Lab Assignment 1

D7045E

2020-10-31

Write a program using either OpenGL or WebGL with a simple GUI that displays a 2D Koch Snowflake for certain recursion depths. Wikipedia has a good description of this fractal curve:

https://en.wikipedia.org/wiki/Koch_snowflake

The GUI should provide a way to set/change the recursion depth between 1 and 7. It should at all time show the snowflake as it looks at the given depth.

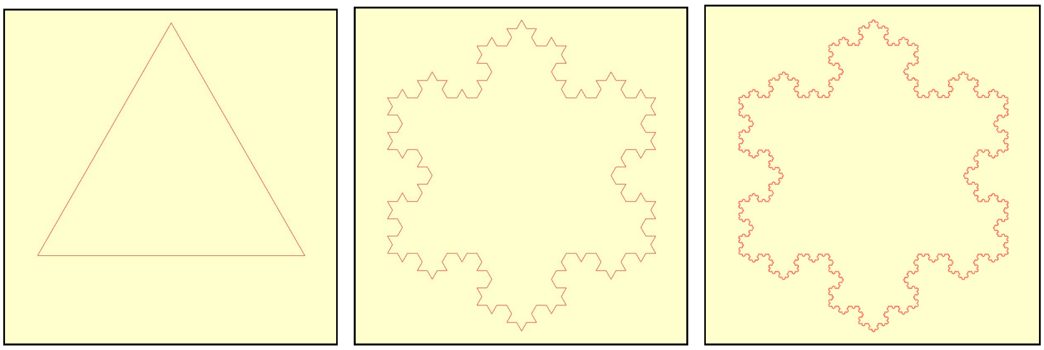

Figure 1: The snowflake at depths 1, 4, and 7 (right-most).

1. Computing the snowflake as a sequence of vertices

Represent the snowflake as a list of vertices as they occur along the boundary. Use recursion over the depth to compute the list.

- NB! Be sure to use vector operations, **not angles and trigonometry**. As a general rule, avoid to compute with angles because it often causes numerical problems.

Limit coordinates at all depths to lie in the range]-1..1[only, since this is what will be displayed. This means you have to be careful when you set the coordinates on depth 1, or the lower part of the snowflake might extend beyond -1 in the negative y direction at depths greater than 1. Find a positioning such that for recursion depths larger than 1, the snowflake is centered close to the origin.

Figure 2 below depicts how a single edge $[p_0,p_1]$ between p_0 and p_1 is replaced by a sequence of 4 shorter edges $[p_0,q_0]$, $[q_0,a]$, $[a,q_1]$, and $[q_1,p_1]$ when the depth increases by 1. Assuming side lengths are 1 at depth i (to the left in the figure), they decrease to ⅓ at depth i+1 (shown to the right of the figure). Note that the length of $[q_0,q_1]$ is also ⅓ but not in the figure simply because there is no good place to write it.

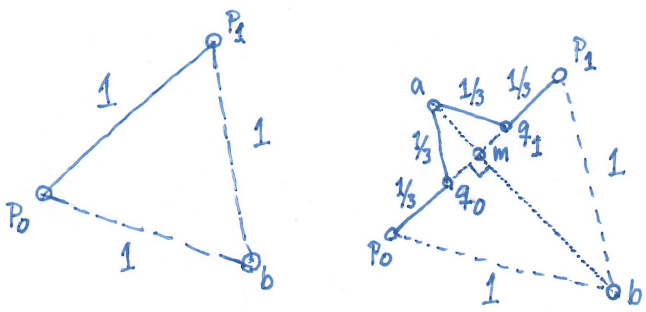

Figure 2: Calculating vertex coordinates.

In Figure 2, m and b are helpers that can be useful in calculating the vertex coordinates of q_0, a, and q_1. While m is the midpoint of $[p_0,p_1]$, b (as in *base*) is a vertex on the snowflake at depth i such that $[p_0,p_1]$, $[p_0,b]$, and $[p_1,b]$ all have the same length. The base must lie on the opposite side of the line through $[p_0,p_1]$ than a.

At depth i+2, when we replace $[q_0,a]$ with 4 shorter edges, q_1 is the base. Likewise, q_0 is the base when $[a,q_1]$ is replaced. When edge $[p_0,q_1]$ is replaced, the added triangular dent will protrude upwards so the base must lie below on $[p_0,b]$. When $[q_1,q_1]$ is replaced, the base instead lies on $[p_0,b]$.

2. Graphics and coloring

You may base your graphics part of your program on the example from Lecture 1, where a simple outline with (just) 5 edges is displayed. Change the fragment shader so the outline of the snowflake becomes red. Also change the background so it is pale yellow rather than grey.

With the exception of depth 1, when the snowflake is just a triangle, the snowflake should be centered at the origin and displayed in the center of the yellow area.

3. Grading and bonus assignments for higher grades

Grade 3

Passing the assignment as stated above gives the grade 3.

Grade 4

For grade 4, do what is required for grade 3, and also the following:

Make the whole snowflake simultaneously

- *rotate slowly at a constant pace;*
- *slowly and smoothly scale up and down (oscillate in relative size); and*
- *circle (move, translate) around the origin.*

Help: Check out the course book *Interactive Computer Graphics*: Sections 3.8-3.10 in the 6th Edition or Sections 4.8-4.10 in the 7th Edition. Note that they treat 3D in those sections, not 2D, but it's essentially the same: Just skip/erase row 3 and column 3 from the transformation matrices, and use $[x_i,y_i,1]^T$ for each point (x_i,y_i) that is transformed. Or pretend (x_i,y_i) is a 3D point with z=0 (represented by $[x_i,y_i,0,1]$) and use the transformation matrices as they are.

Grade 5

For grade 5, do what is required for grades 3 and 4, and also:

Color the area inside the snowflake red and color the snowflake curve itself black. Do not re-color the pixels more than a (low) constant number of times.

Help: Find/compute a set of triangles that together cover the interior of the snowflake. In addition tothe big triangle at depth 1, each recursive step can contribute with its own unique triangle to this set.