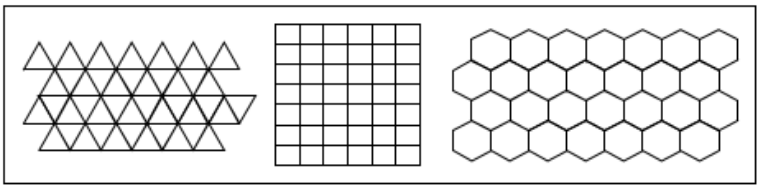
CS 371 – Assignment 1

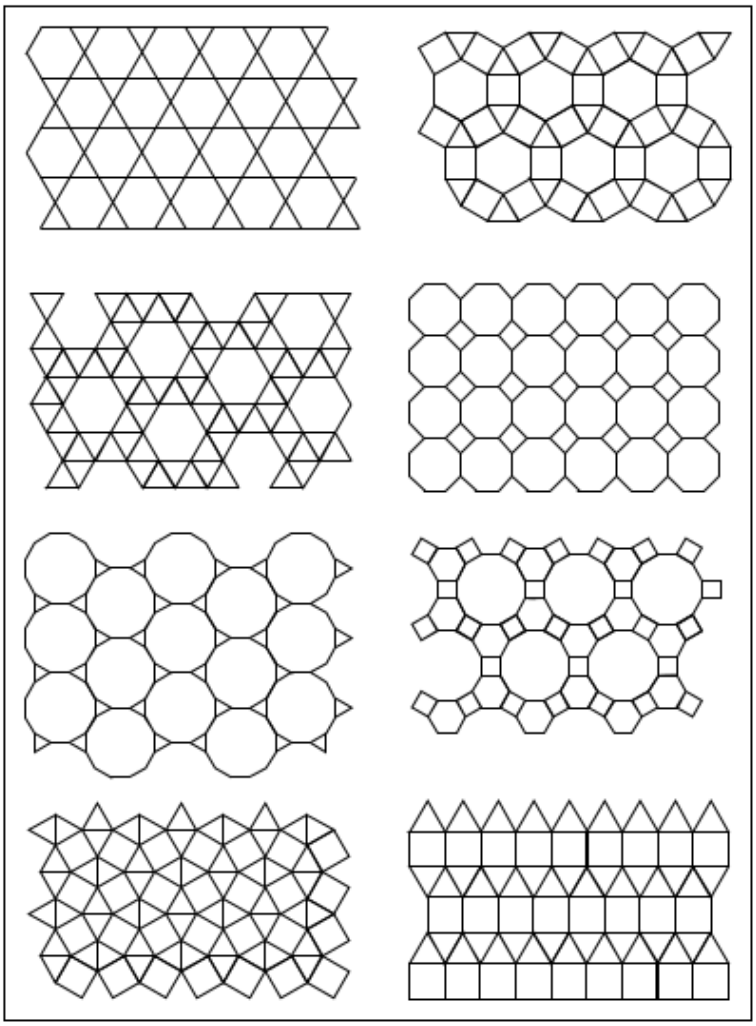
Due: 11:59pm, Friday, September 15th

This assignment will require that you thoroughly understand viewports and world coordinate windows. Computer graphics offers a powerful tool for creating pleasing pictures based on geometric objects. One of the most intriguing types of pictures is called a *tiling* – a picture composed of smaller objects that apparently repeat "forever" in all directions. For example, the figure below shows three basic tilings composed of triangles, squares, and hexagons respectively. In such a case the triangle, square, or hexagon are called the *prototile* for the tiling. To draw a tiling over the plane based on such a prototile, a collection of viewports is created side-by-side that covers the display surface, and the prototile is drawn once in each viewport.



Tilings comprised of just one prototile can also be thought of from the perspective of each vertex in the tiling and then considering the polygons that surround the vertex. For instance, in the tiling by triangles, each vertex is surrounded by six triangles. The *Schläfli number* for this tiling is 3.3.3.3.3.3, the six three's denoting 6 triangles. Similarly the Schläfli number for the tilings by squares and hexagons are 4.4.4.4 and 6.6.6 respectively.

Tilings can also be comprised of more than one prototile. For example, below are eight tilings known as the *Archimedean tilings*.



Here the Schläfli number for the tiling in the upper left corner is 3.6.3.6, meaning that each vertex is surrounded by four polygons – a triangle, a hexagon, a triangle, and a hexagon. Similarly the Schläfli numbers for the other seven tilings are 4.6.4.3, 3.3.3.3.6, 4.8.8, 3.12.12, 4.6.12, 3.3.4.3.4, and 3.3.3.4.4.

Minimally what you must do on this assignment is to pick one of these eight Archimedean tilings and write a WebGL program that tiles your canvas with that pattern. Of course, you are encouraged to go well beyond this minimal (90%) requirement to earn GGW points.

Your source code files *must be named assignment1.html and assignment1.js* (all lower case). You should put those two files (and only those two files) in a zip file named *assignment1.zip* and submit the zip archive to the "Assignment 1 Submissions" dropbox on D2L. If your source code files are named differently or if you submit multiple files to the dropbox, 10 points will be deducted from your score.

I will definitely read the introductory documentation block in your Javascript file. By doing that I should learn

- The Schläfli number of the pattern you chose for your tiling.
- Any options you have that might qualify for "golly-gee-whiz" points. Be sure you tell what what you're doing above and beyond "plain vanilla" and how one must use the program to access those options. If you fail to document these "golly-gee-whiz" features, you won't receive credit for them.
- Any internet sources from which you may have cut-and-pasted some code into your program. Remember that it is OK to do this as long as you cite the source.

Additionally, be sure that you indent your code correctly – Use `jsbeautifier.org` if it's not easy to do in your editor.

Finally, one extra tool you can use to debug your JavaScript code is `jshint`, which is installed on the Linux lab machines. Typing `jshint <filename>.js` will typically output a bunch of warnings or errors that will help you find typos and other bugs in your code. For more info, see `jshint.com`.