

УДК 515.2

ЗАСТОСУВАННЯ R-ФУНКЦІЙ ДЛЯ ПОБУДОВИ ГЕОМЕТРИЧНИХ ОРНАМЕНТІВ З ЕЛЕМЕНТАМИ СИМЕТРІЇ

Челомбітько В.Ф.

к.т.н., доцент, кафедра Медіасистем та технологій,
Харківський національний університет радіоелектроніки

Анотація. Наведено спосіб практичного застосування R-функцій до складання у неявному вигляді рівнянь зображень з елементами симетрії у середовищі математичного пакету Maple.

Ключові слова: МЕТОД R-ФУНКЦІЙ, ВІЗЕРУНОК, ГЕОМЕТРИЧНІ ОРНАМЕНТИ, РЕКУРСИВНІ ПОБУДОВИ, MAPLE.

Вступ

Орнаменти є надбанням світової культури, адже вони відображають особливості характеру цілих народів. Орнамент – це візерунок, що складається з ритмічно впорядкованих елементів. Орнамент будується за особливими законами, за допомогою певних засобів. Головний засіб – це геометричне перетворення. Орнаменти бувають спокійні і виважені, але бувають імпульсивні і різкі для сприйняття [1]. Для України орнаменти займають особливе місце, адже вони визначають прояви знакових систем трипільської культури, і є відомими ще з стародавніх часів [2-4]. Тому актуальними будуть дослідження, спрямовані на формалізацію побудови засобами обчислювальної техніки орнаментів з елементами симетрії.

У процесі історичного та культурного розвитку у місцевостях України утворились характерні орнаментальні мотиви і композиції, найбільш улюблена і поширена колірна гама, специфічні техніки виконання. Майстри розвивали і вдосконалювали кращі досягнення своїх попередників. Геометричні орнаменти складались з різноманітних геометричних фігур: трикутників, ромбів, квадратів, кружечків, тощо. У назвах орнаментальних мотивах вражає образна спостережливість, тонке поетичне почуття. Це «барвінок», «хмелик», «курячий брід», «гарбузове листя», «зозулька», тощо [1]. Здавна елементи орнаменту символізували певні явища природи та природні істоти. Наприклад, «вужики» або «кривульки» - то знаки води, тобто життя. А чотирикутники, кружечки - то символ Сонця [2].

Існує значна кількість математичних способів опису та побудови орнаментів засобами обчислювальної техніки. В даній роботі перевагу віддано теорії R-функцій. В роботі [5] розкрито ідеї теорії R-функцій, надано приклади її деяких впроваджень. Застосування методу R-функцій до побудови рівнянь зображень, що мають симетрію, наведено в роботі [6]. В роботі [7] розглянуто

побудову симетричних функцій для симетричних зображень. Але відкритим залишається питання раціонального застосування R-функцій до побудови рівнянь зображень з елементами симетрії.

Мета та задачі дослідження

Навести способи застосування R-функцій до складання у неявному вигляді рівнянь зображень орнаментів з елементами симетрії відносно точки, відносно прямої та симетрії повороту.

Розглянути способи практичного застосування R-функцій до складання у неявному вигляді рівнянь зображень з елементами симетрії у середовищі пакету Maple.

Розробити алгоритми складання та побудови орнаментів на площині у середовищі алгоритмічної мови Maple, які базуються на головних принципах створення візерунків.

Основна частина

Спочатку наведемо деякі визначення.

На площині Oxy опорною областю A функції $f(x,y)$ називається множина точок, де функція $f(x,y)$ приймає невід'ємні значення [8].

Опорна область визначає геометричну форму множини точок на площині, тому з окремими опорними областями можна виконувати логічні операції об'єднання, перетину та доповнення. Для цього у теорії R-функцій існують аналоги логічних операцій у вигляді R-операцій.

R-диз'юнкція опорних областей $A1$ і $A2$ забезпечується операцією над функціями f_1 і f_2 , що дозволяє одержати таку функцію, опорна область якої є об'єднанням областей $A1$ і $A2$. R-диз'юнкція позначається

$$f_3(x, y) = f_1(x, y) \vee_R f_2(x, y) \quad \text{або} \quad A3 = A1 \vee_R A2.$$

R-кон'юнкція опорних областей $A1$ і $A2$ забезпечується операцією над функціями f_1 і f_2 , що дозволяє одержати таку функцію, опорна область якої є перетином областей $A1$ і $A2$. R-кон'юнкція позначається

$$f_4(x, y) = f_1(x, y) \wedge_R f_2(x, y) \quad \text{або} \quad A4 = A1 \wedge_R A2.$$

R-заперечення опорної області $A1$ – це операція над функцією f_1 , що дозволяє одержати таку функцію, опорна область якої є доповненням $A1$ до всієї площини Oxy . R-заперечення позначається

$$f_5(x, y) = \overline{f_1(x, y)}, \quad \text{або} \quad A5 = \overline{A1}.$$

Аналітично це зводиться до зміни знаків у виразі $A1$: $f_1(x,y) \geq 0$. Іншими словами, за бажанням можемо обрати одну з двох опорних областей, що доповнюють одна одну до площини Oxy .

На рис. 1 надано ілюстрації зазначених R-операцій. Найпростішими аналітичними реалізаціями цих R-операцій відображені у таких формулах.

R-диз'юнкція: $f_1 \vee_R f_2 = \max(f_1, f_2) = 0,5 \{f_1 + f_2 + |f_1 - f_2|\}$;

R-кон'юнкція: $f_1 \wedge_R f_2 = \min(f_1, f_2) = 0,5 \{f_1 + f_2 - |f_1 - f_2|\}$.

