Tessellation and Twist

COMP 4302/6909 — Assignment 2

Due: February 14, 2020 (Friday) 11:59 PM

Assignment Objective:
In this assignment you should start getting familiar with the graphics development environment, the compiler and the WebGL programming library, particularly by exploring how to render a set of polygons onto the canvas.

Your Task:

Your task is to start exploring the rendering environment. For this, you will be asked to first display a triangle made of only three corners. After that, you will be asked to display a triangle that is tessellated into a much finer triangle mesh. The triangle mesh can be produced by subdividing the original triangle into 20 subdivisions across each of the sides of the original triangle (see Figure below).

Getting Started:

Download the source code available for the demos for the course.

http://interactivecomputergraphics.com/

http://interactivecomputergraphics.com/Code/

http://interactivecomputergraphics.com/SIGGRAPH_ASIA_19/WebGL%202.0/Examples/

You should be able to run and test most of these programs using an HTML5 & WebGL compatible web browser and a computer with hardware-supported graphics capabilities. Alternatively, you can check out different online tutorials on how to render polygons in 2D using WebGL. In cases where some external files are used, the examples will only work if you setup a local web server to try the examples.

Grading:

Your program will be tested and graded using a standard WebGL platform. The grade will be based on your program's functionality (whether or not it works under different settings), as well as the efficiency of your implementation. The weights for different components are as follows:

- Showing two triangles, one simple and the other tessellated, both in line/wireframe drawing style (see below) (10%)
- Showing two tessellated and twisted triangles, both in filled-color form (10%)
- Show four other polygons, starting from a square up to an octagon (skipping heptagon: square, pentagon, hexagon, octagon), in line form (20%)
- Show the same as in the previous item, but as tessellated polygons (in line/wireframe drawing style), each polygon being split into a smaller set of triangles (using 5 recursive subdivisions) (20%)

- Show the same as the item above, but as filled and twisted polygons, starting from a triangle on the left to an octagon (skipping heptagon: triangle, square, pentagon, hexagon, octagon), in filled-color form (20%)
- Provide an interface to select among the different combinations (for example, insert a menu item to select the shape to be drawn, or another menu to select the rendering mode (filled vs. wireframe), or have buttons in the html page to select among the options above. (20%)
- Extra/Bonus Marks: Show the tessellated polygons with different colors per vertex, using flat or smooth shading (10%)
- Explain your modes of operation in a text file (how the interface is expected to be operated) so the TA can understand how to operate your program and what he is expected to experience.

What and How to Hand in:

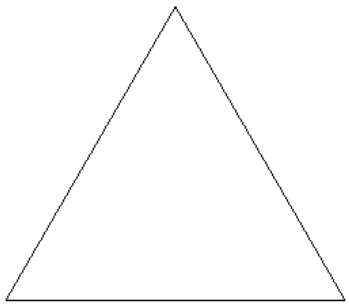
You are handing in the source of your program, as well as any data files required for running your program. Your source code must contain sufficient internal documentation to facilitate grading. This includes your names, student numbers, a brief description of what the program does, which items you claim to have completed and a listing of known bugs, if any, at the top of the file. Send in your source program(s) through the Direct2Learn Dropbox as a single .zip file. No late submission is allowed.

Verifying your submission:

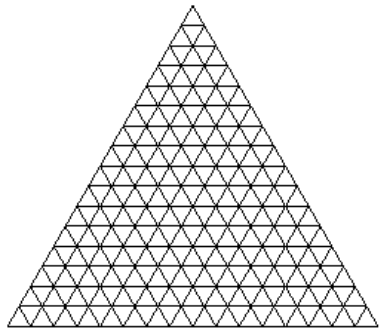
Since late submissions are not allowed, you should be careful to make sure each of your submissions has been properly done. Once you have placed your assignment into the Dropbox, click on the link that indicates our submission has been done, download it, open it, revise the contents, test that the program you have submitted actually is the one intended (not the source files, for example), and if it does not correspond to what you want the TA to mark, you must resubmit again, until you are satisfied, the TA will only mark the last submission with the timestamp that is prior to the indicated deadline.

Include any special instructions for the TA in a README.txt file.

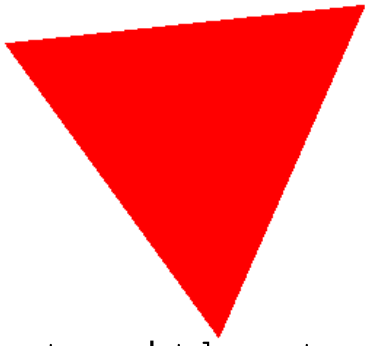
Some of the outputs of the program should look like this:



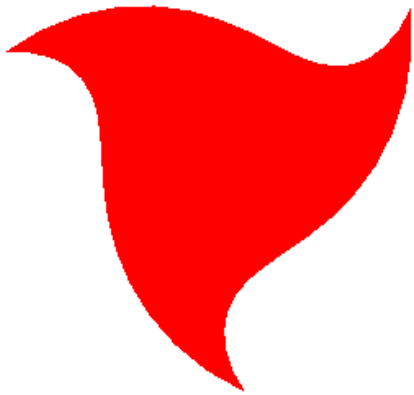
triangle



Tessellated triangle



Twist without tessellation



Twist with tessellation