7. Generate fractal patterns using i) Bezier ii) Koch Curve

ii) Koch Curve

Source Code :

```
#include <GL/glut.h>
#include <math.h>

void drawLine(float x1, float y1, float x2, float y2) {
   glBegin(GL_LINES);
   glVertex2f(x1, y1);
   glVertex2f(x2, y2);
   glEnd();
}

void kochCurve(float x1, float y1, float x2, float y2, int iterations) {
   if (iterations == 0) {
      drawLine(x1, y1, x2, y2);
      return;
   }

   float deltaX = (x2 - x1) / 3.0;
   float deltaY = (y2 - y1) / 3.0;
   float xA = x1 + deltaX;
   float yA = y1 + deltaY;
   float xC = x2 - deltaX;
```

```
    float yC = y2 - deltaY;
  float xB = xA + (xC - xA) * 0.5 - (yC - yA) * sqrt(3) / 2.0;
    float yB = yA + (yC - yA) * 0.5 + (xC - xA) * sqrt(3) / 2.0;

    kochCurve(x1, y1, xA, yA, iterations - 1);
    kochCurve(xA, yA, xB, yB, iterations - 1);
    kochCurve(xB, yB, xC, yC, iterations - 1);
    kochCurve(xC, yC, x2, y2, iterations - 1);
}

void display() {
    glClear(GL_COLOR_BUFFER_BIT);
    glColor3f(1.0, 1.0, 1.0); // Set line color to white

    float x1 = -0.5, y1 = 0.0;
    float x2 = 0.5, y2 = 0.0;

    int iterations = 4;

    kochCurve(x1, y1, x2, y2, iterations);

    glFlush();
}
void init() {
    glClearColor(0.0, 0.0, 0.0, 0.0); // Set background color to black
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(-1.0, 1.0, -1.0, 1.0); // Set orthographic projection
}
```

```
int main(int argc, char **argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize(500, 500);
    glutCreateWindow("Koch Curve");
    init();
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```

Output :