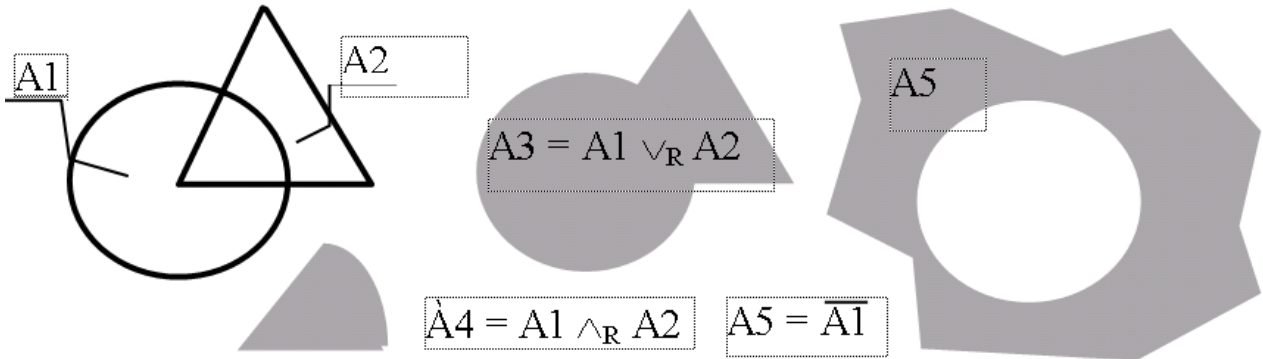


Рисунок 1 – Ілюстрації до R-диз'юнкції, R-кон'юнкції та R-заперечення

Визначення опорних областей складних фігур розберемо на прикладах.

Приклад 1. Визначити опорну область кута між прямими k і m, сторони якого перетинають вісь x.

Запишемо рівняння прямих:

k: $x/4 + y/5 = 1$;

m: $x - y = 1$,

а потім перепишемо їх як нерівності опорних областей:

A1: $1 - x/4 - y/5 \geq 0$

і A2: $x - y - 1 \geq 0$,

та перевіримо підстановкою координат точки 0,0.

Обидві нерівності записано вірно, бо у першому випадку результат додатній, а у другому – від'ємний. Тепер над опорними областями A1 та A2 виконаємо операцію R-кон'юнкції, тобто одержати нову опорну область $F = A1 \wedge_R A2$ (рис. 2).

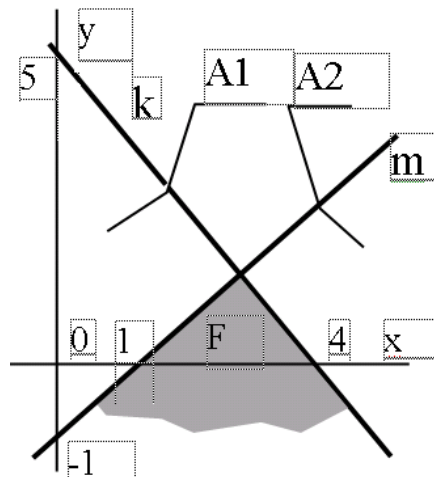


Рисунок 2 – До визначення опорної області кута між прямими

Приклад 2. Визначити опорну область фігури, що нагадує форму «півмісяця».

Цю форму утворюють два кола з опорними областями (рис. 3), що описуються нерівностями $A1: 9 - x^2 - y^2 \geq 0$ та $A2: (x - 3)^2 + (y - 3)^2 - 9 \geq 0$, і над якими теж треба виконати операцію R-кон'юнкції $F2 = A1 \wedge_R A2$.

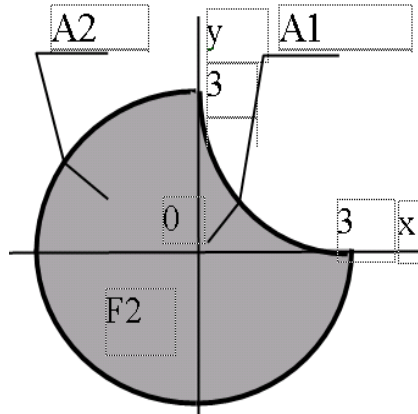


Рисунок 3 – До визначення опорної області «півмісяця»

Зменшення кількості опорних областей спрощує логічні формули. Найчастіше це можна зробити за рахунок використання симетрії зображень, тобто раціонально розташовувати системи координат.

Наприклад, дві і навіть чотири опорні області можна описати одним рівнянням, якщо є симетрія відносно:

– осі x - $A: f(x, |y|) \geq 0$. Наприклад, $A: 2 - |y| \geq 0$ утворює горизонтальну смугу шириною 4 (рис. 4);

– осі y - $A: f(|x|, y) \geq 0$. Наприклад, $A: 2 - |x| \geq 0$ утворює вертикальну смугу шириною 4 (рис. 5);

– двох осей $A: f(|x|, |y|) \geq 0$. Наприклад, $1 - |x|/2 - |y|/3 \geq 0$ утворює ромб із діагоналями 4 і 6 (рис. 6).

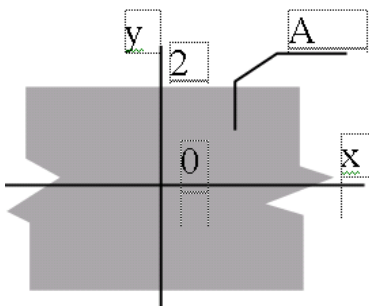


Рисунок 4

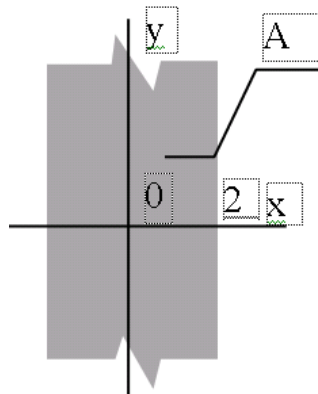


Рисунок 5

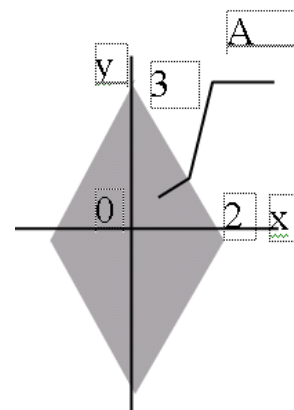


Рисунок 6

У деяких випадках для опису окремих елементів зображень необхідно використовувати додаткові опорні області. Найчастіше це роблять, коли описують різні спряження чи частину зображення треба обмежити з якогось боку.

