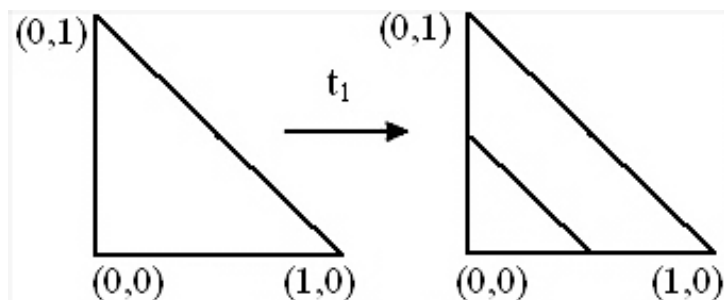


Рис. 2.7. Стискальне перетворення на дійсній осі

Задамо лінійне перетворення  $t_1$ , яке переводить на комплексній площині рівносторонній трикутник у такий, що вдвічі менший за розміром (2.1):

$$t_1 : f_1(Z) = \frac{Z}{2}. \quad (2.1)$$

Якщо тепер змістити цей маленький трикутник по горизонталі вправо на величину, що дорівнює  $1/2$ , то отримаємо перетворення  $t_2$ , що переводить початковий трикутник у трикутник, поданий на рис. 2.8 справа.

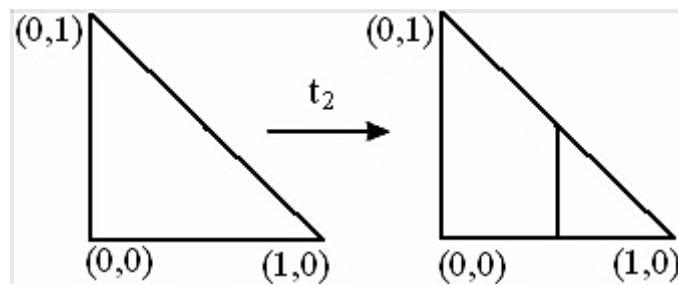


Рис. 2.8. Зсув на дійсній осі

Відповідна цьому перетворенню функція дорівнює (2.2):

$$t_2 : f_2(Z) = \frac{Z+1}{2}. \quad (2.2)$$

Останній, третій маленький трикутник виходить за допомогою перетворення  $t_3$ , поданого на рис. 2.9.

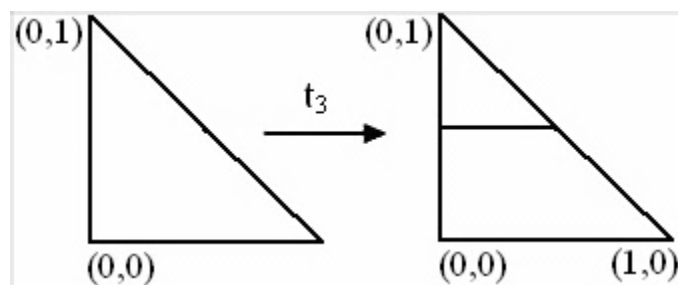


Рис. 2.9. Стискальне перетворення на уявній осі

Функція  $f_3(Z)$  виходить з  $f_1(Z)$  із трансляцією на комплексний вектор (2.3):

$$t_3 : f_3(Z) = \frac{Z+i}{2}. \quad (2.3)$$

У результаті три наведені лінійні функції виконують шукане стискальне перетворення початкового трикутника у три трикутники у два рази меншого розміру. Якщо тепер кожен із цих трьох маленьких трикутників піддати цим трьом стискальним перетворенням, то тоді виникне вже 9 трикутників, у 4 рази менших за початковий. Таким чином, виконуючи цю процедуру певну кількість разів, отримаємо трикутник Серпінського.

Декілька різних варіантів реалізації трикутника Серпінського наведені на рис. 2.10. Перший є класичним трикутником Серпінського, інший імітує "комаху" і третій – архітектурну споруду.

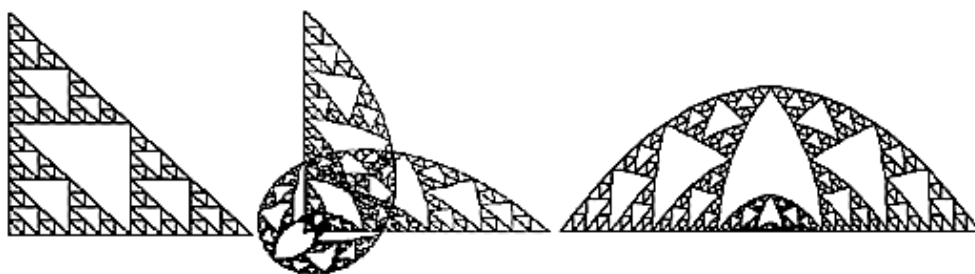


Рис. 2.10. Варіанти трикутника Серпінського

Для побудови фракталів алгебри використовуються ітерації нелінійних відображень, що задаються простими формулами алгебри. Прикладами фракталів алгебри є множини Мандельброта і Жюліа.

Алгоритм побудови фракталів алгебри досить простий і заснований на створенні фрактальних об'єктів на площині. Сутність алгоритму полягає в тому, що одне комплексне число  $Z_n = x_n + iy_n$  перетвориться в інше комплексне число  $Z_{n+1} = x_{n+1} + iy_{n+1}$  за ітераційним правилом, яке подане формулою  $Z_{n+1} = f(Z_n)$ , де  $f(Z_n)$  – певна нелінійна функція  $Z_n$ ,  $n$  – номер ітерації.

Множина Жюліа поряд із безліччю Мандельброта посідає центральне місце в теорії фракталів. Множина Жюліа є межею області тяжіння нерухомих точок комплексних відображень.

Найбільш вивченим із них є відображення, подане формулою (2.4):

$$f(Z) = Z^2 + c, \quad (2.4)$$

де  $c = c_1 + ic_2 - \text{const}$ ,  $Z$  і  $c \in \mathbb{C}$ ;

$\mathbb{C}$  – множина всіх комплексних чисел.

Розглянутий алгоритм можна подати як прямий і заснований на такій теоремі. Припустимо, що  $|c| < 2$ . Нехай  $z \in \mathbb{C}$  і нехай  $Z_n = f(n)(Z)$  для  $n = 1, 2, 3$ . Якщо існує таке  $n$ , що  $|Z_n| \Rightarrow 2$ , то має місце (2.5):

$$\lim_{n \rightarrow \infty} Z_n = \infty, \quad (2.5)$$

тобто орбіта (2.6):

$$\left[ f^{(n)}(Z) \right]_{n-1}^{\infty} \quad (2.6)$$

прагне до нескінченності і  $Z$  не належить множині Жюліа. На цій теоремі заснований перший алгоритм – побудова на основі початкового відображення.

Другий алгоритм побудови, заснований на оберненому відображенні, спирається на властивість множини Жюліа, яка полягає в тому, що для зворотного відображення, тобто (2.7):

$$f^{-1}(Z) = \sqrt{Z - C} \quad (2.7)$$

множина Жюліа – атрактор.

Результати реалізації наведених алгоритмів подані на рис. 2.11.

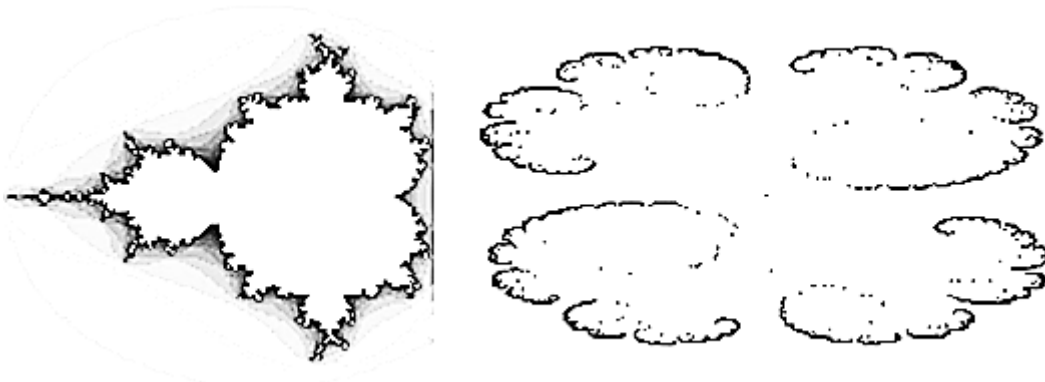


Рис. 2.11. Множини Мандельброта і Жюліа

Ще одним відомим класом фракталів є стохастичні фрактали, які отримують у тому випадку, коли в ітераційному процесі випадковим чином змінюють які-небудь його параметри. У цьому разі виходять об'єкти, дуже схожі на природні, – несиметричні дерева, порізані берегові лінії і т. д. Двовимірні стохастичні фрактали використовуються під час моделювання рельєфу місцевості і поверхні моря.

Алгоритм побудови стохастичних фракталів ще називають грою в "хаос", у якості початкової множини вибирають одну точку:

$x_0$  – початкова точка (з довільними координатами),

$$x_1 = T_1(x_0) \text{ або } T_2(x_0) \text{ або } T_3(x_0),$$

$$\dots$$

$$x_n = T_1(x_{n-1}) \text{ або } T_2(x_{n-1}) \text{ або } T_3(x_{n-1}).$$

На кожному кроці, замість того щоб застосовувати відразу три перетворення  $T_1(S)$ ,  $T_2(S)$ ,  $T_3(S)$ , застосовується тільки одне, вибране випадковим чином. Таким чином, на кожному кроці виходить рівно одна точка. Виявляється, що після певного перехідного процесу точки, згенеровані алгоритмом, точно заповнюють фрактальне зображення.

Чудовою властивістю алгоритмів, заснованих на теорії IFS, є те, що їхній результат (атрактор) абсолютно не залежить від вибору початкової множини  $E_0$  або початкової точки  $x_0$ . У разі детермінованого алгоритму це означає, що в якості  $E_0$  можна взяти будь-яку компакту множину на площині: гранична множина, як і раніше, співпадатиме із заданим фрактальним зображенням. У разі застосування алгоритму побудови стохастичних фракталів вибір початкової точки  $x_0$  не впливає на формування заданого фрактального зображення.

Для рівномірного розподілу точок по екрану у алгоритмі побудови стохастичних фракталів афінне перетворення слід вибирати з імовірністю (2.8):

$$P_i = \frac{\det(A_i)}{\sum_{j=1}^m \det(A_j)}, \quad (2.8)$$

де  $A_i$  – матриця відповідного афінного перетворення ( $i = 1, 2, \dots, m$ ).

Очевидно, що  $p_1 + p_2 + \dots + p_m = 1$ .

Результат побудови фрактального дерева за допомогою алгоритму побудови стохастичних фракталів наведений на рис. 2.12 (фрактальне дерево).

Для побудови використовувалися стискальні афінні перетворення (2.9):

$$\begin{aligned} x_{n-1} &= a \cdot x_n + b \cdot y_n + e, \\ y_{n-1} &= c \cdot x_n + d \cdot y_n + f. \end{aligned} \quad (2.9)$$

Рівняння можна подати в матричному вигляді (2.10):

$$\begin{pmatrix} x_{n-1} \\ y_{n-1} \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} x_n \\ y_n \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}. \quad (2.10)$$





Рис. 2.12. Фрактальне дерево

У загальному випадку афінне перетворення на площині задається шістьма незалежними дійсними числами. Два числа  $e$  і  $f$  описують переміщення фігури на площині, а чотири числа  $a$ ,  $b$ ,  $c$ ,  $d$  задають обертання фігури за незмінного початкового положення.

За допомогою алгоритму побудови стохастичних фракталів створюються складні фрактали (рис. 2.13).

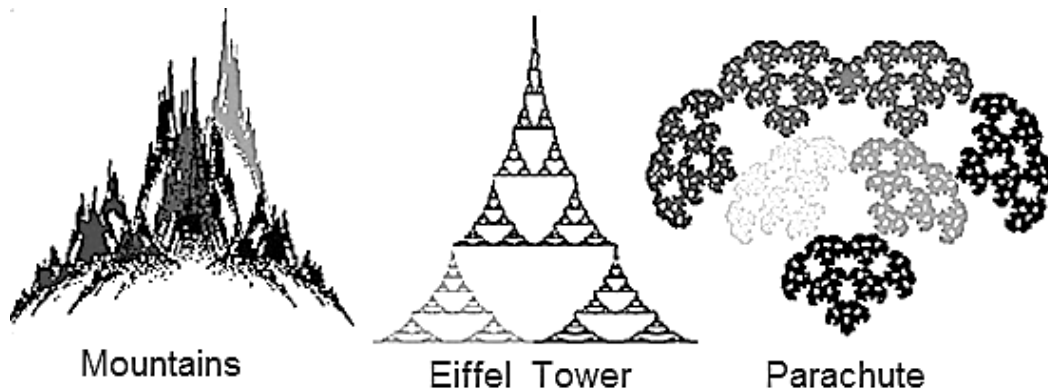


Рис. 2.13. Стохастичні фрактали

Існують фантастичні зображення рослин, отримані за допомогою програм. Це так звані L-системи, які становлять віртуальний програмований пристрій, що складається з пера, керованого ним механізму і аркуша паперу. Механізм, що управляє пером, дозволяє:

накреслити відрізок прямої заданої довжини в напрямку поточної орієнтації пера (команда  $f$ );

змінити орієнтацію пера відносно поточної позиції на заданий відносний кут (команди + і –);

запам'ятати поточний стан (команда [);

згадувати стан (команда ]), що раніше запам'ятав.

