Objective

This project is about modifying surface meshes. In particular, you will write a program in the Python version of Processing that reads in polyhedral models and creates geodesic surface using triangle subdivision. Your program will also demonstrate the "corners" representation by visualizing corner operations such as next, opposite and swing. All of the mesh objects that you create and modify will be made of only triangles.

More Flexible Late Policy

For this project only, you can turn in the project up to five days late. Each late day will incur a 3-point penalty instead of the usual 5-point penalty. This late policy does not apply to any of the other projects in the course.

Project Description

The provided code draws a single triangle on the screen. This triangle can be rotated by clicking and dragging the mouse in the window. The provided code also reads a mesh from a file and prints this information on the screen.

Your finished program will be able to read a mesh from a file and display that mesh. More importantly, with a keystroke your program will create a subdivided version of the mesh, replacing the old mesh in memory. Your program should also show corners adjacency information by visualizing a current corner as a small sphere. Your program will respond to the following keystroke commands:

1-4: Read in a mesh file (tetrahedron, octahedron, icosahedron, star).

d: Create the triangle subdivided current mesh (you should be able to do this more than once).

i: Inflate the points of the mesh so that they lie on the unit sphere.

r: Toggle between white and randomly colored faces.

c: Toggle between showing and not showing the current corner as a sphere.

n: Change the current corner using the "next" operator.

p: Change the current corner using the "previous" operator.

o: Change the current corner using the "opposite" operator.

s: Change the current corner using the "swing" operator.

The subdivision that you will implement does not actually produce what is known as a subdivision surface. What you

will implement is slightly less complex than making a subdivision surface, because the rule for placing the new vertices on edges is simple, and because you will not be moving the original vertices. You will place the new vertex for an edge at exactly the midpoint of the edge. For each original triangle, you will use its three original vertices together with the three new vertices at the edge midpoints as the vertices of four new triangles that replace the original triangle.

A large part of this project is being able to perform triangle subdivision of the surface more than once, so **make sure** your program can do this.

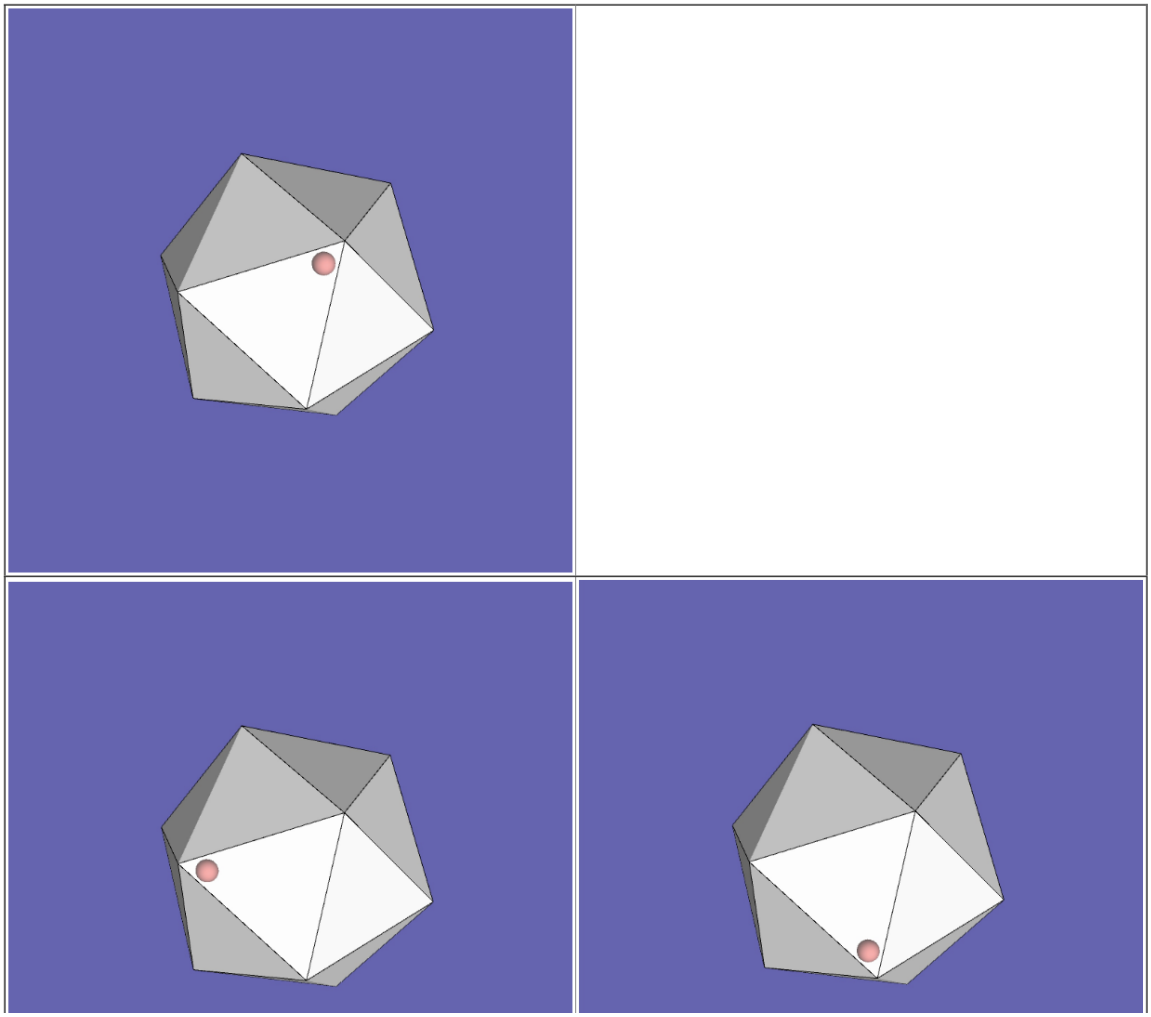
You should initialize the "current corner" to be zero. The display of the current vertex should be toggled by the "c" key. Other keys will change the current vertex to be other corners, through the next, previous, opposite and swing operators. Be sure that the sphere that represents the current corner is clearly at the **corner** of one particular triangle. It is **not sufficient** to draw a sphere at a vertex, since this doesn't show which triangle the corner belongs to. Remember that the swing operator (key press "s") should move the current corner around a vertex, and that this motion should be visible as you travel between corners.

