Report: Computer Graphics **Xucheng Chen**
COMS 4160 xc2360
Prof. Zheng Date: 02/23/2017

Running the code

I basically finished all coding in eclipse, so the easiest way to compile and run the code would be just import the code in eclipse and then configure the build path to include the lwjgl library. After importing the project, just run Main.java:). If you have any problems running the sample code, please contact me and I will personally demo it in office hours.

Implementation

0.1 Mesh to Half Edge

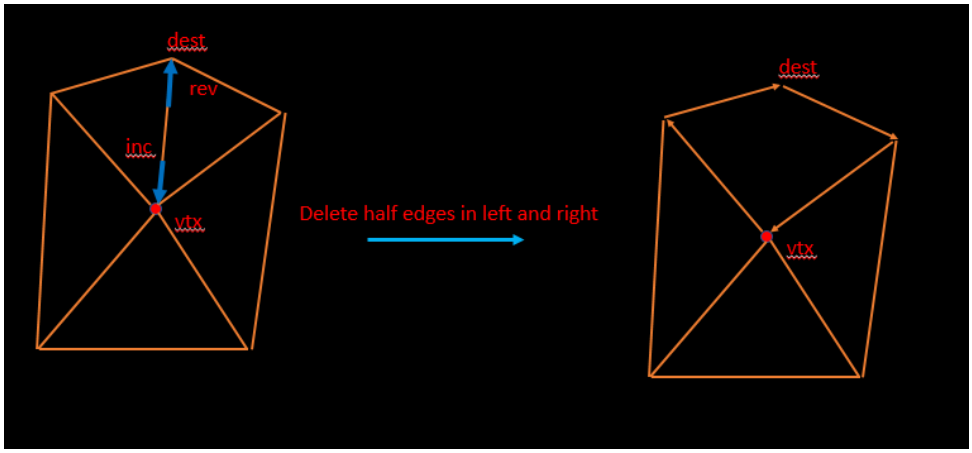
for each face:

- create three vertexes using function createVertex(), which will assign normals, positions and vertex index to each vertex
- create three halfedges and link it with corresponding vertexes
- create face with face index and link with half edges
- for each pair of vertexes, for example v1 and v2, sort them and use 'v1-v2'(if v1 is smaller) as the key for hash map, the value of the map would be index of two adjacent faces

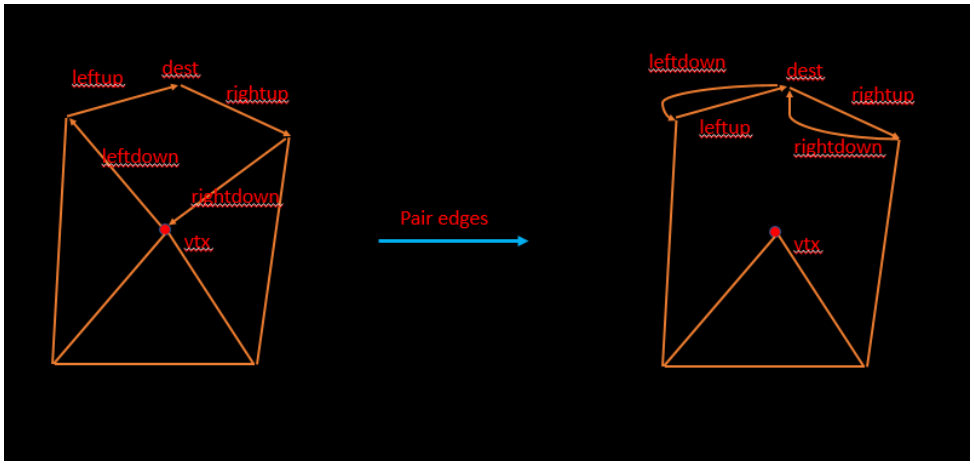
after processing all faces, we need to add reverse edges for each half edge, which is completed using function addRevEdge()

0.2 Remove vertex

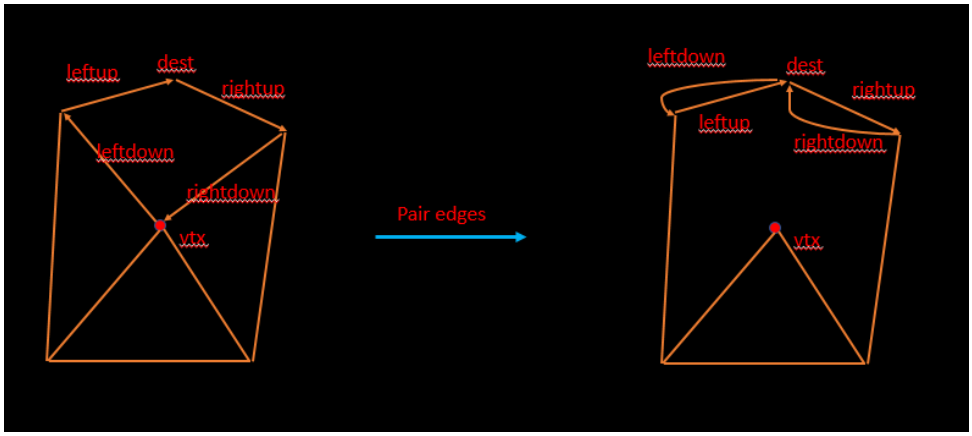
- get all half edges pointing to vertex and save it in an array
- delete all half edges belonging to the two adjacent faces of half edge 'inc', which is labelled left and right on the graph