Під станом у цьому випадку розуміють трійку чисел  $(x, y, a)$ , де  $x$  і  $y$  – це координати пера і  $a$  – це кут, що визначає напрям орієнтації пера. Таким чином, задавши певний початковий напрям  $a_0$ , визначивши відносний кут повороту  $\alpha$  і задавши довжину відрізка, за допомогою послідовності команд  $f + f + f + f$  можна намалювати квадрат. Визначивши відносний кут повороту в  $60^\circ$ , за допомогою послідовності команд  $f + + f + + f$  можна намалювати рівносторонній трикутник.

На рис. 2.14 наведено два приклади фрактальних структур, побудованих за допомогою L-системи.

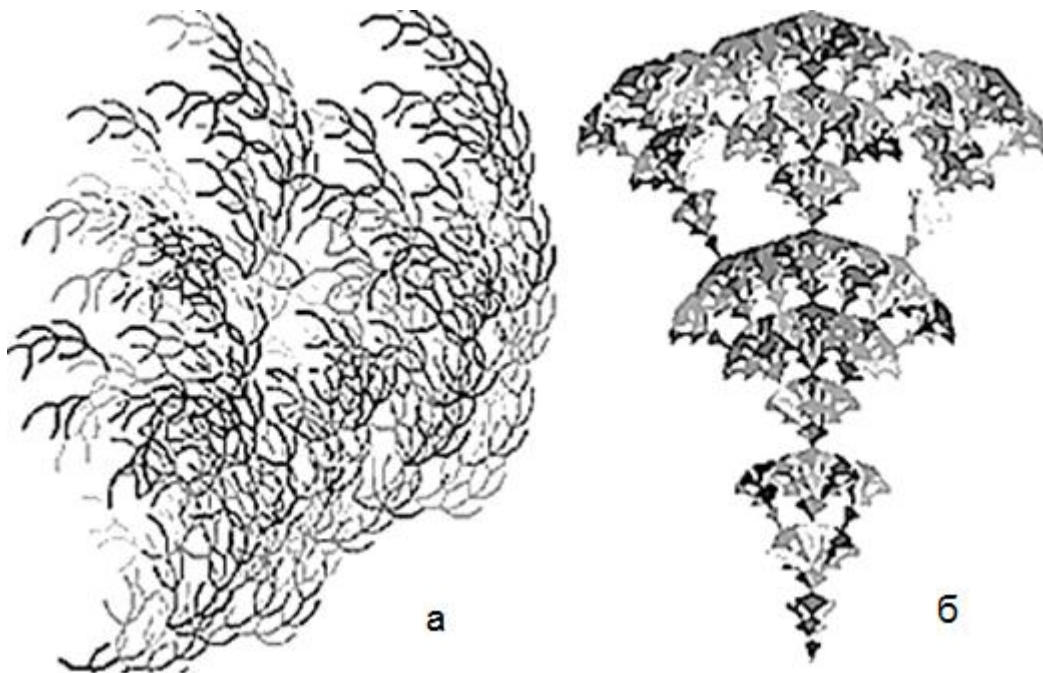


Рис. 2.14. Фрактали, побудовані за допомогою L-системи

Для створення фрактальних структур рис. 2.14 (а і б) використовують відповідні рядки команд:

$a = \{ " ", "- ", "f", "+", "f", "+", "[", "+", "f", "- ", "f", "- ", "]", "- ", "[", "- ", "f", "+", "f", "+", "f", "]" \};$

$b = \{ " ", "f", "[", "+", "f", "f", "]", "[", "- ", "f", "f", "]", "f", "[", "- ", "f", "]", "[", "+", "f", "]", "f" \}.$

Метод L-систем дозволяє зменшувати або збільшувати глибину рекурсії гілки. А змінюючи закон руху, можна отримувати різні зображення (рис. 2.15).



Рис. 2.15. Фрактали, побудовані за допомогою L-систем

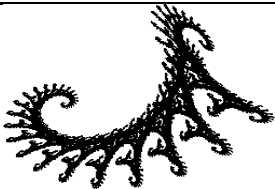

Потрібно відзначити, що фрактали – це область математичного мистецтва і інтуїції, коли за допомогою найпростіших формул і алгоритмів виходять картини надзвичайної краси і складності. У контурах побудованих зображень нерідко вгадуються листя, дерева і квіти. Одне з найбільш потужних додатків фракталів лежать в комп'ютерній графіці – це побудова ландшафтів, дерев, рослин і генерування фрактальних текстур.

#### 2.4. Завдання для самостійної роботи


Використовуючи стискальне афінне перетворення (2.9) і (2.10), згенеруйте у середовищі Mathcad фрактали, наведені у табл. 2.1.

Таблиця 2.1




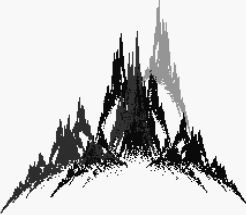

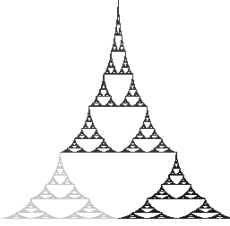
#### Фрактали

Вид фрактала	Значення коефіцієнтів (a, b, c, d, e, f)
1	2
	Dragon { 0.824074 0.281482 -0.212346 0.864198 -1.882290 -0.110607 0.787473 0.088272 0.520988 -0.463889 -0.377778 0.785360 8.095795 0.212527 }
	Spiral{ 0.787879 -0.424242 0.242424 0.859848 1.758647 1.408065 0.895652 -0.121212 0.257576 0.151515 0.053030 -6.721654 1.377236 0.052174 0.181818 -0.136364 0.090909 0.181818 6.086107 1.568035 0.052174 }

Продовження табл. 2.1

1	2
	<pre>Swirl { 0.745455 -0.459091 0.406061 0.887121 1.460279 0.691072 0.912675 -0.424242 -0.065152 -0.175758 -0.218182 3.809567 6.741476 0.087325 }</pre>
	<pre>Dendrite{ 0.0 -0.7 0.7 0.0 0.0 0.0 0.5 0.7 0.0 0.0 0.7 0.3 0.0 0.5 }</pre>
	<pre>Flower{ -0.87 0.32 -0.32 -0.87 0.00 0.00 0.89 -0.32 -0.11 0.11 -0.32 1.32 -0.11 0.11 }</pre>
	<pre>Windy{ 0.358591 0.329160 -0.334007 0.580720 -5.889050 1.924278 0.361396 0.796218 0.136942 0.075187 0.719074 1.086998 0.408452 0.638604 }</pre>
	<pre>Fern{ 0 0 0 0.165 0 0 0.01 0.84 -0.045 0.045 0.86 0 1.6 0.85 0.25 -0.26 0.23 0.25 0 1.6 0.07 -0.135 0.28 0.26 0.245 0 0.44 0.07 }</pre>
	<pre>Maple{ 0.49 -0.01 0 0.62 0.25 0.02 0.316 0.27 -0.52 0.4 0.36 0 -0.56 0.316 0.18 0.73 -0.5 0.26 0.88 -0.08 0.316 0.04 0.01 -0.5 0 0.52 -0.32 0.052 }</pre>
	<pre>Star{ 0.62 0 0 0.09 0 0 0.03 -0.79 0.55 -0.55 -0.79 1.79 0.55 0.97 }</pre>
	<pre>Leaf{ 0.000000 0.243900 0.000000 0.305300 0.000000 0.000000 0.010000 0.724800 0.033700 -0.025300 0.742600 0.206000 0.253800 0.710544 0.158300 -0.129700 0.355000 0.367600 0.138300 0.175000 0.137386 0.338600 0.369400 0.222700 -0.075600 0.067900 0.082600 0.142169 }</pre>

Закінчення табл. 2.1

1	2
	<pre>Tree{ 0.05 0 0 0.6 0 0 0.028000 0.05 0 0 -0.5 0 1 0.023256 0.459627 -0.321394 0.385673 0.383022 0 0.6 0.279070 0.469846 -0.153909 0.171010 0.422862 0 1.1 0.209302 0.433013 0.275 -0.25 0.476314 0 1 0.255814 0.421325 0.257115 -0.353533 0.306418 0 0.7 0.204651 }</pre>
	<pre>Spiral{ 0.143268 0.372630 0 0.411999 -0.424035 0.228385 0.05 0.817528 -0.472 0.472 0.817528 -0.06 0.051538 0.95 }</pre>
	<pre>Tree{ 0.458805 -0.225654 0.072682 0.601870 -0.001772 0.319478 0.31 0.342683 0.375610 -0.203368 0.546420 -0.022095 0.329764 0.20 0.135817 0.502731 -0.313469 0.138456 -0.020039 0.217063 0.16 0.253255 -0.489749 0.307978 0.349892 -0.007397 0.198101 0.23 0.066406 0.000000 0.000000 0.479424 -0.014923 -0.024215 0.10 }</pre>
	<pre>Mountains{ 0.5 0 0.2 0.4 -0.6 0 0.25 -0.2 0 0.2 0.5 0.3 0.8 0.25 -0.8 0 0 0.8 0.1 0 0.25 0.2 0 0 0.6 0 0.35 0.25 }</pre>
	<pre>Branch{ 0.409445 0.635563 0.699769 -0.364293 -1.362228 6.947697 0.6 0.440026 -0.218392 0.413335 0.376756 -3.087240 3.169590 0.25 0.107881 0.438639 0.084792 0.257114 -5.570585 1.155089 0.15 }</pre>
	<pre>Eiffel_Tower{ 0.5 0 0 0.3 0 0 0.246154 0.5 0 0 0.3 0.5 0 0.246154 0.5 0 0 0.7 0.25 0.3 0.507692 }</pre>

## 2.5. Інструментальні засоби для створення 3D-фракталів

### 2.5.1. XenoDream

XenoDream – призначена для створення тривимірних фракталів за допомогою спеціальних будівельних блоків (Holon).

Програма має стандартний інтерфейс (рис. 2.16), який включає:

заголовок;

робоче вікно;

панель інструментів вгорі і знизу від робочого вікна;

панель управління, що складається з декількох вкладок.

На вкладці File панелі управління знаходяться дві круглі кнопки швидкого старту:

зелена кнопка – для випадкового вибору фрактала з наявних заготовок;

синя кнопка – для випадкового вибору параметрів фрактала.

