\*

\* file: Sierpinski.kt

\* date: 2019-06-23

\* @author: SS

\*/

/\*\*

\* Starting from n = 1, this program loops forever displaying

\* the level-n Sierpinski triangle, waits for the user to hit

\* any key, then displays the level-((n+1)%8) Sierpinski triangle,

\* and waits for a key hit by the user, and so on, ad infinitum.

\*/

import processing.core.PApplet

import kotlin.math.\*

fun main(args: Array<String>) {

PApplet.main("SierpinskiSketch");

}

class SierpinskiSketch : PApplet() {

var level = 0

override fun settings() {

fullScreen()

noLoop()

}

override fun draw() {

background(250F)

val len = 0.8F \* min(width, height)

sierpinski((width-len)/2F, len, len, ++level)

level %= 8

}

override fun keyPressed() {

redraw()

}

fun sierpinski(x: Float, y: Float, len: Float, n: Int) {

if (n < 1) return

if (n == 1) {

fill(0)

triangle(x, y, x+len, y, x+len/2F, y-sqrt(3F)\*len/2F)

fill(255)

} else {

sierpinski(x, y, len/2F, n-1)

sierpinski(x+len/2F, y, len/2F, n-1)

sierpinski(x+len/4F, y-sqrt(3F)\*len/4F, len/2F, n-1)

}

}

}