Check-in summary

We decided to start by implementing Tutte embedding of 3D models to 2D textures, using starting code from assignment 2. We referenced [Floater, 2002], some lecture slides from the University of Columbia, and [Floater, 2011] to figure out how to create a Tutte embedding.

Coding this, we realized that we can create a better Tutte embedding than what we have to make the Tutte embedding more general, and give us an API to modify vertices so that it will be easy to code up the optimized texture map algorithm.

Going forward, we hope to:

- make our current project read in obj files more robustly, so we can use more sample models
- Replace the vector library with eigen so we can handle arbitrary matrix dimensions
- Come up with a way to automatically read in the boundary vertices
- Get our hands on harder obj models (hopefully with a defined seam already)
- Start focusing on bijective parameterization

Once these goal are finished we hope to start optimizing the algorithms and create a UI so users can see the changes from 3d mesh to tutte parameterization to bijective parameterization.

The example screenshot below was created using parts of assignment 2. We got rid of lighting, stopped the mesh from drawing, and drew edges instead. We chose arbitrarily one of the faces as the 2d boundary and mapped those vertices z-values to 0. We then repurposed the transformation and joint drawing code to create a 3d wireframe (on the left) and the resulting tutte parameterization (on the right).