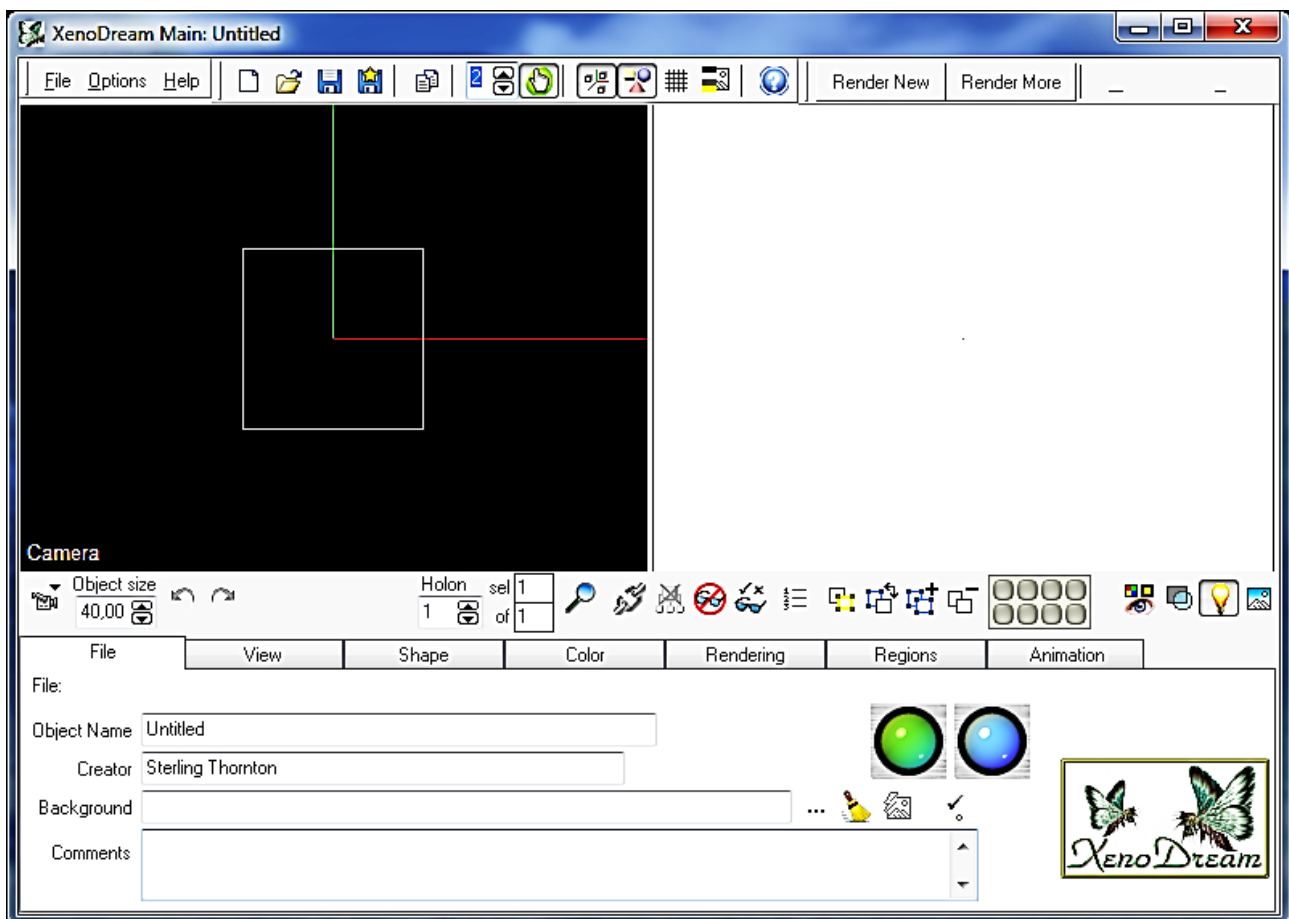


Рис. 2.16. Інтерфейс програми XenoDream

Потрібно натиснути на зелену кнопку кілька разів, щоб з'явився фрактал у вигляді листка папороті (рис. 2.17).

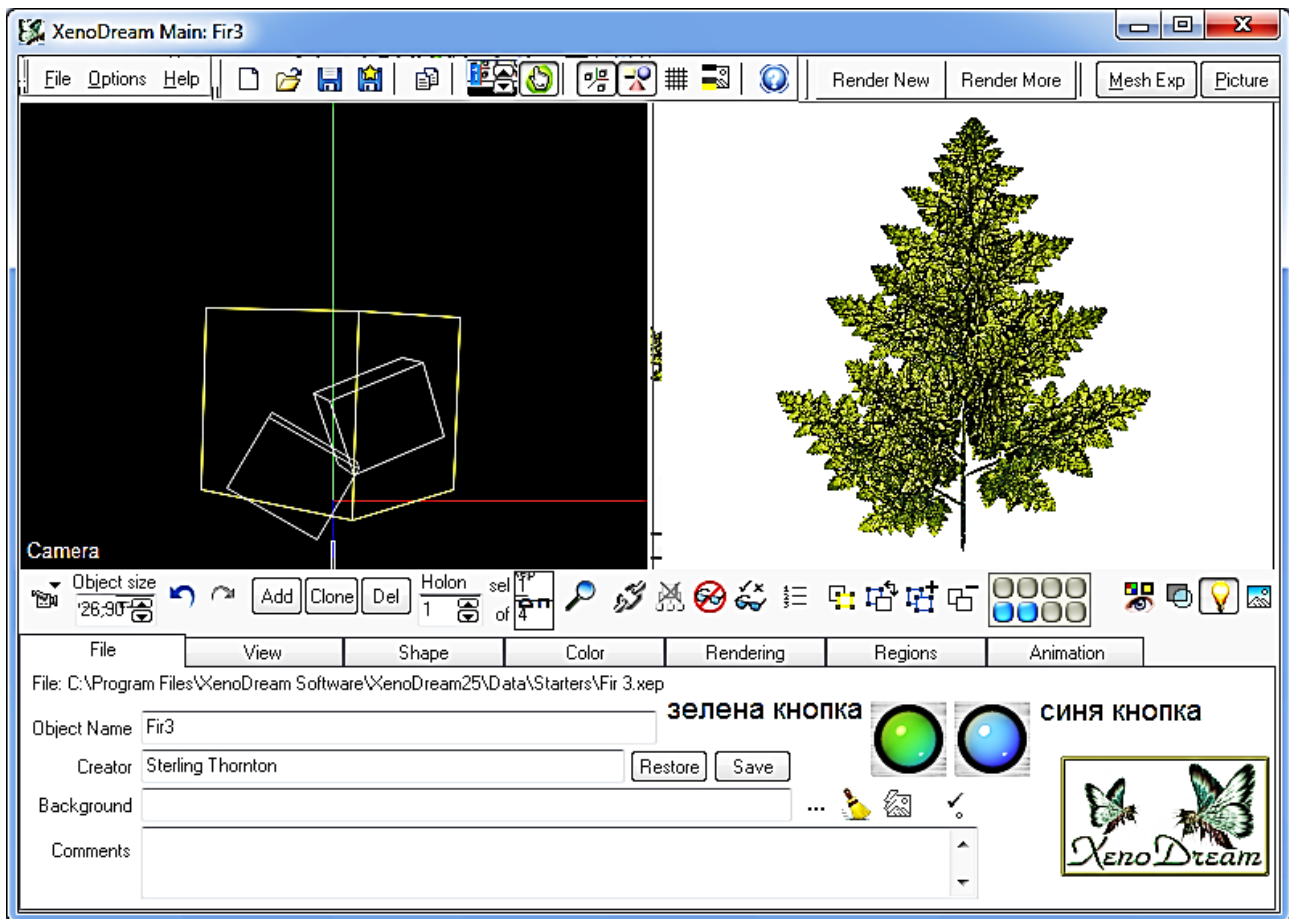


Рис. 2.17. Фрактал у робочому вікні програми

Жовтим підсвічується активний будівельний блок (Holon). Загалом фрактал складається з двох будівельних блоків.

Після завантаження фрактала в робоче вікно програми активізувався нижній ряд кнопок панелі інструментів. За допомогою цих кнопок виконується управління будівельними блоками.

Вкладка View панелі управління призначена для зміни позиціонування фрактала в робочому вікні (рис. 2.18).

Для управління позиціонуванням фрактала використовуються чотири елементи управління. Перші два переміщують фрактал уздовж осей X, Y і Z, а другі два елементи управління обертають фрактал навколо цих осей.

Сітка дозволяє фіксувати проміжні варіанти позиціонування фрактала. Вона може запам'ятовувати до 20 положень фрактала у просторі.

Вкладка Shape панелі управління дозволяє (рис. 2.19):

- управляти режимами позиціонування фрактала;
- центрувати і вирівнювати будівельні блоки;
- встановлювати стилі будівельних блоків;
- перетворювати (Metamorph) будівельні блоки.

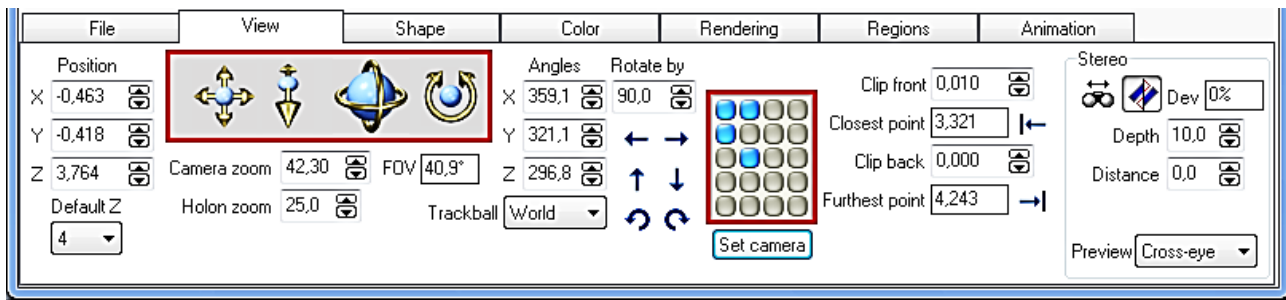


Рис. 2.18. Вкладка View панелі управління

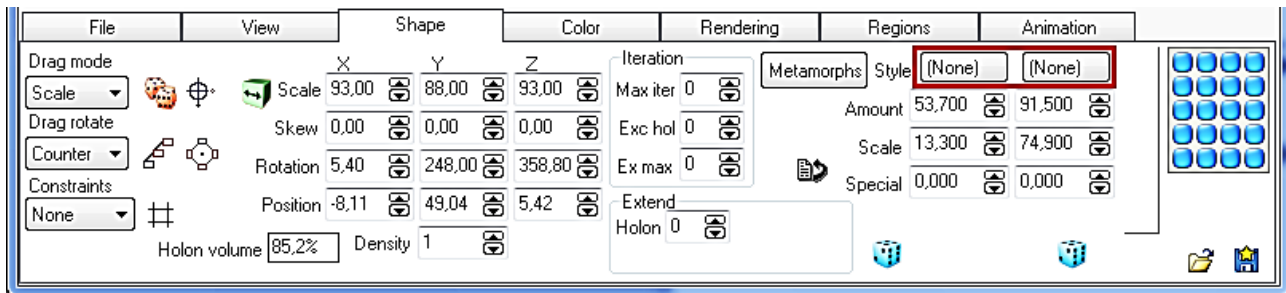


Рис. 2.19. Вкладка Shape панелі управління

Перетворення може бути застосоване до всіх будівельних блоків відразу або окремо. Зміна стилю будівельних блоків проводиться за допомогою кнопок. Кожен будівельний блок може мати до двох стилів, якщо натиснути на кнопку "None", з'являється вікно вибору стилів (рис. 2.20).

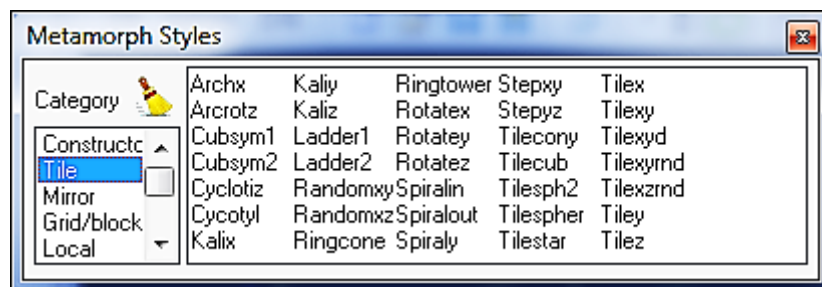


Рис. 2.20. Вікно стилів будівельних блоків

Вкладка Color панелі управління (рис. 2.21) призначена для: вибору методу роботи з кольором Holon Sequence і Metamorph; налаштування параметрів градієнта або вибору зі списку заздалегідь налагоджених.

Вкладка Rendering (рис. 2.22) призначена для: управління освітленням; налаштування прозорості будівельних блоків; встановлення розміру зображень.



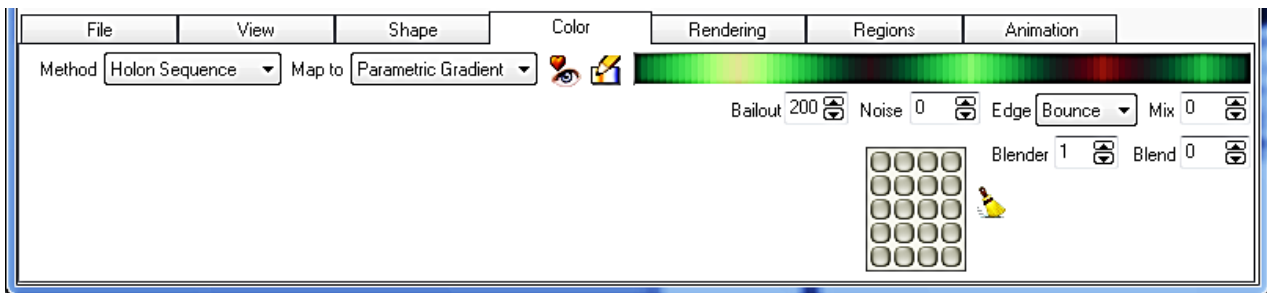


Рис. 2.21. Вкладка Color панелі управління

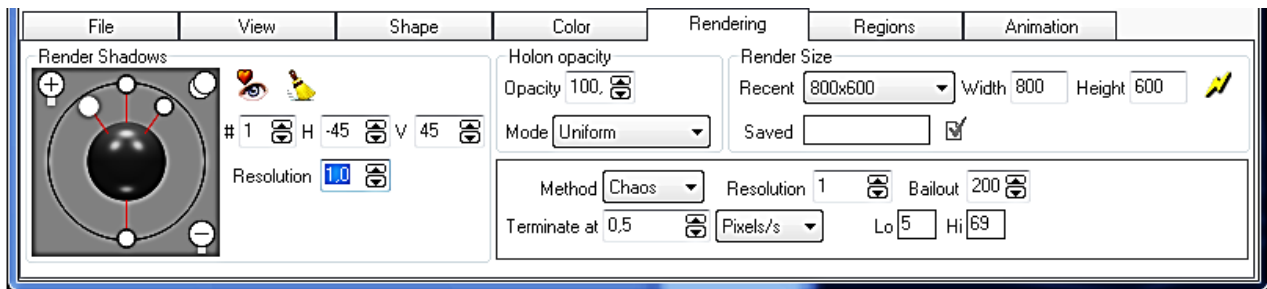


Рис. 2.22. Вкладка Rendering панелі управління

Після встановлення і налаштування параметрів вкладки Rendering натискаємо кнопку "Render new", відкривається вікно (рис. 2.23).