Приклад 3. Для побудови спряження кута, утвореного опорними областями $A1$ та $A2$, спочатку за допомогою додаткової області $A3$ треба сформувати зрізаний кут, а потім до нього приєднати опорну область кола $A4$ (рис. 7). Логічна формула такого зображення буде мати такий вигляд

$$B = (A1 \wedge_R A2 \wedge_R A3) \vee_R A4.$$

Приклад 4. Для зображення на рис. 8 горизонтальну смугу спочатку слід обмежити з боку круга, а потім необхідно об'єднати з ним. З урахуванням симетрії це обмеження зроблено з використанням опорних областей $A1$ і $A2$. Остаточно одержуємо

$$B = (A1 \wedge_R A2) \vee_R A3.$$

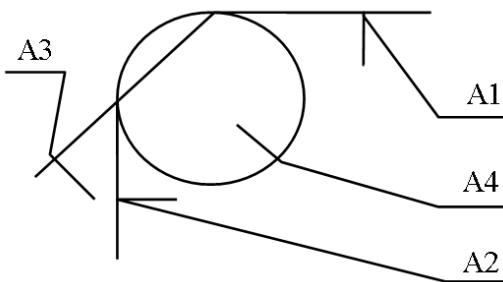


Рисунок 7

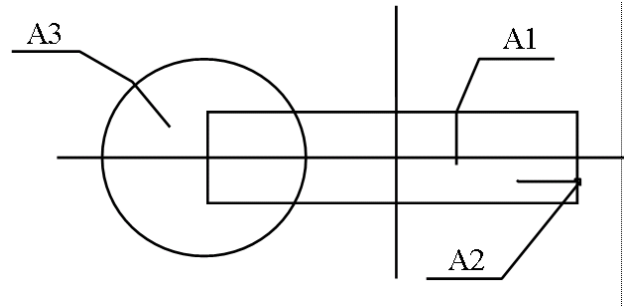


Рисунок 8

Для програмної реалізації побудови орнаментів необхідно задати R -операції у вигляді (тут і далі збережено синтаксис мови Maple):

$$\begin{aligned} o &:= (a, b) \rightarrow (a+b+abs(a-b))/2: \\ p &:= (a, b) \rightarrow (a+b-abs(a-b))/2: \end{aligned}$$

Задамо першу і другу осі симетрії у вигляді

$$\begin{aligned} q &:= 2: \quad \alpha := \text{Pi}/3.: \\ \text{fosi1} &:= x*\cos(\alpha) + y*\sin(\alpha) - q: \\ q &:= 2: \quad \alpha := \text{Pi} - \text{Pi}/3.: \\ \text{fosi2} &:= x*\cos(\alpha) + y*\sin(\alpha) - q: \end{aligned}$$

Модульний примітив (рис. 9) задамо у вигляді

$$\begin{aligned} \text{f1} &:= (x, y) \rightarrow 4 - abs(x) - abs(y): \\ \text{f2} &:= (x, y) \rightarrow -5 + (x)^2 + (y)^2: \\ F &:= (x, y) \rightarrow p(\text{f1}(x, y), \text{f2}(x, y)); \end{aligned}$$

Для здійснення побудови симетричного модульного примітиву відносно першої осі симетрії слід застосувати формули:

$$\begin{aligned} X1 &:= 2*q*\cos(\alpha) - y*\sin(2*\alpha) - x*\cos(2*\alpha); \\ XI &:= 2 - \frac{y\sqrt{3}}{2} + \frac{x}{2} \\ Y1 &:= 2*q*\sin(\alpha) - x*\sin(2*\alpha) + y*\cos(2*\alpha); \end{aligned}$$

$$Y1 := 2\sqrt{3} - \frac{x\sqrt{3}}{2} - \frac{y}{2}$$

$$F1 := (x, y) \rightarrow F(X1, Y1);$$

На рис. 10 наведено сумісне зображення обраного модульного примітива з побудованим симетричним відносно першої осі зображенням примітива.

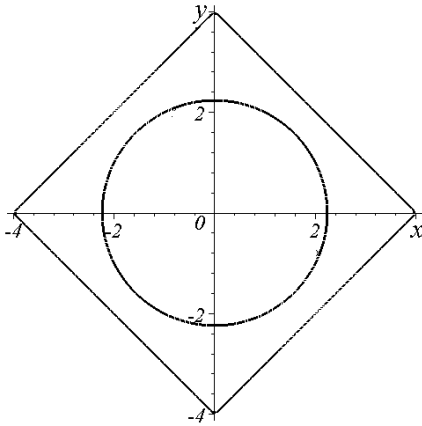


Рисунок 9 – Приклад модульного примітива

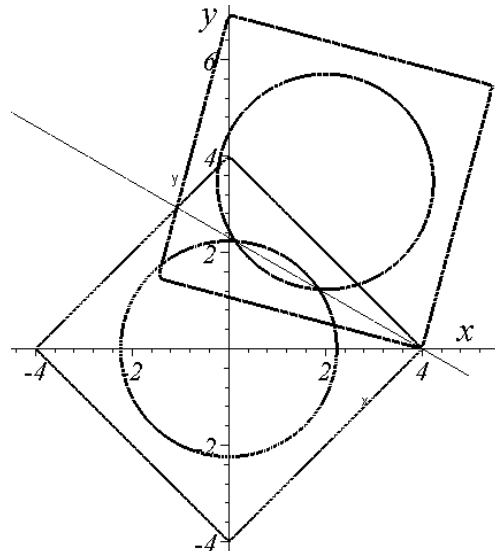


Рисунок 10 – Об'єднання модульного примітива з симетрією від осі № 1

Для здійснення побудови симетричного модульного примітиву відносно другої осі симетрії слід застосувати формули:

$$X2 := 2*q*\cos(\alpha) - y*\sin(2*\alpha) - x*\cos(2*\alpha);$$

$$X2 := \frac{x}{2} + \frac{y\sqrt{3}}{2} - 2$$

$$Y2 := 2*q*\sin(\alpha) - x*\sin(2*\alpha) + y*\cos(2*\alpha);$$

$$Y2 := 2\sqrt{3} + \frac{x\sqrt{3}}{2} - \frac{y}{2}$$

$$F2 := (x, y) \rightarrow F(X2, Y2);$$

На рис. 11 наведено сумісне зображення даного модульного примітива з побудованим симетричним відносно обох осей зображеннями.

До незаперечних переваг застосування R-функцій є можливість оперувати з опорними областями примітивів. Наприклад, якщо використати послідовність R-функцій у вигляді

$$FF := (x, y) \rightarrow p(p(F1(x, y), F2(x, y)), -F(x, y)),$$

то одержимо зображення, яке нагадує символ «страхіття» зі стародавніх орнаментів (рис. 12).