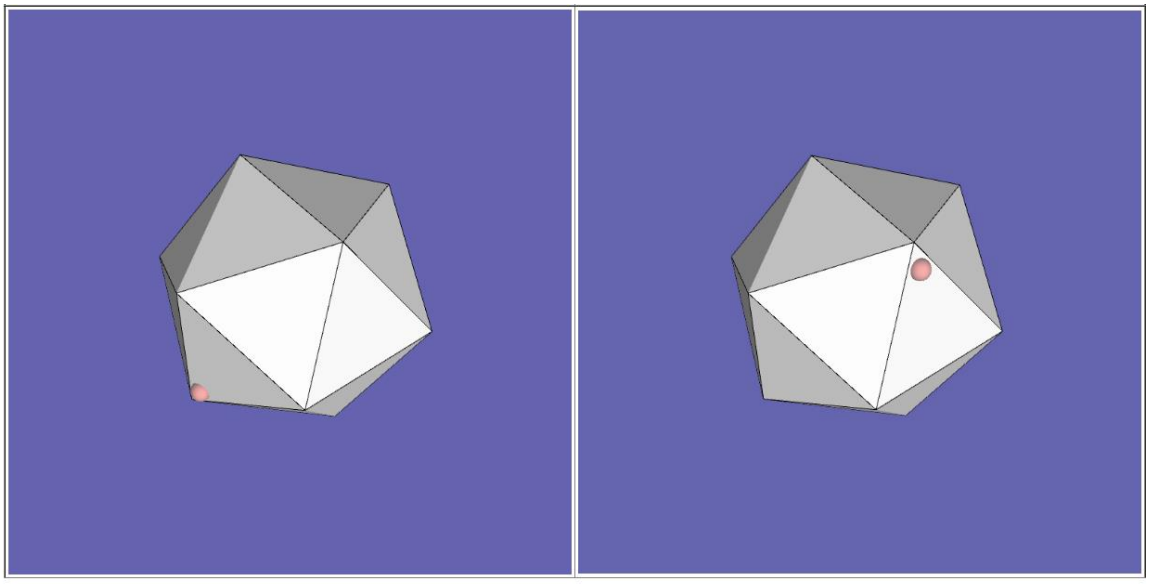
When random colors are turned on, the faces of your model should be colored at random. These random colors should not flicker, but instead should remain steady from one frame to the next.

Please make sure your project has the ability to rotate the current mesh by clicking and dragging with the mouse.