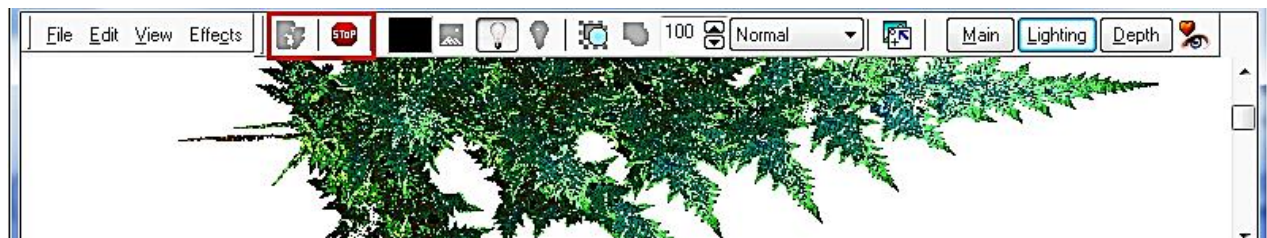


Рис. 2.23. Вікно промальовування фрактала

Елементи управління вікном:

File – зберігає і відкриває зображення фрактала і карту глибини;

Edit – копіює в буфер зображення фрактала;

View – встановлює розмір зображення у відсотках;

Effects – управляє ефектами, такими, як розташування фрактала, шум, інверсія, контрастність та ін.;

Stop – зупиняє поточний рендер зображення, потім його можна відновити, натиснувши кнопку "Start new render".

Крайня права кнопка дозволяє вибрати заздалегідь встановлені параметри рендера.

### **2.5.2. Incendia**

Головне вікно програми Incendia можна умовно розподілити на дві частини – вікно роботи з фракталом і вікно попереднього перегляду.

Вікно роботи з фракталом (рис. 2.24) містить сім вкладок: Fractals, Render, Camera, Editors, About, Volume і Animat.

Перша група елементів управління на вкладці Fractals дозволяє вибрати сам фрактал, натиснувши на фрактальне ім'я. Після вибору фрактала у вікні перегляду з'являється зображення з випадковим набором різних параметрів. В останніх версіях програми міститься понад 40 типів фракталів.

Далі йде коротка інформація про вибраний фрактал:

ім'я фрактала;

кількість перетворень;

кількість базових форм.

Наступна група елементів управління кнопки завантаження і збереження параметрів:

Save Parametres і Load Parametres – кнопки для збереження в текстовому файлі, у якому визначені всі параметри програми і виконується завантаження вже раніше збережених параметрів (розширення файла "\*.par");

Save Buffers і Load Buffers – кнопки для збереження всіх параметрів програми у буфері на конкретний момент і вивантаження їх із буфера (розширення файла "\*.Zb3");

Save Script і Load Script – кнопки збереження і завантаження скриптів. Кнопку Save Script видно тільки у деяких фрактальних типах, що дозволяє користувачеві експортувати вибрані для фрактала сценарії (розширення файла "\*.PRG").

Сценарій може бути змінений у будь-якому текстовому редакторі. Кнопку Load Script видно тільки в процесі вибору фрактала Incendia Script. За допомогою цієї кнопки можна завантажити будь-який Incendia Script.

Робоче вікно вкладки Render (рис. 2.25) має чотири вкладки:

Materials – налаштування текстур об'єктів;

Environment – налаштування середовища;

Shaders – налаштування шейдерів;

Advanced – просунуті елементи управління і налаштування.

У верхній частині вкладки знаходиться група кнопок:

Save Bitmap – кнопка збереження фрактала у BMP-файлі;

Save Mat. і Load Mat. – кнопки збереження і завантаження конфігурації фактичних матеріалів, у тому числі текстури;

Clear Fog – кнопка, що очищає буфер туману;

Clear Shadows – кнопка, що очищає тінювий буфер;

Save PNG – збереження фрактала у png-файлі.

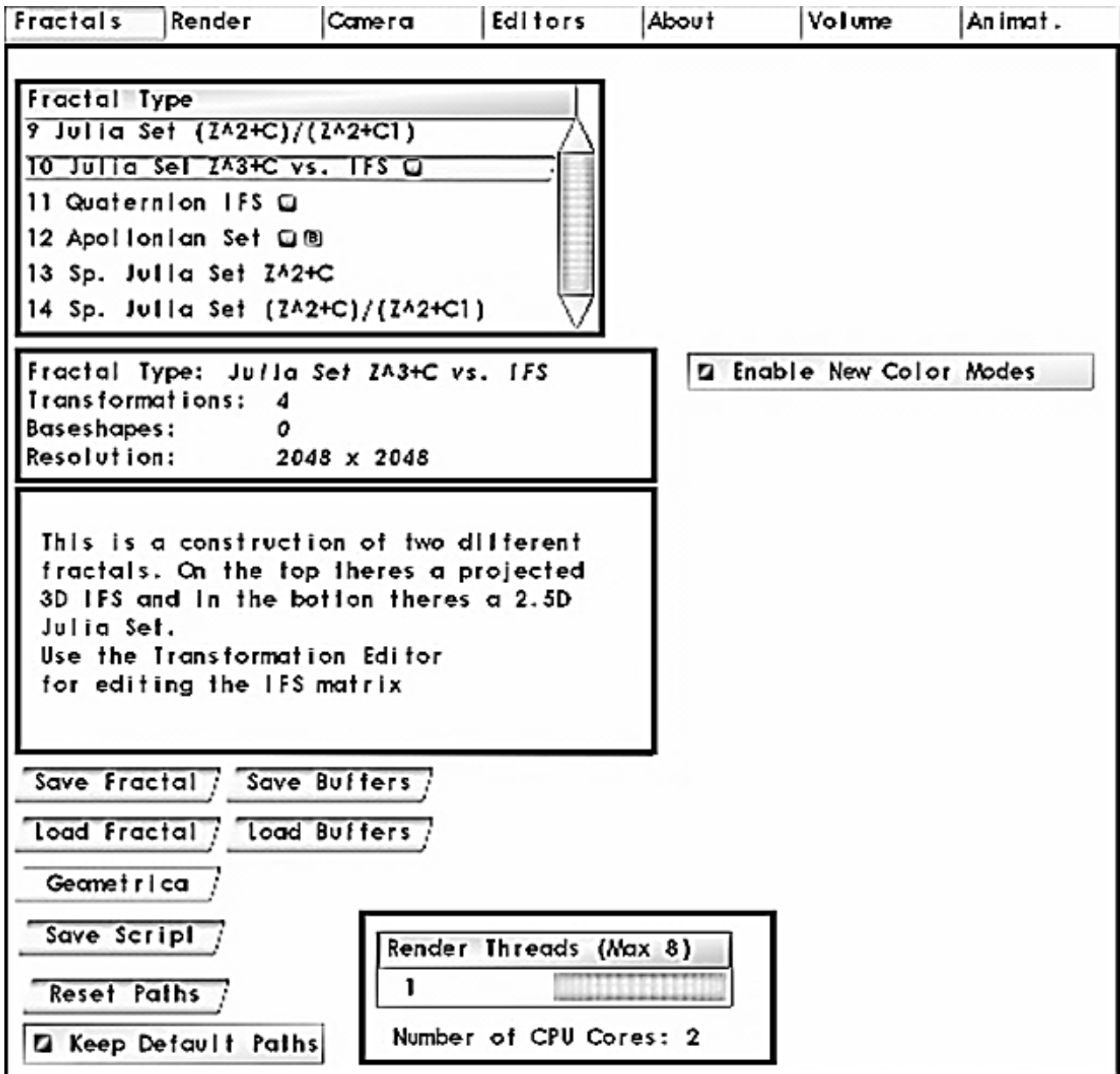


Рис. 2.24. Вікно роботи з фракталом

На вкладці Camera знаходяться елементи управління камерою. Програма Incendia дозволяє створювати анімовані 3D-фрактали.

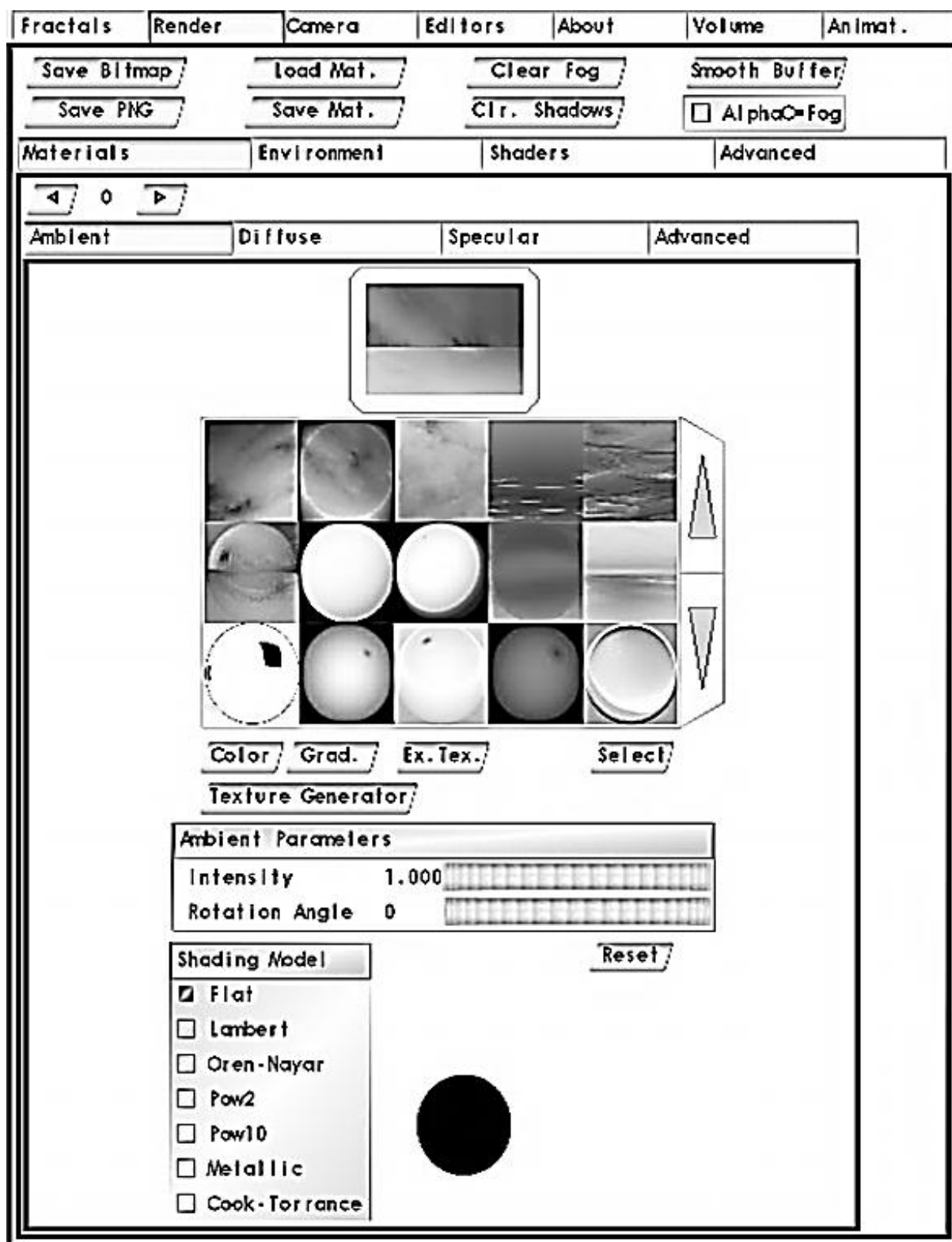


Рис. 2.25. Вікно вкладки Render

Вкладка Editors призначена для створення фрактала на основі базових форм (рис. 2.26).

