Algorithm 2 Shapes

Introduction to Computer Graphics, Fall 2021

Due Friday, September 24th at 6:00pm

Instructions: You may discuss this assignment with other students, but you must abide by the rules stated in the collaboration policy (no notes from any discussions, your handin must be completely written up by you and contain only your own work) and write the logins of the student(s) you collaborated with next to each problem. Hand in the assignment on Gradescope as a PDF (please make sure the document is anonymous) no later than 6:00pm EST on the due date. *Late hand-ins are not accepted under any circumstances.*

Demo

Before starting on this assignment, play around with the shapes demo by running cs1230_demo from the terminal to get an idea of what the completed project would look like, or take a look at the video recording . Make sure you show the Shapes dock with menu item, as it may not be visible by default. Also make sure under the tab, orbit camera is toggled on so you can rotate and zoom with your mouse/trackpad. Questions in this assignment will be based on the demo.

Notation: p_1 and p_2 refer to the values of Parameter 1 and Parameter 2, respectively. These parameters affect the degree of tessellation of each shape. Parameter 1 generally describes the number of "stacks" while Parameter 2 generally describes the number of "slices".

1 Cube

[1 point] 1.1: Take a look at one face of the cube. Change the tessellation parameter (p_1). Notice how the number of triangles on the face changes. When $p_1 = n$, how many triangles are on one face?

[1 point] 1.2: The cube you'll be tessellating is a unit cube centered at the origin, extending from $-\frac{1}{2}$ to $\frac{1}{2}$ along each axis. Consider the face that lies in the +YZ plane (meaning that it is parallel to the YZ plane and is in the positive x domain). What is the unit outward-pointing normal vector of this face? (Note: in general, when asked for a normal, you should always give a unit vector, meaning a vector of length one). At a point P at a smooth location on the boundary of some solid S, there's a normal line (perpendicular to the plane that's tangent to S at P). Given a nonzero vector v directed along this line, if $P + tv$ is not in S, for all sufficiently small positive numbers t, then v is said to be a normal. When asked for "the" outward normal, you should report $\frac{1}{||v||}v$, which is a unit-vector in the same direction as v.

2 Cylinder

[1 points] 2.2: Draw a few points of the curve in the xy-plane defined by

$$p(t) = (\cos t, \sin t)$$

where $0 \leq t \leq 2\pi$. (Feel free to use software to do this, but make certain you know which t-values correspond to which points). Now suppose we a want a curve parallel to this one in the plane $z = 2$. Write down a curve-definition that describes this parallel curve in the form

$$q(t) = (\_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_)$$

where $0 \leq t \leq 2\pi$.

[1 points] 2.2: The cylinder you'll be tessellating has a height of one unit, and is one unit in diameter, with the y-axis passing vertically through the center. The caps of the cylinder are regular polygons with n sides (an "n-gon"), where the value of n is determined by p_2 in the demo. You will notice that each cap is cut up like a pizza with n slices. The vertices of the n-gon lie on a circle parallel to the xz plane. How will you figure out the coordinates of these vertices in terms of the radius (0.5) and the parameter θ? (What is the equation of the circle that the vertices lie on? Given a vertex on the boundary-circle of the cap, the parameter θ is defined as the angle, measured in the xz-plane, between the positive x-axis and the line segment from the center of the circle to the point on the boundary-circle.)

[1 points] 2.3: What is the surface normal of the point $(\frac{\sqrt{2}}{4}, \frac{2}{7}, -\frac{\sqrt{2}}{4})$, which lies on the barrel of the cylinder? What about an arbitary point on the barrel of the cylinder (in terms of θ)?

3 Cone

[1 point] 3.1: Look at the cone with Y-axis rotation = 0 degrees, and X-axis rotation = 0 degrees. Note that p_2 sets the number of "sides" of the cone (i.e. faces of the cone that are not the base). Observe that when $p_1 = 2$, three triangles make up one "side" of the cone. How many triangles make up one "side" of the cone when $p_1 = 3$? When $p_1 = 5$? When $p_1 = n$?

[1 point] 3.2: What are the vertex normals at the tip of the cone? This is a tricky question because a singularity does not have a normal. Observe that each of the p_2 "sides" of the cone defines a plane. You will achieve a good shading effect by creating p_2 normal vectors extending from the tip of the cone. Each of these vectors should be normal to the plane defined by the "side" of the cone it is associated with, and each of these vectors should point outward. Note that this means that there will not be a unique normal at the tip of the cone, but instead there will be p_2 different normals. Draw a simple schematic sketch illustrating the normals for all of the triangles at the tip of the cone. You may assume p_2 is any number greater than or equal to 3. As long as it is clear that you "get the idea", you will recieve full credit.

