#include <cmath>
#include <iostream>
#include <string>
#include "console.h"
#include "gwindow.h"
#include "gobjects.h"
#include "simpio.h"
using namespace std;

const int WINDOW_WIDTH = 800; // GWindow dimensions
const int WINDOW_HEIGHT = 600;
const int LINE_HEIGHT = 17; // Thickness of the line
const int LINE_SPACING = 20; // Top of one line to the top of next
const int STARTING_X = 50; // Initial line placement and size
const int STARTING_Y = 50;
const int STARTING_WIDTH = 700;
const bool DO_COLORS = true; // B&W drawing or rainbow colors
const string COLORS[] = {"MAGENTA", "BLUE", "CYAN", // Rainbow colors "GREEN", "YELLOW", "ORANGE", "RED"};
const int N_COLORS = sizeof(COLORS) / sizeof(string); // take CS107

void cantorSet(GWindow& window, int x, int y, int length, int levels);

int main() {
    GWindow window(WINDOW_WIDTH, WINDOW_HEIGHT);
    window.setWindowTitle("Cantor Set Fractal");
    if (!DO_COLORS) {
        window.setColor("BLACK");
    }
    while (true) {
        int nLevels = getInteger("How many levels? ");
        if (nLevels > 0) {
            window.clear();
            cantorSet(window, STARTING_X, STARTING_Y, STARTING_WIDTH, nLevels);
        } else {
            break;
        }
    }
    return 0;
}

/*
 * Draws the Cantor Set fractal
 * (UPDATE: with rainbow Pride colors!)
 * @param window is the graphics window to use as a canvas for drawing
 * @param x, y are the desired (x,y) coords of top left corner of the drawing
 * @param length is the desired horizontal width of the drawing
 * @param levels is the desired levels of depth for the fractal drawing
 * */

/*
 * cantorset.cpp
 * @author Cynthia Lee, from code by Marty Stepp
 * @version 1920-Aut
 * A program to demonstrate the order of operations in the unfolding of
 * a recursive function, by animating the drawing of a simple fractal
 * image called the Cantor Set.
 */
A level-1 Cantor Set is just a line, and a level-2 set is a line followed by the first third and last third of that line, and so on. Here is a crude text illustration of a level-4 Cantor Set:

-------------------------------
---------         ---------
---   ---         ---   ---
- -   - -         - -   - -

A level-0 or below produces no output.

```c
void cantorSet(GWindow& window, int x, int y, int length, int levels) {
    if (levels > 0) {
        // draw our own line
        if (DO_COLORS) {
            int colorsIndex = levels % N_COLORS; // % to ensure valid index
            window.setColor(COLORS[colorsIndex]);
        }
        window.fillRect(x, y, length, LINE_HEIGHT);
        window.sleep(20); // slows drawing process, for observation

        // recursively draw next lines
        int newY = y + LINE_SPACING;
        int newLength = length / 3;
        int newLevels = levels - 1;
        // left third
        cantorSet(window, x, newY, newLength, newLevels);
        // right third
        cantorSet(window, x + 2 * length / 3, newY, newLength, newLevels);
    }
}
```