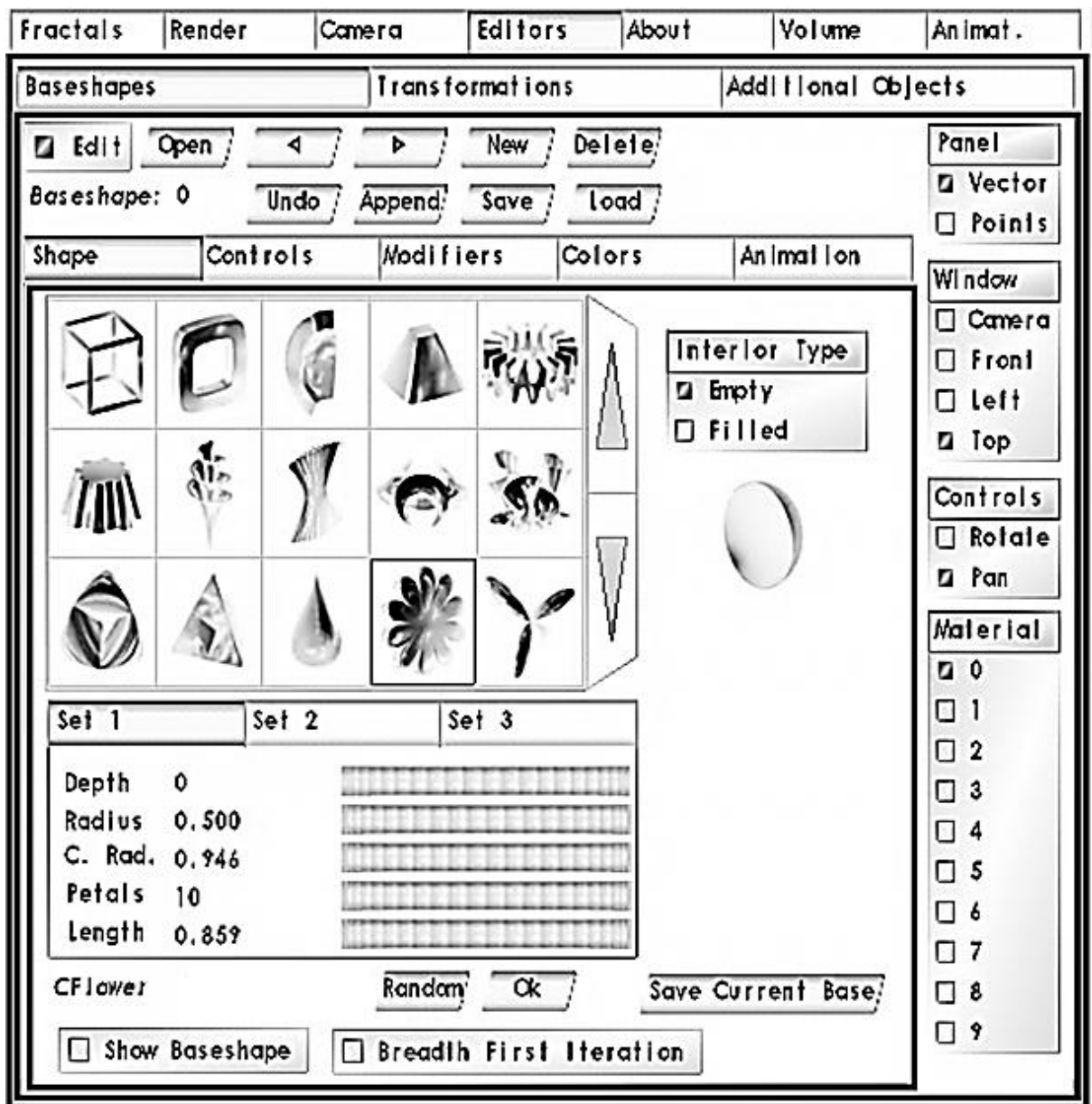


Рис. 2.26. Вкладка Editors

Вкладка складається з двох основних вкладок – Baseshape і Transformations.

Для активації вкладки Baseshape (рис. 2.26) треба натиснути кнопку Edit. У верхній частині цієї вкладки знаходиться група кнопок:

Edit відкриває і закриває редактор базових форм;

Open відкриває всі елементи редактора;

New додає нової форми під фрактал, відкриваючи доступ до поточних параметрів;

Delete видаляє поточну форму із фрактала;

дві кнопки зі стрілками дозволяють переключатися між створеними формами (якщо їх більше від однієї);

Undo відмінює останню дію;

Append, Save, Load дозволяють додавати у програму, зберігати і завантажувати у фрактал інші базові форми у форматі bsf.

Далі йде блок роботи з базовими формами, який включає чотири вкладки – Shape, Controls, Modifiers colors, Anim.

Вкладка Shape включає вікно вибору базових форм.

Вкладка включає елементи управління параметрами тієї чи іншої форми:

Depth – параметр, що визначає кількість фрактальних ітерацій;

Radius – параметр, що визначає фактичний розмір базової форми;

Random задає випадкові параметри поточної форми.

Під вікном базових форм розміщені кнопки, які дозволяють побачити конкретно вибрану форму у вікні попереднього перегляду:

Show Baseshape дозволяє показати або приховати основну форму у центрі фрактала;

Breadth First Iteration встановлює випадкову кількість ітерацій базової форми.

Controls (рис. 2.27) дозволяє переміщувати і обертати фрактал у просторі.

Елементи управління цією вкладкою обмежують переміщення фрактала на окремих осях:

F – повне вільне переміщення по осям;

X – обмеження переміщення по осі X;

Y – обмеження переміщення по осі Y;

Z – обмеження переміщення по осі Z;

0 – переміщення заборонені, базова форма закріплена на початку координат.

Пункт Rotation має таке ж призначення, як і Translation, тільки заважає обертанню базових форм:

++X повертає базову форму на 90 градусів по осі X;

++Y повертає базову форму на 90 градусів по осі Y;

++Z повертає базову форму на 90 градусів по осі Z;

--X повертає базову форму на -90 градусів по осі X;

--Y повертає базову форму на -90 градусів по осі Y;

--Z повертає базову форму на -90 градусів по осі Z.

Rf відбиває фактичну орієнтацію базової форми, внаслідок чого напрями осей змінюються на протилежні.

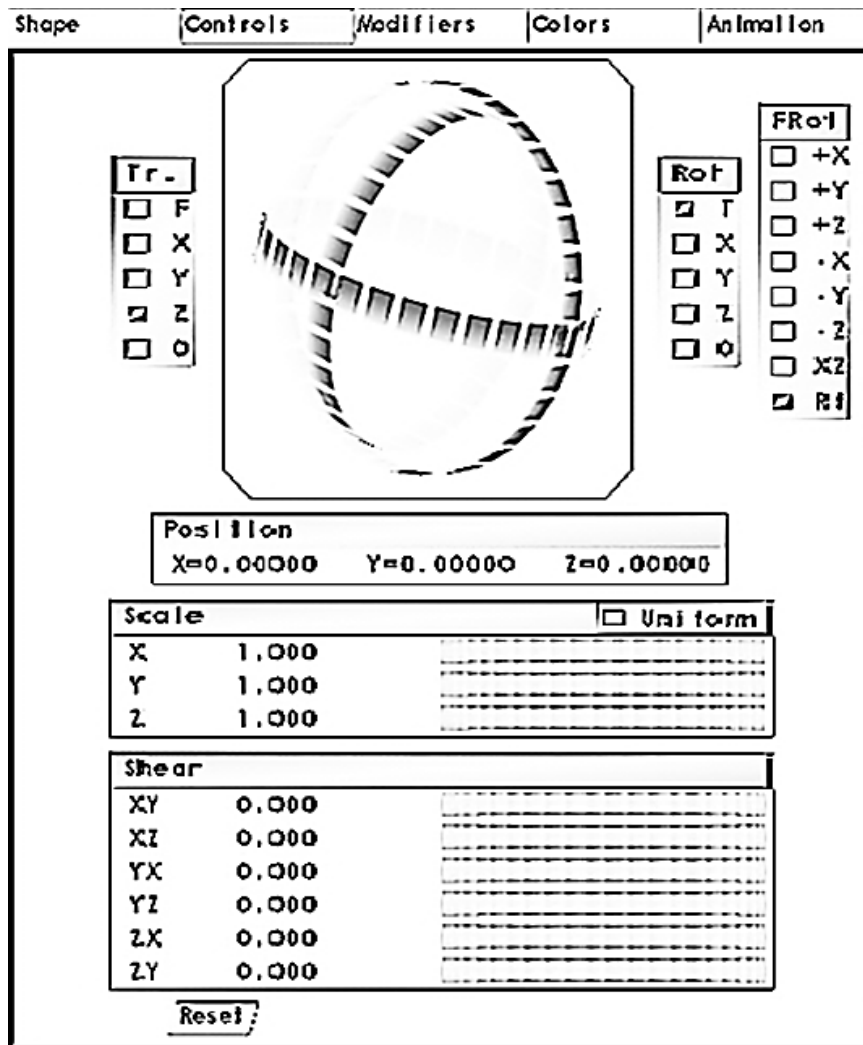


Рис. 2.27. Вікно вкладки Controls

Вкладка Colors відповідає за колір базових форм.

Функції вкладки Transformations ідентичні до тих, що знаходяться у вкладці Controls Baseshapes (рис. 2.28).

На вкладках Baseshapes і Transformations відображається Panel, яка має два режими – Vector (Векторний) і Point (Точковий). Ці два режими впливають на відображення фрактала у вікні попереднього огляду. Векторний режим зручний тим, що в ньому можна легко побачити вибрану форму (вона підсвічується червоним кольором) і легко рухати її мишкою у вікні попереднього огляду. Але він не дозволяє побачити, як виглядатиме фрактал у кінцевому вигляді. Це дозволяє зробити точковий режим. Вибравши цей режим, можна подивитися, що приблизно становитиме майбутній фрактал. Панель Window дозволяє змінювати положення камери, завдяки чому можна подивитися на фрактал із різних сторін. Пункт Controls змінює режим контролю переміщення фрактала Rotate – обертання, Pan – панорамування.

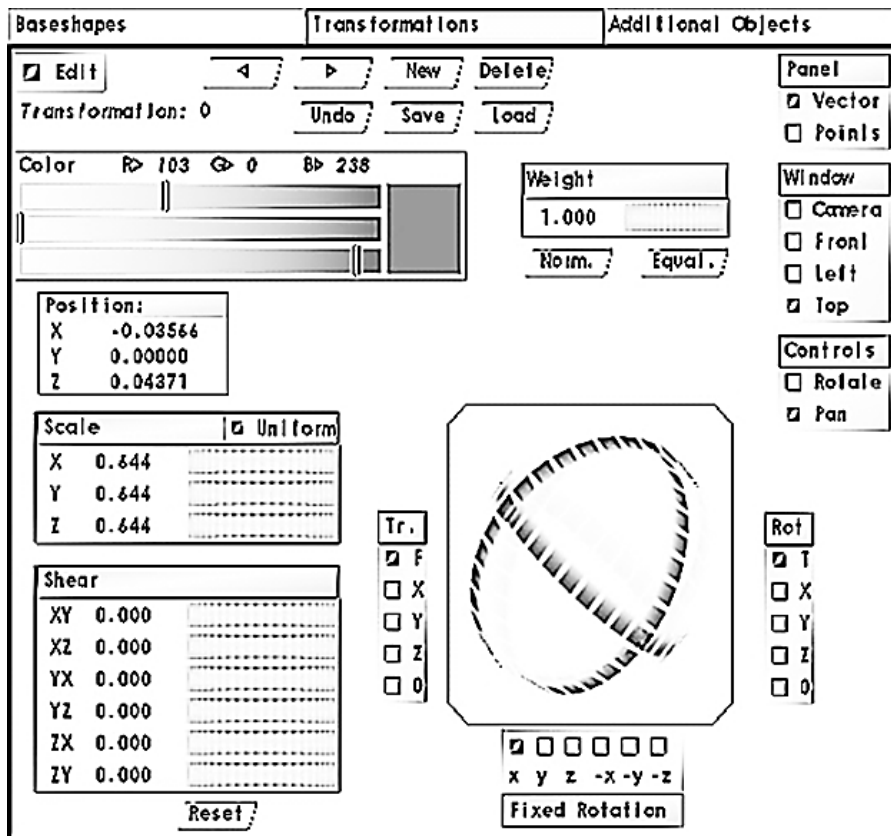


Рис. 2.28. Вікно вкладки Transformations

Вікно попереднього перегляду фрактала дозволяє подивитися, як виглядатиме фрактал (рис. 2.29).

