



Національний технічний університет України
«Київський політехнічний інститут»

Факультет Прикладної Математики
Кафедра Системного Програмування і Спеціалізованих
Комп'ютерних Систем

ПРАКТИЧНА РОБОТА №4
з дисципліни
«Комп'ютерна Графіка»
ТЕМА: «Фрактальна графіка»

Група: KB-11

Виконав: Брюханов О.

Оцінка: ____

Київ – 2023

Код програми

```
// koch.pde
```

```
void kochCurve(float x1, float y1, float x2, float y2, int order) {
    if (order == 0) {
        line(round(x1), round(y1), round(x2), round(y2));
        return;
    }

    float Lx = (x2 - x1) / 3.0;
    float Ly = (y2 - y1) / 3.0;

    float Bx = x1 + Lx;
    float By = y1 + Ly;
    float Dx = Bx + Lx;
    float Dy = By + Ly;

    float Cx = (Bx + Dx) / 2;
    Cx -= (1 / 2) * (Dx - Bx);
    Cx += (sqrt(3) / 2) * (Dy - By);

    float Cy = (By + Dy) / 2;
    Cy -= (sqrt(3) / 2) * (Dx - Bx);
    Cy += (1 / 2) * (Dy - By);

    kochCurve(x1, y1, Bx, By, (order - 1));
    kochCurve(Bx, By, Cx, Cy, (order - 1));
    kochCurve(Cx, Cy, Dx, Dy, (order - 1));
    kochCurve(Dx, Dy, x2, y2, (order - 1));
}

void setup() {
    size(1280, 720);
    background(255);

    kochCurve(350, 150, 750, 150, 4);
    kochCurve(750, 150, 550, 490, 4);
    kochCurve(550, 490, 350, 150, 4);
}
```

```
// mandelbrot.pde
```

```
float cabs(float[] a) {
    return sqrt(pow(a[0], 2) + pow(a[1], 2));
}

float[] cmul(float[] a, float[] b) {
    float[] res = {-1, -1};
    res[0] = (a[0] * b[0]) - (a[1] * b[1]);
    res[1] = (a[1] * b[0]) + (a[0] * b[1]);

    return res;
}
```

```

float[] cadd(float[] a, float[] b) {
    float[] res = {a[0] + b[0], a[1] + b[1]};

    return res;
}

int mandelbrot(float[] c, int max_iter) {
    float[] z = {0, 0};
    int n;
    for (n = 0; n < max_iter && cabs(z) <= 2; n++)
        z = cadd(cmul(z, z), c);

    return n;
}

void mandelplot(int w, int h) {
    int[] Re = {-2, 1};
    int[] Im = {-1, 1};

    for (int i = 0; i < w; i++) {
        for (int j = 0; j < h; j++) {
            float real = Re[0] + (i / (float) w) * (Re[1] - Re[0]);
            float imag = Im[0] + (j / (float) h) * (Im[1] - Im[0]);
            float[] c = {real, imag};

            int m = mandelbrot(c, 120);
            if (m == 120) {
                set(i, j, #000000);
                continue;
            } else {
                int hue = 256 * m / 120;
                colorMode(HSB, 360, 100, 100);
                set(i, j, color(hue, 90, 90));
            }
        }
    }
}

void setup() {
    size(1280, 720);
    background(255);

    mandelplot(1280, 720);
}

// Fern.pde

void fern(int its) {
    float[] coords = {1, 0};
    float old_x;
    int r = 80;

```

```

for (int i = 0; i < its; i++) {
  old_x = coords[0];
  int cookie_for_ruby = (int) random(0, 100.1);
  if (cookie_for_ruby <= 85) {
    coords[0] = 0.85 * coords[0] + 0.04 * coords[1];
    coords[1] = -0.04 * old_x + 0.85 * coords[1] + 1.6;
  } else if (cookie_for_ruby <= 92) {
    coords[0] = 0.25 * coords[0] - 0.26 * coords[1];
    coords[1] = 0.23 * old_x + 0.25 * coords[1] + 1.6;
  } else if (cookie_for_ruby <= 99) {
    coords[0] = -0.15 * coords[0] + 0.3 * coords[1];
    coords[1] = 0.26 * old_x + 0.2 * coords[1] + 0.44;
  } else {
    coords[0] = 0;
    coords[1] *= 0.16;
  }

  int hue = 360 * cookie_for_ruby / 100;
  colorMode(HSB, 360, 100, 100);
  set((int) (400 + r * coords[0]), (int) (850 - r * coords[1]), color(hue, 90, 90));
}
}

void setup() {
  size(1000, 1000);
  background(255);

  fern((int) pow(10, 5));
}

```

Результати