Крім того, за допомогою R-функцій можна будувати графік функції $FF(x, y)$ (рис. 13), а також лінії рівня функції $FF(x, y)$ (рис. 14), що може доповнювати виразні можливості при створенні орнаментів.

Для програмної реалізації побудови симетрій відносно точки необхідно задати R-диз'юнкцію та R-кон'юнкцію, відповідно, у вигляді синтаксису мови Maple [9]:

```
o := (a,b) -> (a+b+abs(a-b))/2:
p := (a,b) -> (a+b-abs(a-b))/2:
```

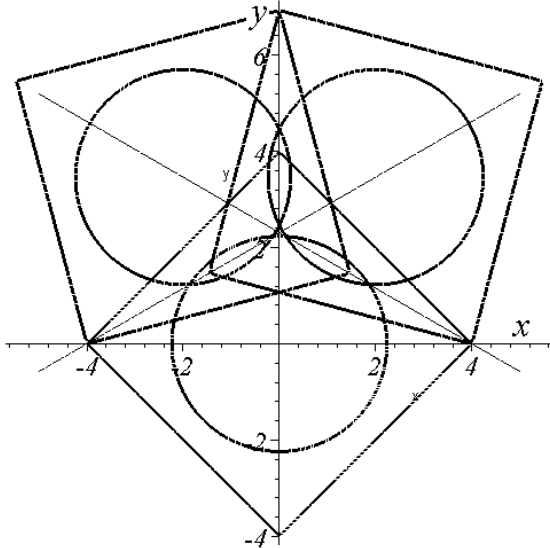


Рисунок 11 – Об'єднання модульного примітива з відбиттям від обох осей

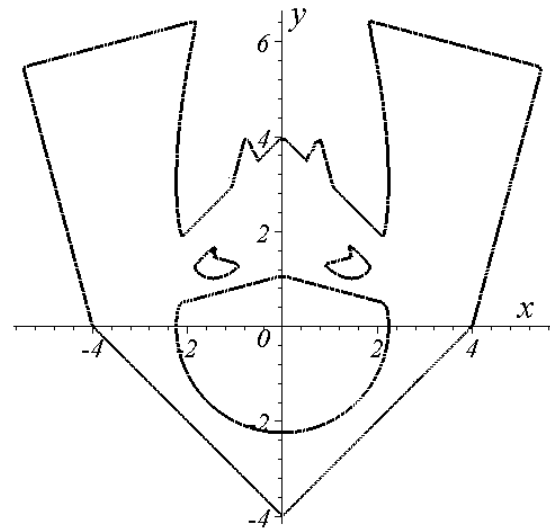


Рисунок 12 – Зображення, яке одержано за допомогою R-функцій

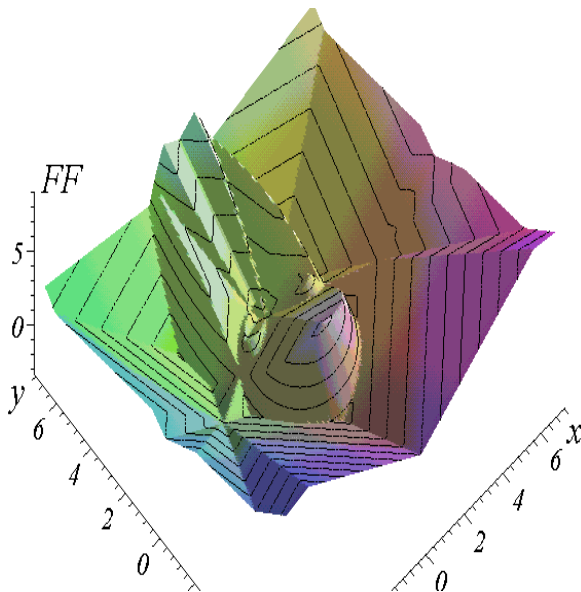


Рисунок 13 – Графік функції $FF(x,y)$

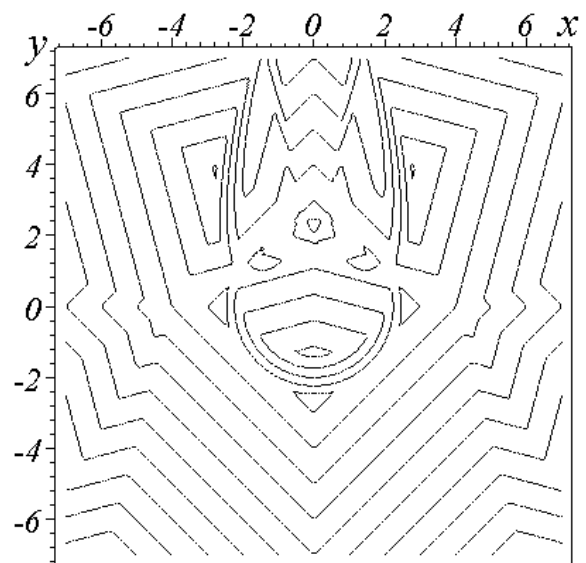


Рисунок 14 – Лінії рівня функції $FF(x,y)$

Модульний примітив (як приклад, зображений на рис. 15) задамо у вигляді

```
f1 := (x,y) -> 3 - abs(x) - abs(y):
f2 := (x,y) -> -0.8 + x^2 + y^2:
f3 := (x,y) -> -y:
f4 := (x,y) -> o(f2(x,y), f3(x,y)):
F := (x,y) -> p(f1(x,y), f4(x,y));
```

Розглянемо побудову симетрій відносно точки. Задамо першу і другу точки симетрії координатами

$$\begin{aligned}x01 &:= 0.5: & y01 &:= 1: \\x02 &:= -0.5: & y02 &:= 1:\end{aligned}$$

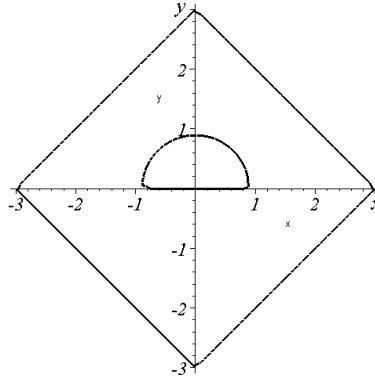


Рисунок 15 – Приклад модульного примітива

Для побудови симетричного модульного примітива відносно першої точки симетрії слід застосувати формули:

$$\begin{aligned}X1 &:= 2*x01 - x; \\Y1 &:= 2*y01 - y; \\F1 &:= (x, y) \rightarrow F(X1, Y1); \end{aligned}$$

На рис. 16 наведено сумісне зображення обраного модульного примітива з побудованим симетричним відносно першої точки зображенням примітива.