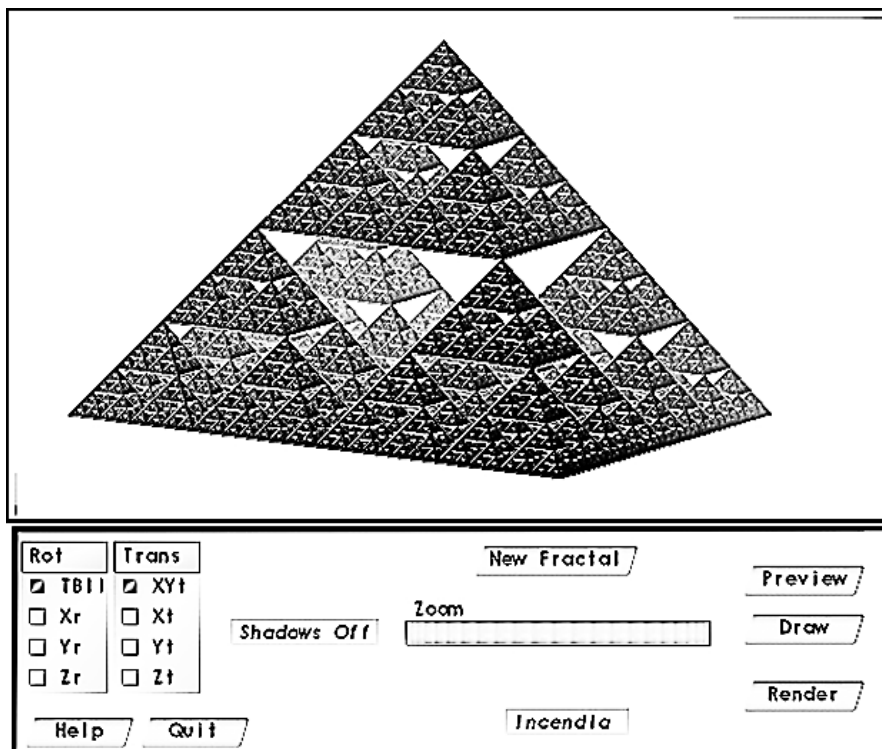


Рис. 2.29. Вікно попереднього перегляду



Кнопка New Fractal дозволяє створити новий фрактал на основі вибраного типу з випадковими параметрами.

Кнопки Preview і Draw використовуються для перегляду фрактала у двох режимах Draw (білі крапки) і швидкого попереднього перегляду.

У режимі Draw можна обертати і пересувати фрактал мишкою. Ліва кнопка миші управляє трекболом обертання, а права кнопка миші управляє переміщенням фрактала. Для зручності у пункті Rot можна обмежити обертання на різних осях, вибравши конкретний напрям:

Tbll – у будь-якому напрямку;

Xr – вгору, вниз;

Yr – управо, вліво;

Zr – за годинниковою стрілкою, проти годинникової стрілки.

Переміщення фрактала правою кнопкою миші також може бути обмежене конкретними напрямками у пункті:

Trans-XYt – у будь-якому напрямку;

Xt – по горизонталі;

Yr – по вертикалі;

Zt – по осі Z.

Повзунок Zoom контролює коефіцієнт масштабування. Масштабування також можна здійснювати коліщатком миші.

*Примітка:* кнопки перегляду зображення працюють тільки за умови, коли відключені параметри Baseshapes і Transformations на вкладці Editors, отже, якщо не вдається подивитися на своє творіння, насамперед перевіряйте і вимикайте саме ці дані.

Кнопка Render запускає процес рендерингу. Якщо зупинити процес рендерингу і знову натиснути кнопку Render, з'явиться попередження, що попередній вміст буфера буде стертий.

## **2.6. Експорт фрактала з Incendia у 3ds max**

Відкриваємо Incendia і вибираємо тип фрактала у вікні роботи з фракталом 1 3D Inverse Tree (рис. 2.24).

Після цього у вікні попереднього перегляду натискаємо кнопку New Fractal кілька разів доти, поки не з'явиться потрібний вид фрактала (рис. 2.29).

Далі у вікні роботи з фракталом (рис. 2.24) вибираємо вкладку Volume. У полі Volumetric Resolution вибираємо воксельне розширення і, відповідно, якість моделі, та швидкість прорахунку. Включаємо рендер (Start Volumetric Render) і після прорахунку натискаємо на кнопку Export Volume as OBJ (рис. 2.30).

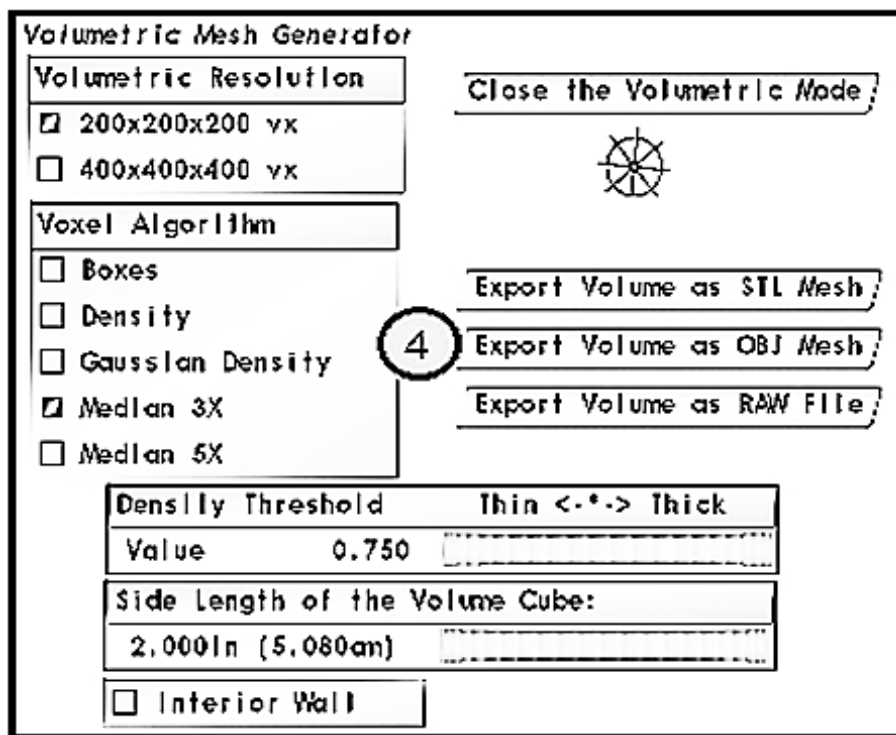


Рис. 2.30. Вкладка Volume вікна роботи з фракталом

3D-модель фрактала створена.

Після запуску 3ds max перетягуємо збережений в Incendia фрактал у його робоче вікно (рис. 2.31).

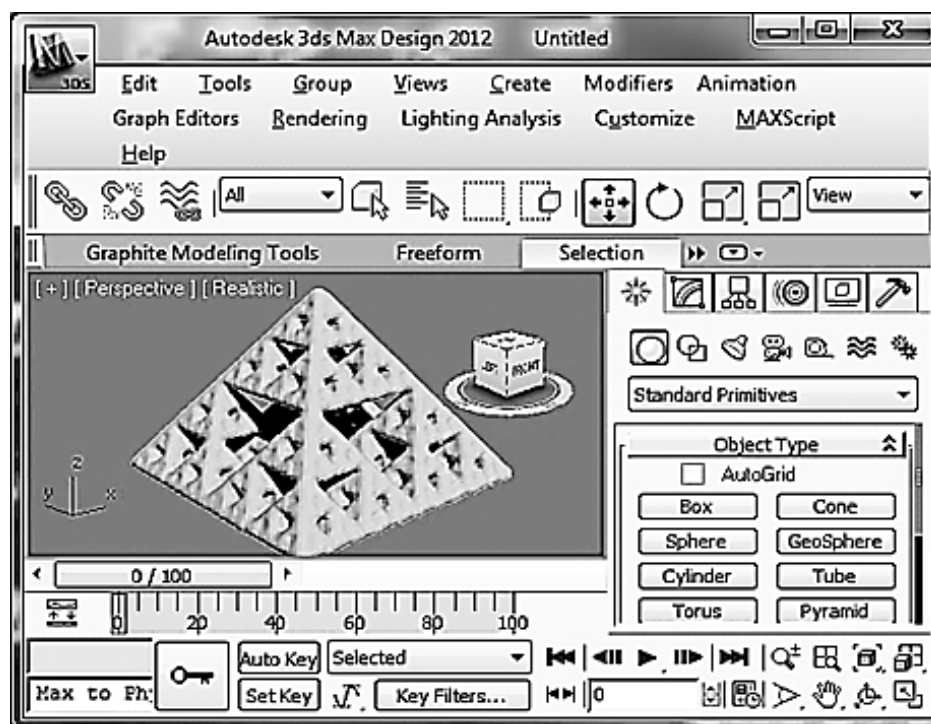


Рис. 2.31. Створений в Incendia фрактал у робочому вікні 3ds max

Тепер клацнемо десь поряд із фракталом і таким чином знімаємо з нього виділення.

У середовищі 3ds max можна модифікувати фрактал і призначити йому матеріал. Основні модифікатори, які можна застосувати до фрактала:

- Bend (вигин);
- Twist (скручування);
- Taper (загострення);
- Stretch (розтягання);
- Mirror (дзеркальна копія);
- Melt (плавлення);
- Push (надування);
- Ripple (брижі).

Застосуємо до піраміди Серпінського модифікатор Twist (рис. 2.32).

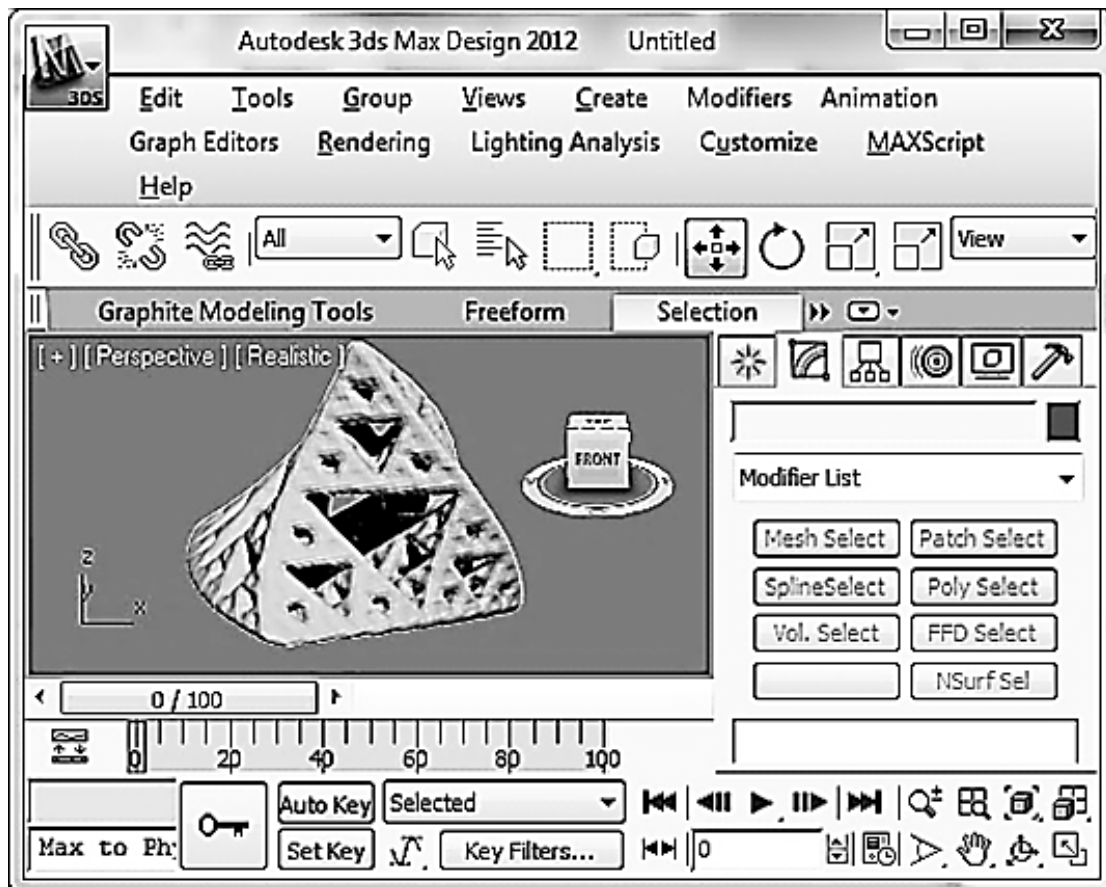


Рис. 2.32. Застосування модифікатора Twist до піраміди Серпінського

Після виконання всіх операцій у 3ds max виконується рендер і збереження 3D-фрактала.