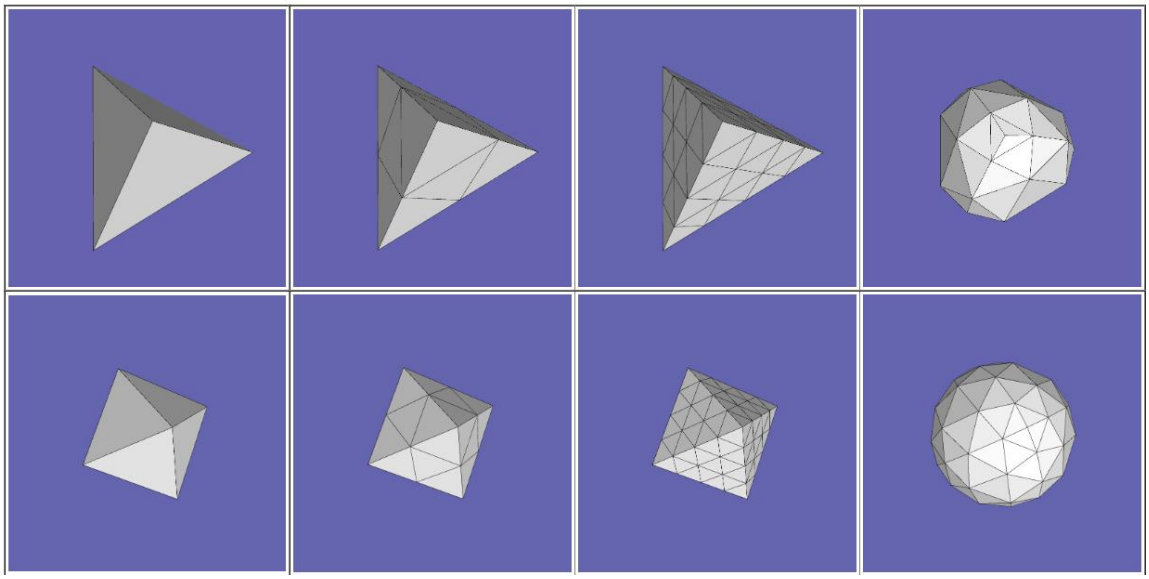
Results

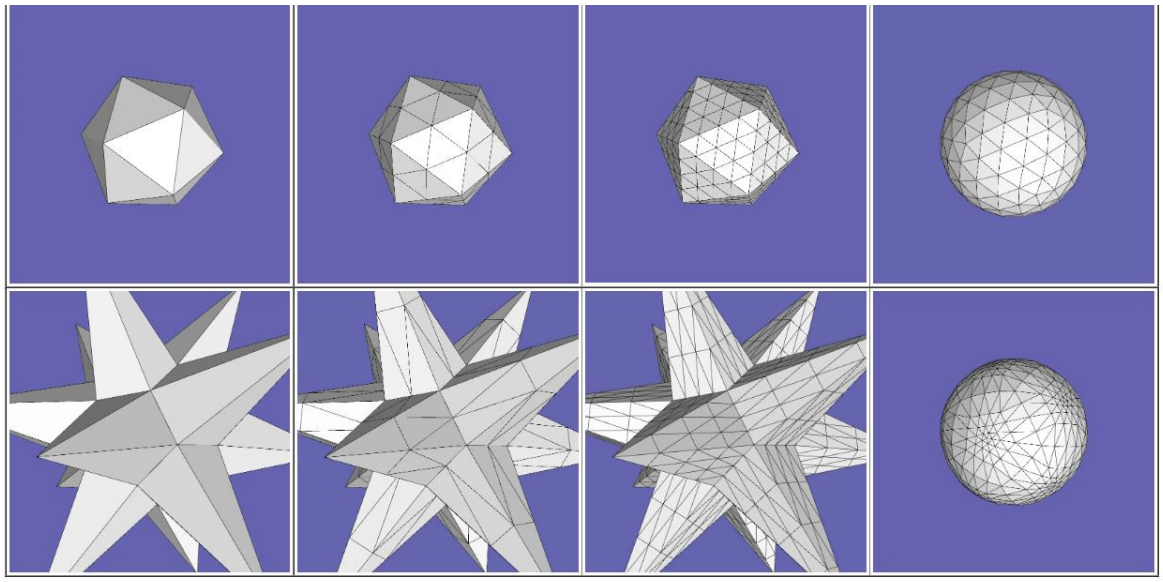
Below are images that show how you should visualize the corners operators. In the top row, one corner is illustrated by placing a small sphere in the corner. The remaining four images show where this corner would be moved by each of the following operators: next, previous, opposite, and swing, respectively.





Each line below shows an original object, the object subdivided once, subdivided twice, and subdivided twice and inflated.





Below shows a subdivided icosahedron, without and with randomly colored faces.