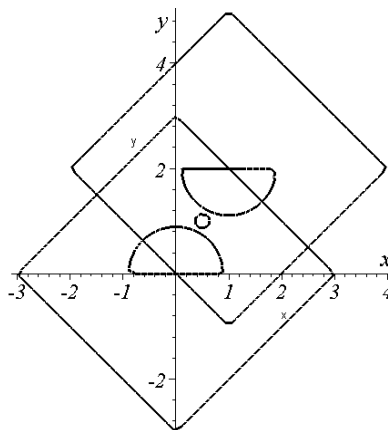


Рисунок 16 – Об'єднання модульного примітива з симетричним від точки

Для здійснення побудови симетричного модульного примітива відносно другої точки симетрії слід застосувати формули:

$$\begin{aligned}X2 &:= 2*x02 - x; \\Y2 &:= 2*y02 - y; \\F2 &:= (x, y) \rightarrow F(X2, Y2); \end{aligned}$$

На рис. 17 наведено сумісне зображення модульного примітива з побудованим симетричним відносно обох точок зображеннями.

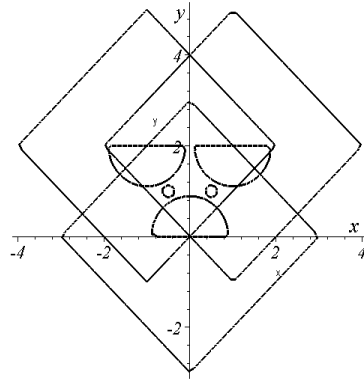


Рисунок 17 – Модульні примітиви із подвійною симетрією

До переваг застосування R-функцій є можливість оперувати з опорними областями примітивів. Якщо використати послідовність R-функцій у вигляді

$$FF := (x, y) \rightarrow p(o(F1(x, y), F2(x, y)), F(x, y)), \quad (1)$$

то одержимо зображення елемента орнаменту (рис. 18).

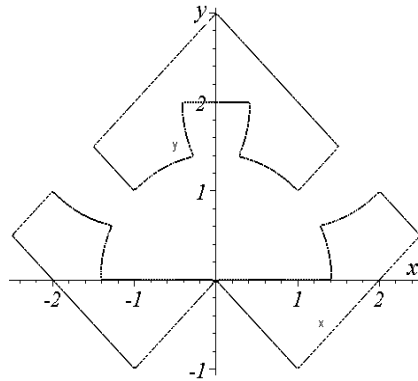


Рисунок 18 – Зображення, яке одержано за допомогою R-функцій (1)

На рис. 19 і 20 зображені елементи орнаменту, побудовані за допомогою послідовностей R-функцій, відповідно, у вигляді

$$FF := (x, y) \rightarrow o(p(F1(x, y), F2(x, y)), F(x, y)); \quad (2)$$

$$FF := (x, y) \rightarrow o(p(F1(x, y), -F2(x, y)), F(x, y)); \quad (3)$$

$$FF := (x, y) \rightarrow o(p(-F1(x, y), -F2(x, y)), F(x, y)). \quad (4)$$

Крім того, за допомогою R-функцій можна будувати лінії рівня функції $FF(x, y)$ (рис. 21-22), що може доповнювати виразні можливості при створенні орнаментів (особливо в кольорі).

Геометричне перетворення площини – це взаємно однозначне відображення цієї площини на себе. Найбільш важливим геометричним перетворенням є рух, тобто перетворення, що зберігає відстань, а також масштабування [10]. Прикладами руху є центральна симетрія, паралельний перенос, поворот і осьова симетрія. Мотив – це головний елемент орнаменту, основна його складова. Подальшим узагальненням буде побудова орнаментів за допомогою комп'ютерних технологій.

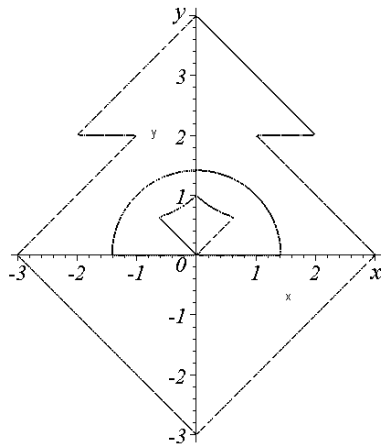


Рисунок 19 – Зображення, одержане за допомогою R-функцій (2)

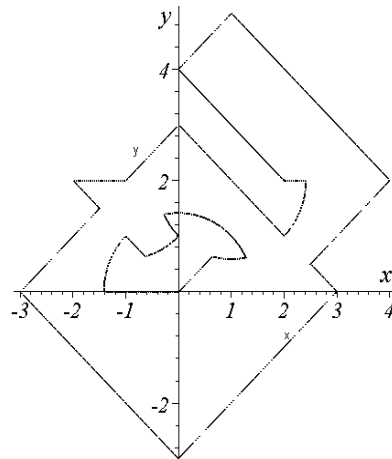


Рисунок 20 – Зображення, одержане за допомогою R-функцій (3)

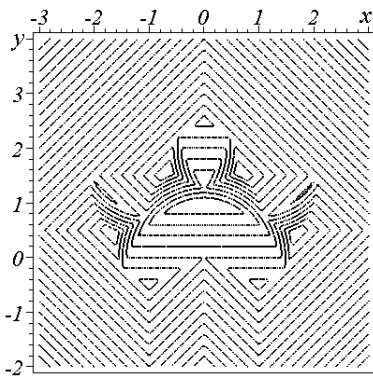


Рисунок 21 – Лінії рівня функції (4)

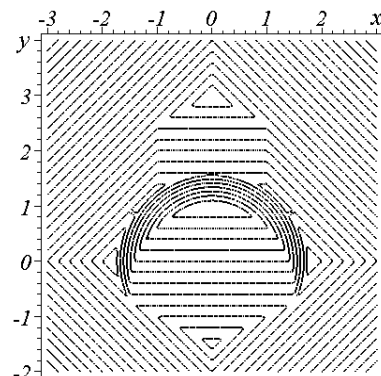


Рисунок 22 – Лінії рівня функції (5)

Графічні візерунки звичайно створюють для декоративних заставок, реклами, демонстрації можливостей апаратури. Програми для створення комп'ютерних візерунків звичайно базуються на досить простих алгоритмах, які можна вважати тестами майстерності програмування. Головним тут є ідея створення декоративного ефекту.