## 2.7. Практичний приклад

Відкриваємо Incendia і вибираємо тип фрактала. Для побудови фантастичного міста найкраще підходить тип фрактала 10. Julia Set  $Z^3+C$  vs. IFS. Цей фрактал має різні види. Щоб отримати вид фрактала (рис. 2.33), треба кілька разів натиснути на кнопку New Fractal.

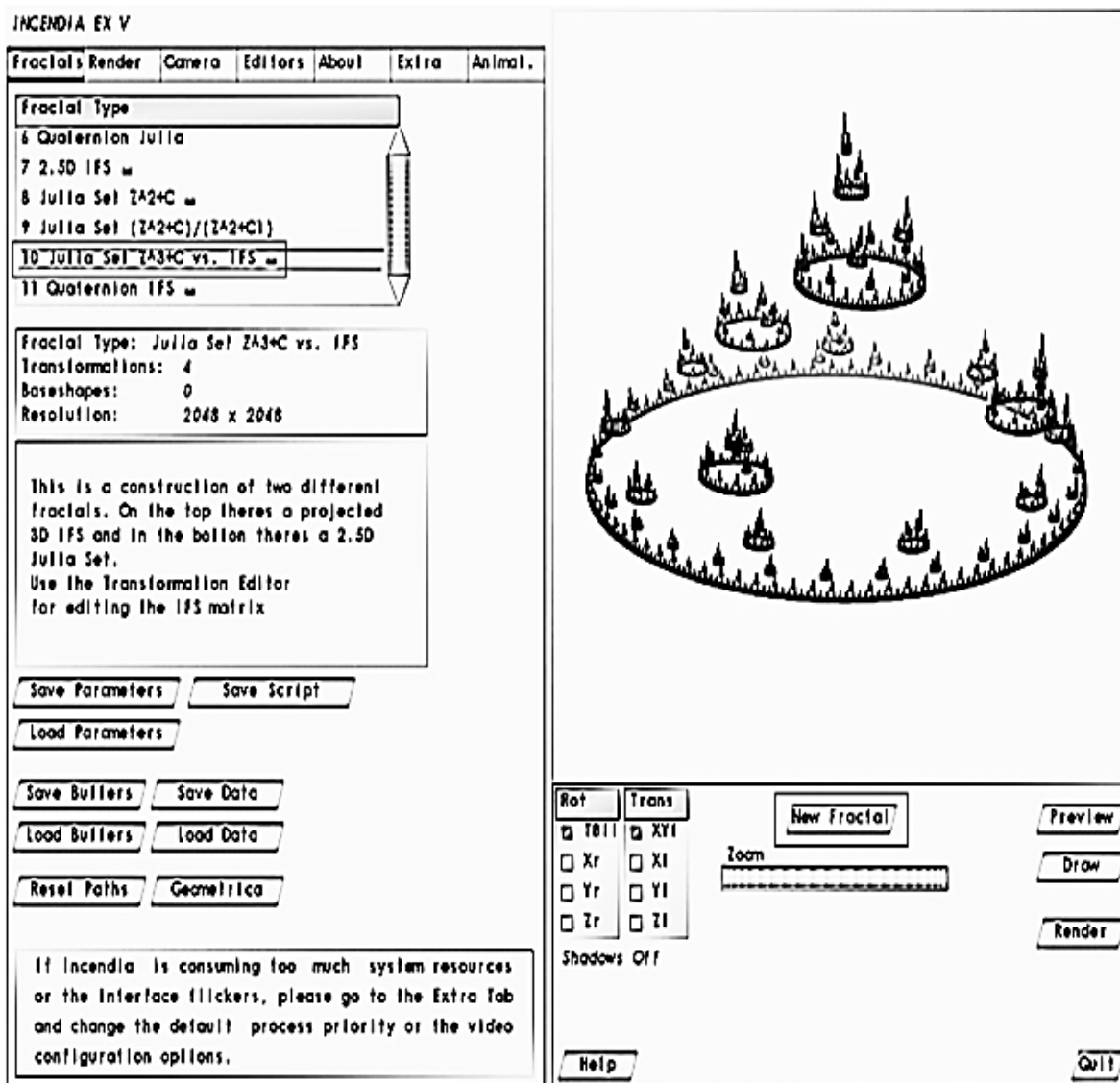


Рис. 2.33. Тип фрактала 10. Julia Set  $Z^3+C$  vs. IFS

Цей фрактал не має базових форм. Для побудови фантастичного міста їх потрібно додати. Для додавання потрібної форми переходимо на вкладку Editors, далі вибираємо вкладку Baseshapes, натискаємо на кнопку Edit і далі натискаємо на кнопку New. На панелі форм Shape, що відкрилася, вибираємо потрібну і задаємо її параметри. У вікні перегляду форма виділена червоним кольором. Її за допомогою мишки можна переміщувати.

На рис. 2.34 показана перша вибрана форма, яка імітуватиме рослинність. Задамо для неї параметри і текстуру.

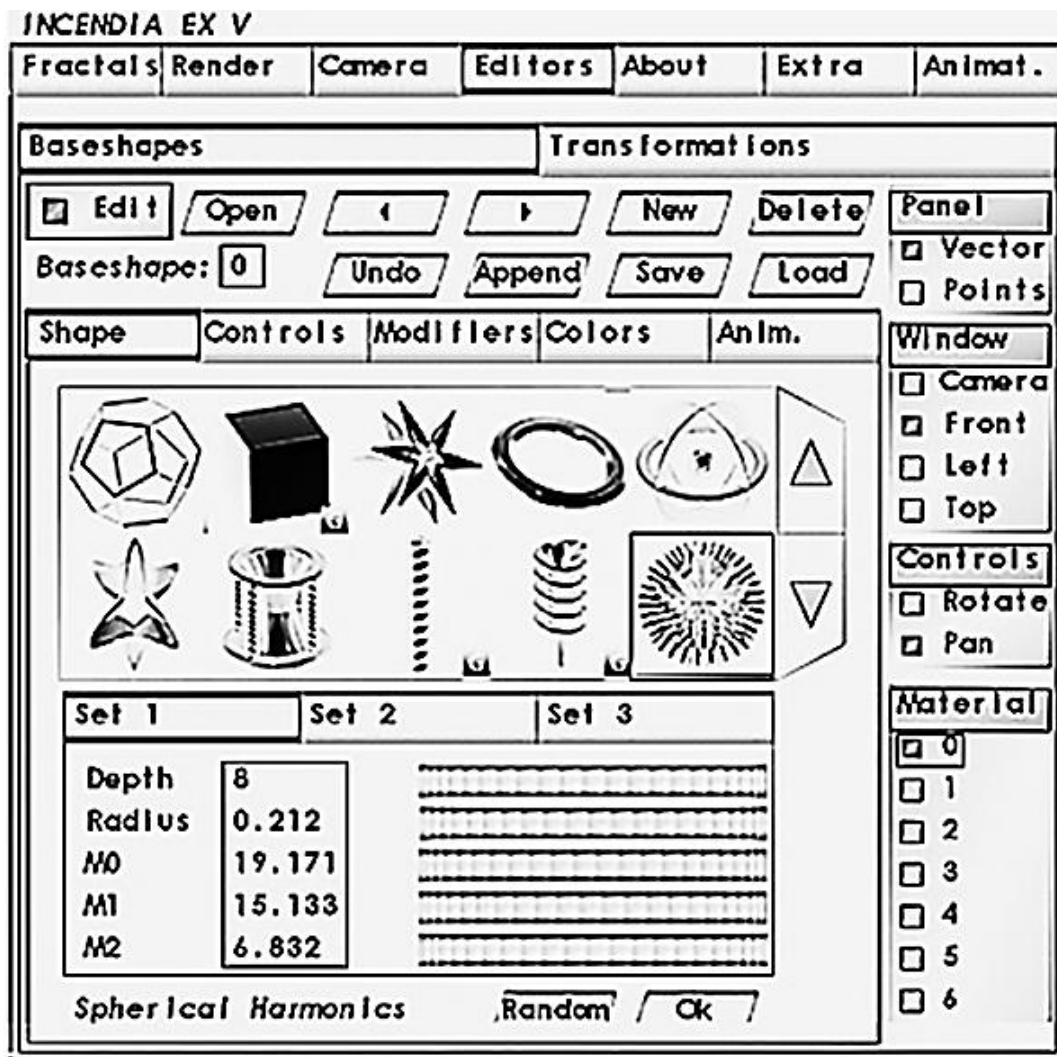


Рис. 2.34. Вибір форми із заданими параметрами для рослинності

Текстуру задають на вкладці Colors, активізують елемент управління Use Surface Texture і натискають на кнопку Load Texture. Далі вибирають потрібну текстуру (рис. 2. 35).

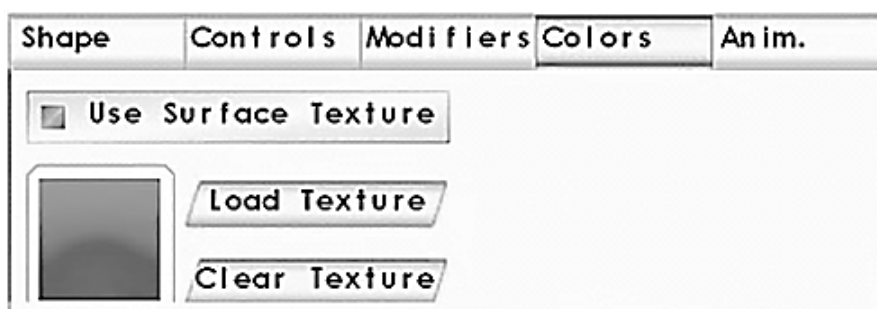


Рис. 2.35. Вікно вибору текстури для форми



Результат вибору текстури можна побачити у вікні перегляду, натиснувши кнопку Points (рис. 2.36).

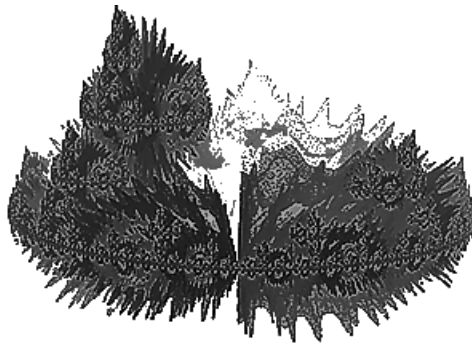


Рис. 2.36. Результат вибору текстури

Для створення стіни міста додамо форму із заданими параметрами (рис. 2.37) і вкажемо для неї матеріал 1.

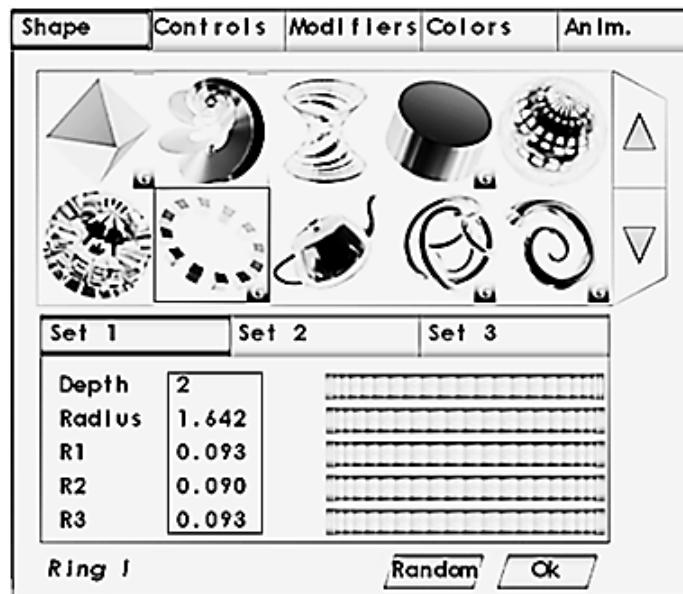


Рис. 2.37. Вибір форми із заданими параметрами для стін міста

Задаємо текстуру стіни у вигляді цегляної кладки.

Для побудови замку скористаємося двома формами із заданими параметрами (рис. 2.38).

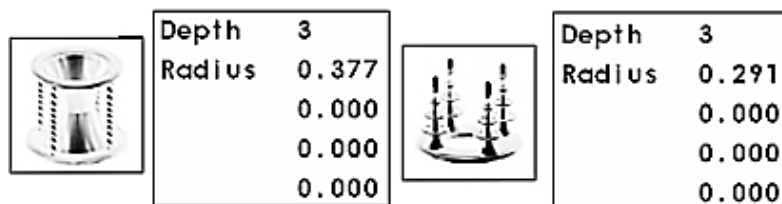


Рис. 2.38. Форми для замку із заданими параметрами

Перемикаємося на вкладку Controls і, якщо необхідно, обертаємо форму навколо осі X, щоб колони було краще видно. Текстуру для цієї форми задати не можна, але матеріал вказуємо 1.