[1 point] 3.3: Consider the line L in Figure 1, which is formed by the points $(0, \frac{1}{2}, 0)$ and $(\frac{1}{2}, -\frac{1}{2}, 0)$. What is the slope of this line in the XY plane (i.e. what would the slope of this line be if we ignored the z dimension)?

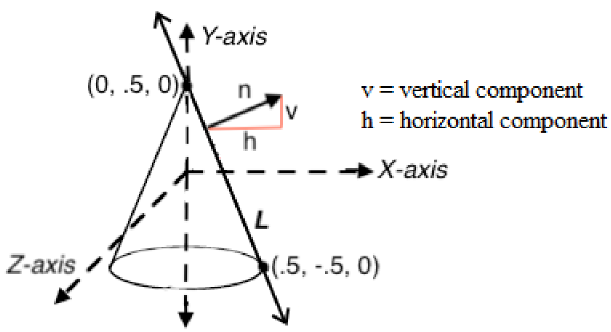

Figure 1: The line representing the sloping side of a cone. This diagram applies to questions 3.3-3.5.

[1 point] 3.4: The line L in Figure 1 represents the sloping side of the cone. Recall that a line with slope m is perpendicular to a line with slope $-\frac{1}{m}$. Using this perpendicular slope, we can find the vertical and horizontal components of the normal n on the cone body. The vertical component is the component parallel to the Y axis. Using the slope you calculated in problem 3.3, calculate the magnitude of this vertical component in the normalized normal vector n. (Your answer should be a number)

[1 point] 3.5: The horizontal component is the component parallel to the x-axis. What is the magnitude of this component in the normalized normal vector n? (Your answer should be a number)

4 Sphere

[1 point] 4.1: In Shapes, you'll need to compute normals yourself. What is the normalized surface normal of the sphere at the surface point $(r, \phi, \theta) = (0.5, \pi/2, \pi/4)$? Give your answer in Cartesian coordinates. Please fully evaluate your answer. You may find the following formulas useful.

Note that the sphere in the demo is tesselated by latitude and longitude, i.e., we make a grid of latitude-longitude values, and associate to each point of this grid a point in 3-space that lies on the sphere. The usual name for longitude is θ, and for latitude is ϕ. Longitude θ ranges over $0 \leq \theta < 2\pi$. Latitude ϕ ranges over $0 \leq \phi \leq \pi$, with $\phi = 0$ corresponding to the north pole ($y = 1$) and $\varphi = \pi$ corresponding to the south pole ($y = -1$). The conversion from spherical to Cartesian coordinates is given by

$$x = r \cdot sin\phi \cdot cos\theta$$

$$y = r \cdot cos\phi$$

$$z = r \cdot sin\phi {\cdot} sin\theta$$

[1 point] 4.2: If you start with a latitude-longitude grid with θ ranging over $[0, 2\pi]$, and then convert to spherical coordinates, the points of your grid with $\theta = 0$ and those with $\theta = 2\pi$ will end up being at the same location in 3-space. That's not so bad for texturing, because you can use θ and φ as texture coordinates. It's terrible for things like 3D printing, where it can be very difficult to prove that the resulting mesh is "watertight": to the analysis procedure, it appears possible that there could be leakage between two identical line segments on the Greenwich meridian. If you have n different θ values and k different φ values, you might, in the basic form, create nk vertices, numbered $0, 1, ..., nk - 1$, with the first k vertices being the ones for $\theta = 0$, and the last k being the ones for $\theta = 2\pi$. In the fancier form, you'd have only $(n-1)k$ different vertices, labelling the last k vertices in your diagram not $nk - k, ..., nk - 1$, but instead, labelling them $0, ..., k - 1$, so that they are the same as the first k. Do a 3×4 example, and write out all the triangle-triples as we did in class, to see how this works out. You might think that you'd need to carefully do things like this in making the sphere for the Shapes assignment, but you do not. In fact, you can make a cube, for instance, out of six square panels, and the locate them so that matching edges happen to line up. From a "watertightness" point of view, this is a complete mess! But from an appearance view, which is what Shapes is all about, it's just fine. Thus this exercise is here to help you understand that things in general are not simple, but for this assignment, they actually are!

5 Design

[1 point] 5.1: In your Shapes project we will be providing less project structure than in Brush; all we've included is a OpenGLShape class and an ExampleShape that is rendered immediately as part of the stencil code. You can feel free to refactor this as much as you see fit.

Regardless you'll want to take advantage of polymorphism in some way. All of the shapes likely share some functionality but certain shapes such the cone and cylinder share specific components such as the circle base that you may want to reuse. How will you use inheritance, composition, or both in Shapes to minimize redundant code and apply good software design?

There is no single correct answer and there are good solutions that use only inheritance, only composition, or both. This question is just to get you thinking about how you plan on structuring your Shapes project.