#include <GL/glut.h> // Header File For The GLUT Library

#include <math.h>

#include<stdio.h>

GLfloat oldx=-300,oldy=300;

GLint iter;

void drawkoch(GLfloat dir,GLfloat len,GLint iter)

{

GLdouble dirRad = 3.14/180 \* dir;

GLfloat newX = oldx + len \* cos(dirRad);

GLfloat newY = oldy + len \* sin(dirRad);

if (iter==0)

{

glVertex2f(oldx, oldy);

glVertex2f(newX, newY);

oldx = newX;

oldy = newY;

}

else

{

iter--;

//draw the four parts of the side \_/\\_

drawkoch(dir, len, iter);

dir += 60.0;

drawkoch(dir, len, iter);

dir -= 120.0;

drawkoch(dir, len, iter);

dir += 60.0;

drawkoch(dir, len, iter);

}

}

void mydisplay(){

glClear( GL\_COLOR\_BUFFER\_BIT );

glBegin(GL\_LINES); //call drawkoch 3 times, one for each side of the triangle, changing direction each time

glColor3f(1.0, 0.0, 0.0); // make lines red

drawkoch(0.0, 20,iter);

glColor3f(0.0, 1.0, 0.0);` // Set The Color

drawkoch(-120.0,20, iter);

glColor3f(0.0, 0.0, 1.0); // Set The Color

drawkoch(120.0,20,iter);

glEnd();

glFlush();

}

void init(void)

{

glClearColor(0.0,0.0,0.0,0.0);

gluOrtho2D(-600.0,600.0,-600.0,600.0);

}

int main(int argc, char\*\* argv)

{

printf("\n How many iteration you want (1/2/3)");

scanf("%d",&iter);

glutInit(&argc,argv);

glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB);

glutInitWindowSize(1000,1000);

glutInitWindowPosition(0,0);

glutCreateWindow("Koch Snowflake - Marcus Young");

init(); /\* Initialize window. \*/

glutDisplayFunc(mydisplay);

glutMainLoop();

}