Далі додаємо вежу (рис. 2.39). Вказуємо для неї матеріал 2. Кількість ітерацій у неї Depth = 0, оскільки вежа в нас буде одна. Вежа вузька і висока. Тому Radius зменшуємо, а висоту Height збільшуємо (можна задати будь-яке значення).

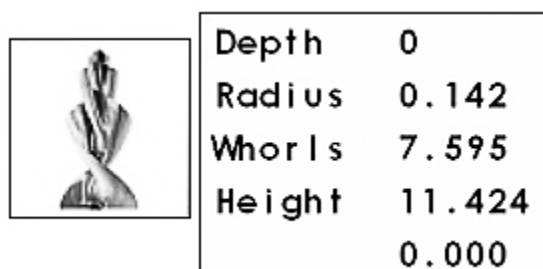


Рис. 2.39. Форма для вежі із заданими параметрами

Задаємо текстуру вежі.

Для додавання птахів використовується форма металевої зірочки із заданими параметрами (рис. 2.40). Щоб зірочка розлетілася, її потрібно поставити досить далеко від центра обертання фрактала.

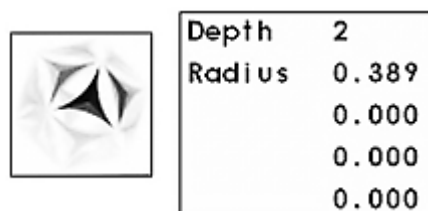


Рис. 2.40. Форма для птахів із заданими параметрами

А можна переключити панелі у режим Points і мишкою розтаскати частини зірочки так, як потрібно.

Спорудження фантастичного міста закінчене (рис. 2.41). У нього чотири трансформації і шість базових форм.

У процесі створення фантастичного міста використовувалися форми, яким присвоювався номер матеріалу. Для вибору матеріалу переходимо на вкладку Render – Materials. Далі вибираємо шаблони матеріалу і дифузії, а також їхні параметри (рис. 2.42).

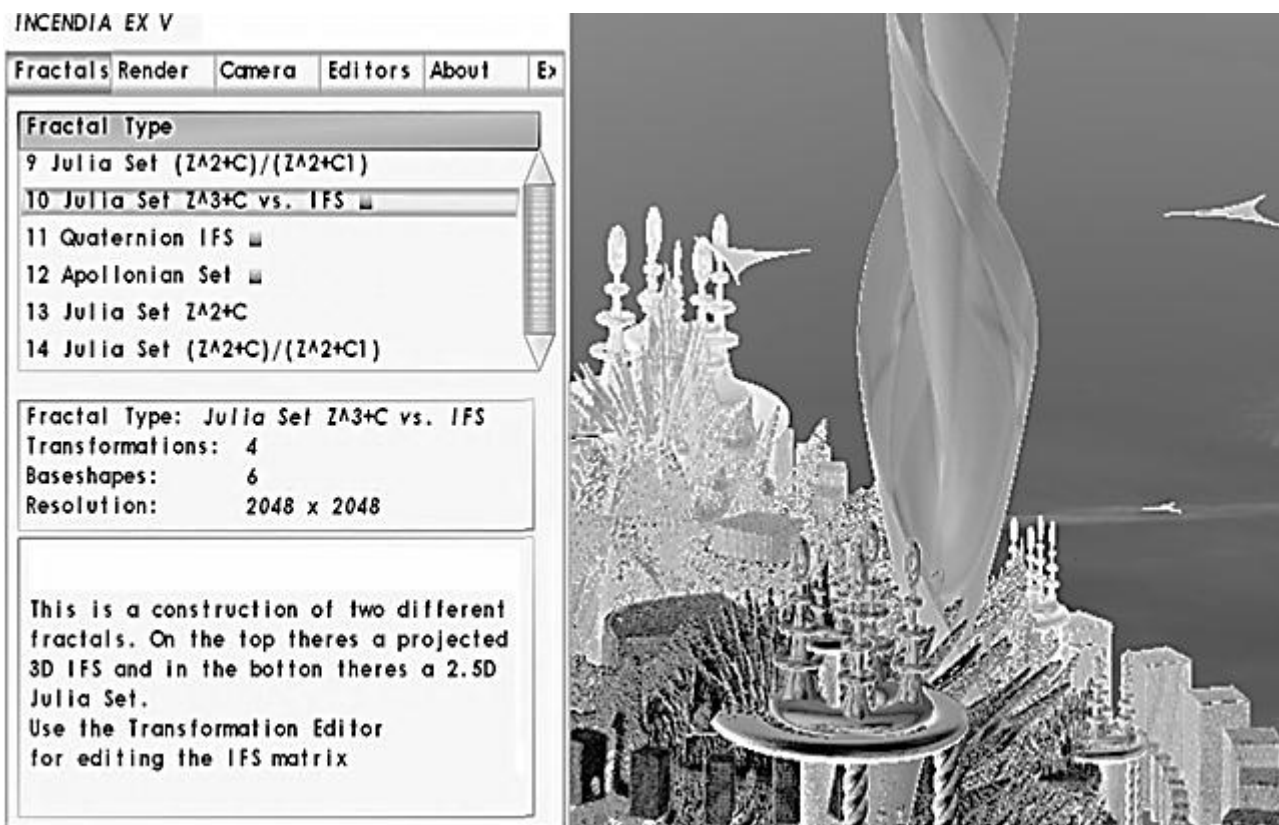


Рис. 2.41. Підсумкові характеристики фрактала фантастичного міста

	Ambient 0.681		Ambient 0.766		Ambient 1.359		Ambient 0.664
	Diffuse 0.754		Diffuse 1.081		Diffuse 0.729		Diffuse 0.391
	Specular 90.00						

Рис. 2.42. Шаблони матеріалів, дифузій і їхні параметри

Для надання природного вигляду фрактальному місту необхідно налаштувати зовнішнє оточення. Для цього треба перейти на вкладку Environment і виконати налаштування:

ввімкнути туман – Enable Fog;

ввімкнути об'ємний туман – Volumetric Fog;

ввімкнути фон – Background;

у якості фону використовують природне освітлення (схід або захід сонця) – Skylight;

активізувати дозвіл використовувати фон як джерело світла для туману – Use Background as fog Color Source;



встановити видимість у тумані – Fog P. A (чим більше значення, тим менший туман і тим далі ми бачимо);

встановити кількість пилу в повітрі – Sun N (чим вона більша, тим більш червоним буде схід або захід);

встановити колір туману – Vol fog Red, Vol fog Green, Vol fog Blue.

Результати налаштування показані на рис. 2.43.

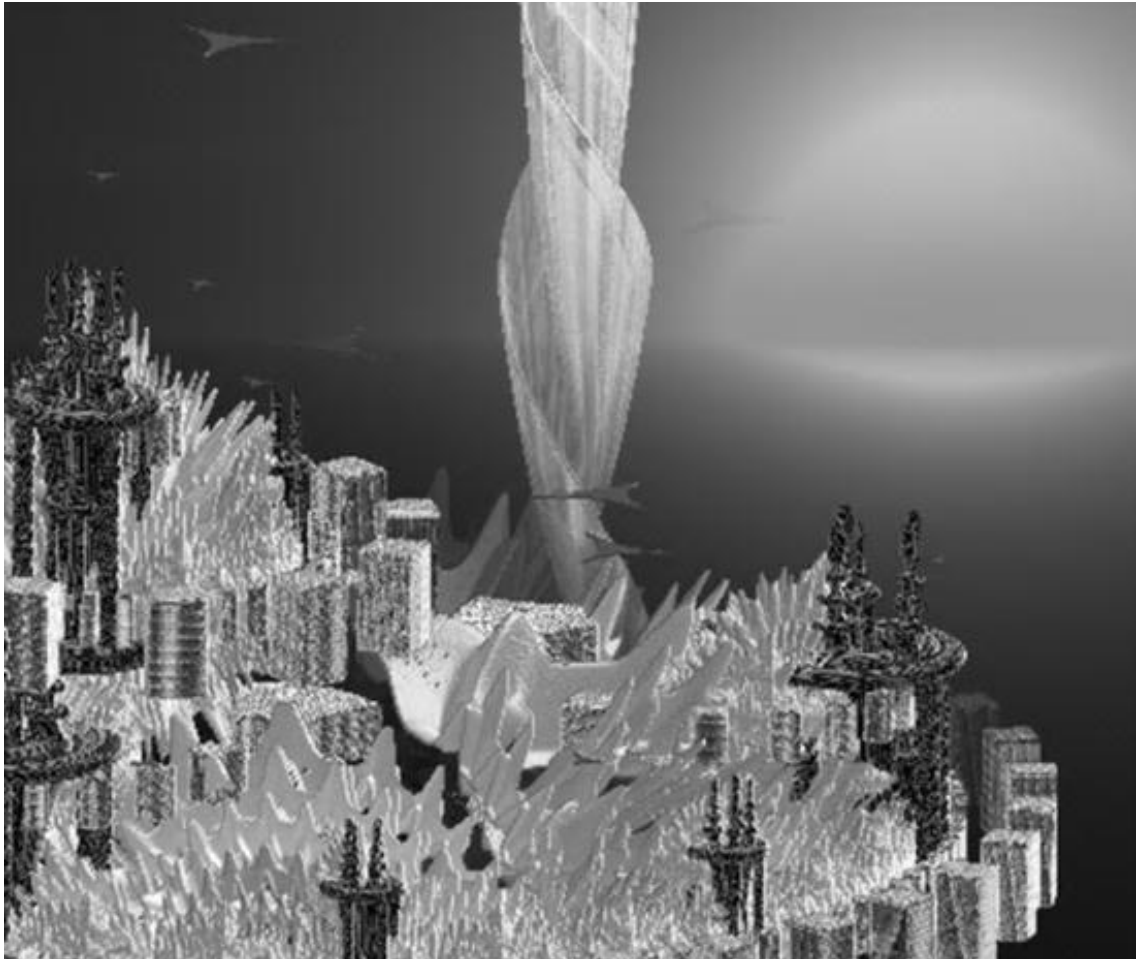


Рис. 2.43. **Результати налаштування зовнішнього оточення**

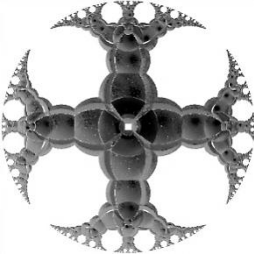
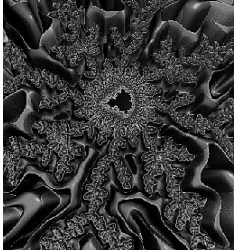


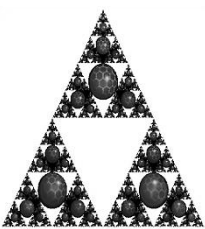
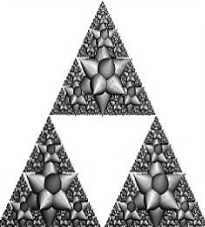
Для створення ефекту сходу сонця над горизонтом переходять на вкладку Camera і у вікні Light Direction мишкою тягнуть сонечко вниз, поки воно не зайде за горизонт, і продовжують тягнути вниз, поки воно не почне підніматися. Далі вмикають тіні Shadow On.

Після всіх налаштувань вмикають Render.

## **2.8. Завдання для самостійної роботи**

Використовуючи програми Incendia і 3ds max, згенеруйте 3D-фрактали, наведені в табл. 2.2.

## Фрактали

№	Вид фрактала	Ім'я фрактала	Операція модифікації у 3ds max
1	2	3	4
1		Cross.zdat	Ripple (брижі)
2		MandelbrotII.zdat	Melt (плавлення)
3		Mandelbrot.zdat	Push (надуття)
4		Metallic Spiral.zdat	Ripple (брижі)
5		Simple Triangle II.zdat	Stretch (розтягання)
6		Simple Triangle III.zdat	Bend (вигин)

1	2	3	4
7		VolumeExample.zdat	Mirror (дзеркальна копія)
8		Hexahedral.zdat	Melt (плавлення)

**Контрольні запитання**

1. Поняття фрактала.
2. Історія фракталів.
3. Класифікація фракталів.
4. 3D-фрактали.
5. Математичний опис фракталів.
6. Програми створення та редагування фракталів.
7. Редактор 3D-фракталів Mandelbulb3D.
8. Генератор тривимірних фракталів Incendia.
9. Інтерфейс Incendia.
10. Фрактали у 3ds max.