Якщо фігуру переміщати не обертаючи відносно свого "центра", то одержимо плоско-паралельний рух фігури, коли будь-який відрізок прямої на фігурі залишається паралельним самому собі. За "центр" фігури може бути прийнята будь-яка точка, жорстко пов'язана з фігурою.

Звичайно "центр" фігури (x_f, y_f) переміщують відносно центра візерунка (x_c, y_c) за певним законом

$$\begin{aligned} x_f &= x_c + F_x; \\ y_f &= y_c + F_y, \end{aligned}$$

де F_x, F_y – функції від параметрів.

У загальному випадку фігура може переміщатися, обертаючись відносно свого "центра" і деформуватися. При цьому параметри процедури зображення фігури повинні включати всі координати точок, які з'єднуються лініями. Координати i -тої точки фігури визначаються за формулами:

$$\begin{aligned} xxi &:= xf + Kxi * ((xi-xf) * \cos(A) - (yi-yf) * \sin(A)), \\ yyi &:= yf + Kyi * ((yi-yf) * \cos(A) + (xi-xf) * \sin(A)), \end{aligned}$$

де A – кут повороту фігури відносно свого "центра", відлічуваного у лівій системі координат екрана за годинниковою стрілкою стосовно осі X ;

xi, yi – початкові координати i -тої точки фігури;

xxi, yyi – нові координати i -тої точки фігури;

Kxi, Kyi – коефіцієнти масштабування координат i -тої точки по осях X і Y .

Наведемо приклад Maple-програми побудови візерунка, у якій реалізовано множину законів руху ліній відносно свого "центра".

```

q := evalf(W);
m := [[0,0],[1,0],[cos(q), sin(q)],[0,0]]:
w := []: t := T:
for k to N do
w := [op(w), m]:
sm := [m[2], m[3], m[1], m[2]]:
m := (1-t)*m + t*sm:
od:

```

Після наведеного підготовчого блоку слід скласти шість блоків, які відрізнятимуться коефіцієнтами у парах тригонометричних функцій

```

a1 := plot(w, axes=none, scaling=constrained):
m := [[0,0],[cos(2*q),sin(2*q)],
      [cos(1*q),sin(1*q)],[0,0]]:
w := []: t := T:
for k to N do
w := [op(w),m]:
sm := [m[2],m[3],m[1],m[2]]:
m := (1-t)*m+t*sm:
od:

```

В результаті виконання всіх шести блоків матимемо заготовку для побудови зображення з використанням оператора

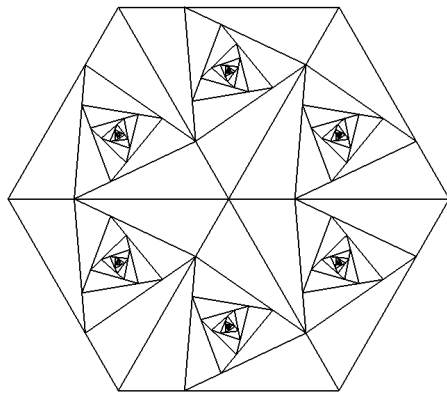
```

display(a1,a2,a3,a4,a5,a6, thickness=2);

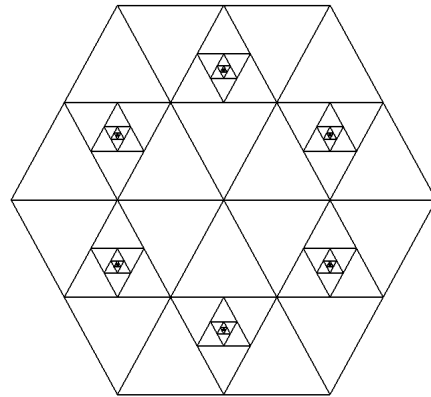
```

На рис. 23 наведено приклади виконання програми побудови візерунка при $N = 10$ залежно від параметрів W і T .

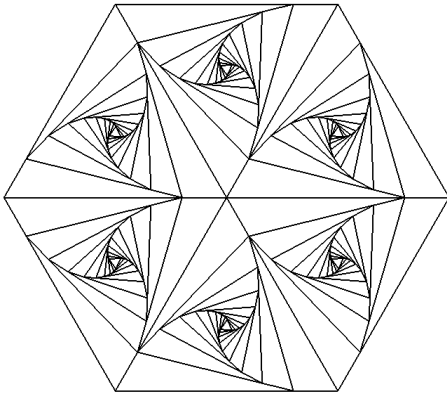
У програмуванні часто використовують рекурсивні оператори, наприклад, процедури, які містять звертання до самих себе. Такі звертання можуть бути прямими – тобто викликом процедури усередині самої процедури, або непрямыми – викликом інших процедур, усередині яких є виклик вихідної процедури.



