

```

#include <GL/glut.h>
#include<iostream>
#include <math.h>
using namespace std;

int iteration;
GLfloat oldx = -0.7, oldy = 0.5;

void drawkoch(GLfloat drctn, GLfloat len, GLint iter) {
    GLdouble angle = 0.0174533 * drctn;
    GLfloat newX = oldx + len * cos(angle);
    GLfloat newY = oldy + len * sin(angle);

    if (iter == 0) {
        glVertex2f(oldx, oldy);
        glVertex2f(newX, newY);
        oldx = newX;
        oldy = newY;
    }
    else {
        iter--;

        drawkoch(drctn, len, iter);
        drctn += 60.0;
        drawkoch(drctn, len, iter);
        drctn -= 120.0;
        drawkoch(drctn, len, iter);
        drctn += 60.0;
        drawkoch(drctn, len, iter);
    }
}

void display() {

    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_LINES);
    glColor3f(0.9, 1.0, 0.8);
    cout << "Enter the number of iterations : ";
    cin >> iteration;
    drawkoch(0.0, 0.04, iteration);
    drawkoch(-120.0, 0.04, iteration);
    drawkoch(120.0, 0.04, iteration);
    glEnd();
    glFlush();
}

int main(int argc, char** argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);

```

```
glutInitWindowSize(750, 750);  
glutInitWindowPosition(0, 0);  
  
glutCreateWindow("Koch snowflake Curve");  
glutDisplayFunc(display);  
glutMainLoop();  
}
```