CS 425 Lab Exercise – Creating and transforming cones

Follow instructions below and in "TODO" comments in the code provided. Submit answers to the given questions via GradeScope, as instructed in class.

1. There are two variations of this exercise, each comprised of 3 files to work on:
   a. Functions version – (coneFunctionsExercise.html, coneFunctionsExercise.js, coneFunctions.js)
   b. OO Class version – (coneClassExercise.html, coneClassExercise.js, coneClass.js)
   c. In each case the first file is the web page, the third is the JavaScript to make and draw a cone, and the second draws axes and some cones by calling the methods in the third.
   d. You may choose to work with either version, or if you have time and interest, you can do both.
   e. (For expediency, these instructions refer to the functions version of the files.)

2. On a piece of paper or the equivalent, draw a sketch of a cone, with the base centered at the origin and the point pointing upwards along the Y axis. The height and radius of the cone should both be 1.0. Label the point of the cone as vertex 0. Then vertex 1 lies along the X axis at X = 1. Vertices continue in a counter-clockwise direction until they get back to the beginning. Vertex 1 has to be duplicated at the end, making it vertex nSectors + 1.

3. In coneFunctions.js, find TODO1 in createCone( ) and insert the code needed to fill the colors and points arrays. Then find TODO2 and insert the code to push the color data to the GPU. Finally in this file, find TODO3 in renderCone( ) and insert the code to connect the color buffer data to the "vColor" vertex shader variable and also fill in the parameters for drawArrays( ) to draw the sides of the cone. [The class version will find the constructor( ) and render( ) methods respectively.]

4. In condFunctionsExercise.js, find TODO4 in init( ) and insert parameters to call createCone( ). Then in the same file find TODO5 and insert correct parameters into 3 functions.

   Run the program. You should see a yellow rectangle with a multicolored cone and axes sticking out the top and sides. Experiment with changing nConeSectors. What do you get when NConeSectors is 3 or 4? What is the minimum to make a "nice" looking cone?

5. In coneFunctionsExercise.js, find TODO6 and TODO7, and edit the code to make more cones, this time with a solid color. How does it look without lighting models set up?

6. Try to adjust the viewpoint to look at the bottom of the cone. Does it have a bottom? Edit the coneFunctions.js file to add one. (May need to comment out the sides to see the bottom well.)