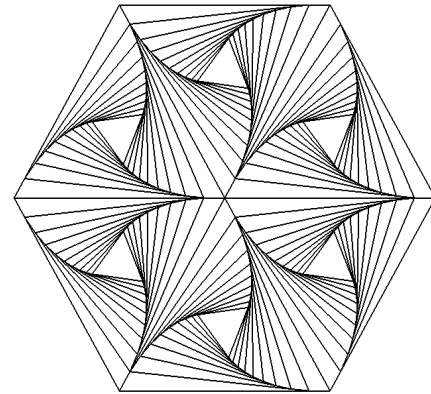
$$T = 0.3; W = \pi/3$$



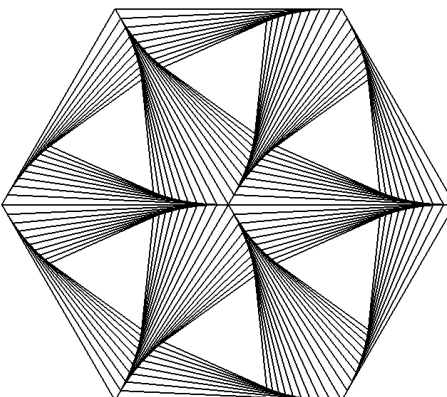
$$T = 0.5; W = \pi/3$$



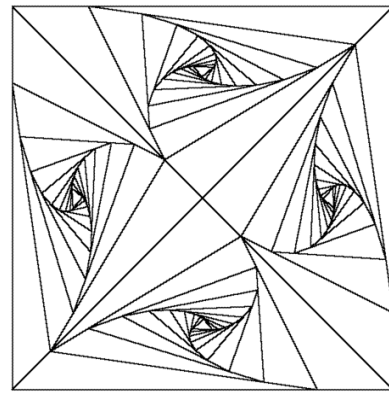
$$T = 0.8; W = \pi/3$$



$$T = 0.9; W = \pi/3$$



$$T = 0.95; W = \pi/3$$



$$T = 0.2; W = \pi/2$$

Рисунок 23 – Приклади виконання програми побудови візерунка залежно від параметрів T і W

Наведемо приклад програми рекурсивної побудови візерунка, де процедура **drevo** звертається до тієї ж процедури **drevo**.

```

drevo := proc(L, N, x0, y0)
local i; global s, p;
options remember;
s := s+1;
p[s] := plot([x0 + L*sqrt(M)*sin(M*t)*cos(t),
          y0 + L*sqrt(M)*sin(M*t)*sin(t), t=-W..W]);
if N > 1 then drevo(L/2, N-1, x0-L, y0+L);

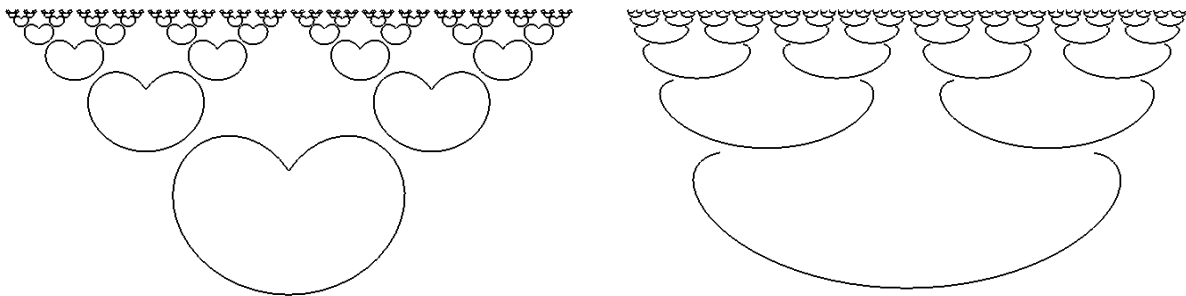
```

```

    drevo(L/2, N-1, x0+L, y0+L) fi;
RETURN(display([seq(p[i], i=1..2^N-1)], axes=NONE,
    scaling=constrained, numpoints=1000,thickness=2))
end:
s:=0: drevo(100, 7, 0, 0);

```

На рис. 24 наведено приклади виконання програми рекурсивної побудови візерунка залежно від параметрів W і M .



$$W = \pi/2; \quad M = 1,5$$

$$W = \pi/5; \quad M = 3$$

Рисунок 24 – Приклади виконання програми рекурсивної побудови візерунка залежно від параметрів W і M

Цікавим є створення візерунків побудовою дзеркальних відображень фігури. В цьому випадку принцип побудови візерунків близький до зображень дзеркального калейдоскопа. У калейдоскопі система із трьох дзеркал створює ефект декількох шестиразових відбиттів набору кольорових кристалів. Математично такий принцип побудови візерунка можна описати так. Є три вихідних з однієї точки променів – осей симетрії. Кут між променями дорівнює $2\pi/3$. Будується перша (початкова) фігура в секторі між першим і другим променями. Потім будується друга фігура як дзеркальне відображення першої фігури відносно другого променя, далі третя фігура, як дзеркальне відображення другої фігури відносно третього променя й так далі.

Наведемо приклад програми створення візерунка побудовою відбиттів початкової фігури відносно трьох осей відбиття.

```

fosi1 := x*sin(Pi/3) - y*cos(Pi/3);
fosi2 := y;
fosi3 := x*sin(Pi/3) + y*cos(Pi/3),

```

де кут вимірюється від осі Ox проти годинникової стрілки.

Початкову фігуру (наприклад, квадрат) задамо рівнянням у вигляді

$$f := (x,y) \rightarrow a - \text{abs}(x-x_0) - \text{abs}(y-y_0):$$

Для побудови результату симетричного відбиття відносно першої, другої та третьої осі симетрії слід застосувати формули:

```

X1 := (x,y) -> y*sin(2*Pi/3)+ x*cos(2*Pi/3);
Y1 := (x,y) -> x*sin(2*Pi/3)- y*cos(2*Pi/3);
f1 := (x,y) -> f(X1, Y1);

```

```

X2 := x;
Y2 := -y;
f2 := (x,y) -> f1(X2, Y2);
X3 := (x,y) -> y*sin(-2*Pi/3)+ x*cos(-2*Pi/3);
Y3 := (x,y) -> x*sin(-2*Pi/3)- y*cos(-2*Pi/3);
f3 := (x,y) -> f2(X3, Y3);

```

Побудову при $a = 1,5$; $x_0 = 2$; $y_0 = 1$ проміжних результатів відбиття здійснюємо операторами

```

F := implicitplot(f(x,y), x=-3..4, y=-3..4);
F1 := implicitplot(f1(x,y), x=-3..4, y=-3..4);
F2 := implicitplot(f2(x,y), x=-3..4, y=-3..4);
F3 := implicitplot(f3(x,y), x=-3..4, y=-3..4);

```

Результуюче зображення (яке можна продовжити в циклі) будується за допомогою оператора

```
display(F, F1, F2, F3, thickness=3);
```

На рис. 25 наведено приклади побудованих зображень.

