

COMP 5411 : Advanced Computer Graphics
Fall 2020
Getting Ready for Programming Assignments
Not to be submitted

You are given a code framework written in C++ and GLFW that loads a mesh into a half-edge data structure and provides basic viewing interactions. You can build your code upon this framework.

Vertex normals and shading. To get acquainted with the half-edge data structure, please compute the vertex normals for smooth shading. The normal at a vertex is computed as a weighted sum of the normals of the facets surrounding the vertex. There are several weighting methods that consider angles and areas. You may simply use the area of facet as the weight. You only need to implement the compute `VertexNormals` function and the mesh shader will handle the other things.

Geodesics on Meshes. For those who want more practice or challenge, you may try calculating geodesic distance between two query points. Geodesic distance between two mesh vertices is the length of the shortest path along the surface that connects the two vertices. As mesh is in fact an undirected graph, Dijkstra's Shortest Path algorithm can be used to find geodesic distances. You can print the $N \times N$ geodesic distance matrix M to a file. Visualize the shortest path between two query points by redrawing the edges on the path with thicker red lines.

For compilation instructions, please refer to the handout **OpenGL Intro** (https://course.cse.ust.hk/comp5411/ogl_beginner/OpenGLIntro.pdf). On Windows, the given source code can be compiled using Microsoft Visual Studio 2015 or higher editions.

Please refer to the supplemental slides for more details. You may contact the TA if you have any questions.