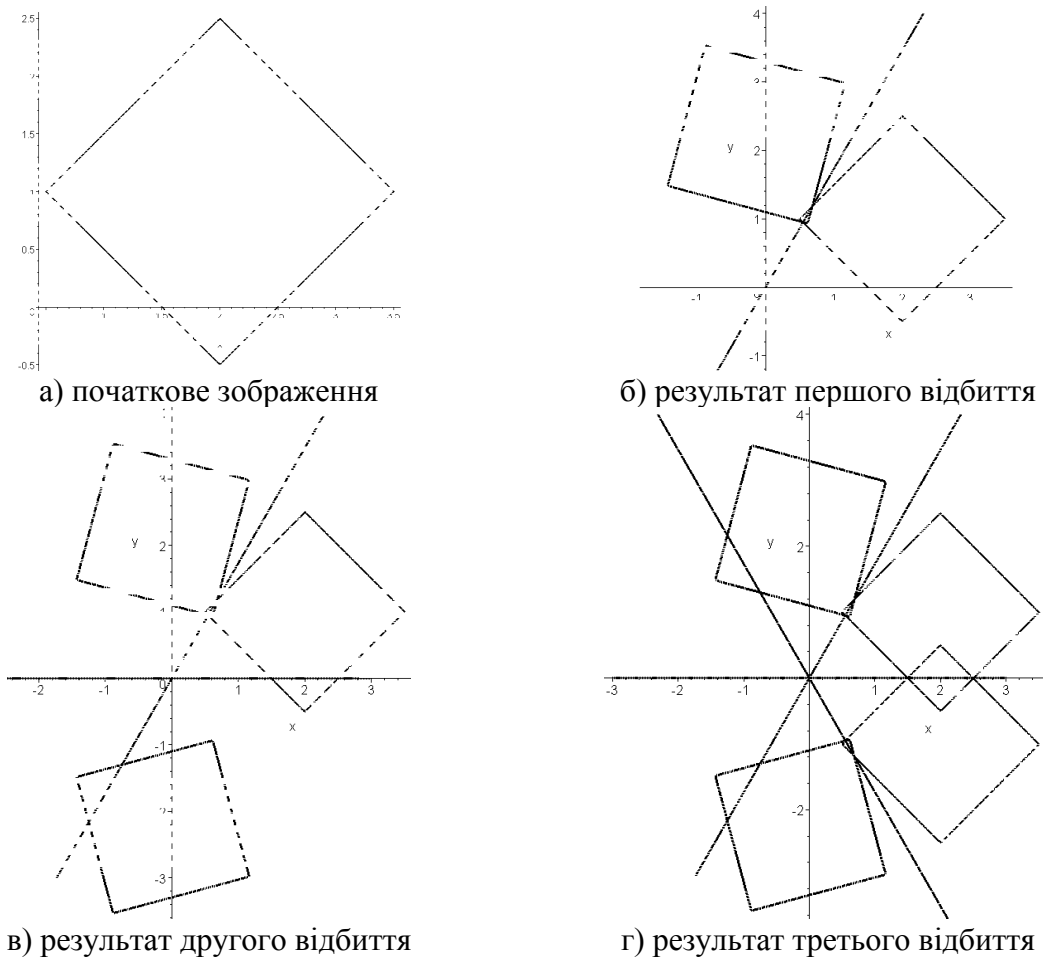


Рисунок 25 – Побудова симетричних зображень дзеркального калейдоскопа

Результати досліджень

Одержано новий спосіб побудови та коригування геометричної форми візерункових кривих засобами R -функцій. Незаперечною перевагою застосування R -функцій є можливість оперувати з опорними областями примітивів. Застосовано методи R -функцій до побудови рівнянь зображень, що мають симетрію.

Розробка геометричних моделей здійснювалася на основі засобів прикладної геометрії, обчислювальної математики, диференціальної та аналітичної геометрії, елементів R -функцій. У практичній реалізації створених моделей та алгоритмів використано елементи комп'ютерного математичного програмування в програмному середовищі пакету Maple. Використання R -функцій дозволило розв'язати задачу, аналогічну оберненій задачі аналітичної геометрії – тобто коли задане зображення необхідно описати його рівнянням.

Висновки

Головні принципи побудови візерунків можна реалізувати у середовищі алгоритмічної мови (наприклад, Maple) шляхом створення алгоритмів побудови зображень на площині. Наведений спосіб практичного застосування R -функцій у середовищі математичного пакету Maple дозволяє складати у неявному вигляді $F(x,y) = 0$ рівняння зображень з елементами симетрії. До переваг способу слід віднести формалізацію побудови зображень.

Список літератури

1. Кожин П.М. О древних орнаментальных системах Евразии // Этнознаковые функции культуры. М.: Наука, 1991. С. 129-151.
2. Збеневич В.Г. Ранний этап трипольской культуры на территории Украины. Киев: Наук. думка, 1989. 224 с.
3. Палагута И.В. Проблемы изучения спиральных орнаментов трипольской керамики // Stratum plus. 1999. № 2. С. 148-159.
4. Ткачук Т.М., Віденко М.Ю. Знакові системи трипільської культури // Енциклопедія трипільської цивілізації. Київ: Укрполіграфмедіа, 2004. Т. 1. С. 433-471.
5. Рвачев В.Л. Теория R -функций и некоторые ее приложения. Киев: Наукова думка, 1982. 552 с.
6. Рвачев В.Л., Шапиро В., Шейко Т.И. Применение метода R -функций к построению уравнений локусов, обладающих симметрией // Электромагнитные волны и электронные системы. 1999. Т. 4. № 4. С. 4-20.
7. Толок А.В., Семерич Ю.С., Шейко Т.И. Построение симметричных функций для симметричных чертежей // Вісник Запорізького державного університету. 2001. № 2. С. 1-16.
8. Челомбійсько В.Ф., Ткаченко В.П. Спосіб опису та побудови симетричних орнаментів за допомогою R -функцій. // Технічна естетика і дизайн. 2011. Вип. 8. С. 301-307.
9. Челомбійсько В.Ф., Ткаченко В.П. Побудова орнаментів з елементами симетрії відносно точки з використанням R -функцій // Геометричне та комп'ютерне моделювання. 2011. Вип. 28. С.99-103.
10. Челомбійсько В.Ф. Реалізація деяких принципів побудови візерунків засобами операторів мови Maple // Теорія та практика дизайну. 2012. Вип. 2. С.127-132.