- set the reverse edge of leftup to leftdown, set the reverse edge of rightup to rightdown

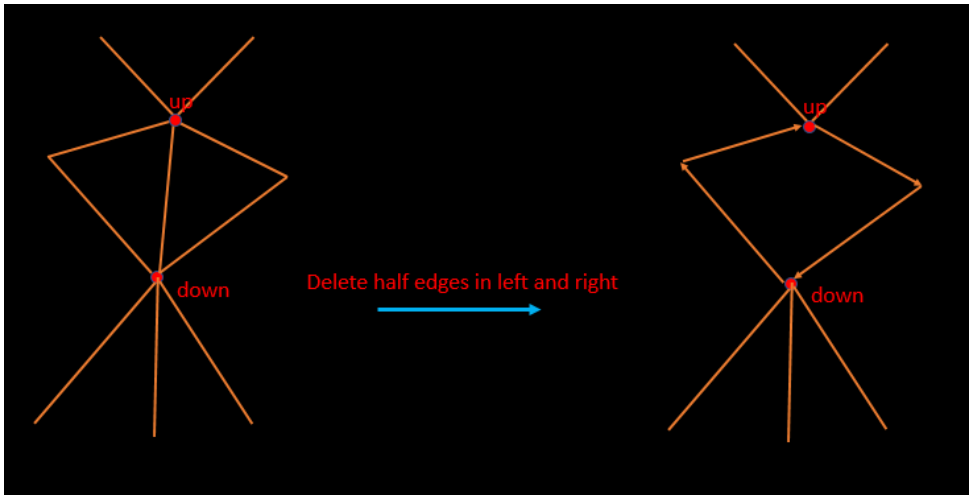


- for the rest edges pointing to vertex, change their vertex to 'dest'

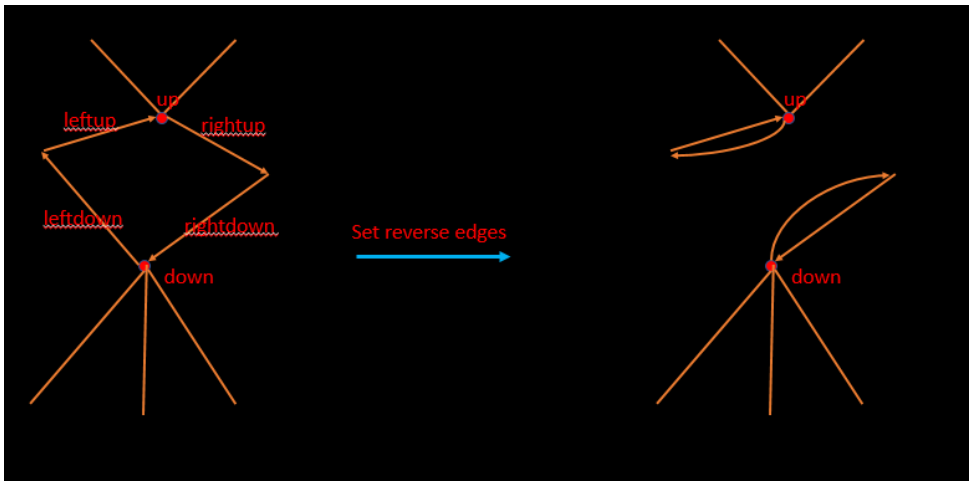


0.3 Collapse edge

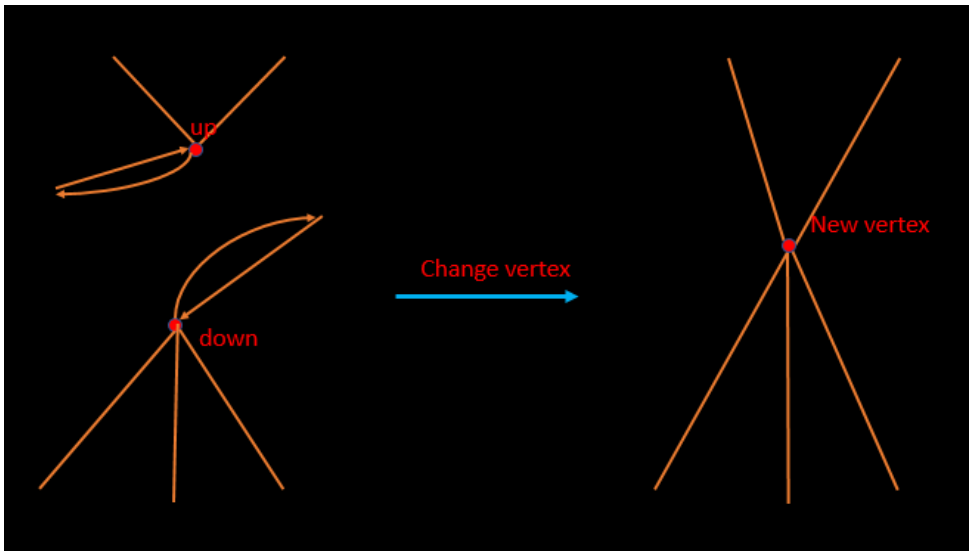
- delete all half edges belonging to the two adjacent faces of half edge 'inc', which is labelled left and right on the graph



- set the reverse edge of leftup to leftdown, set the reverse edge of rightup to rightdown



- for the rest edges pointing to vertex 'up' and 'down', change their vertex to new vertex



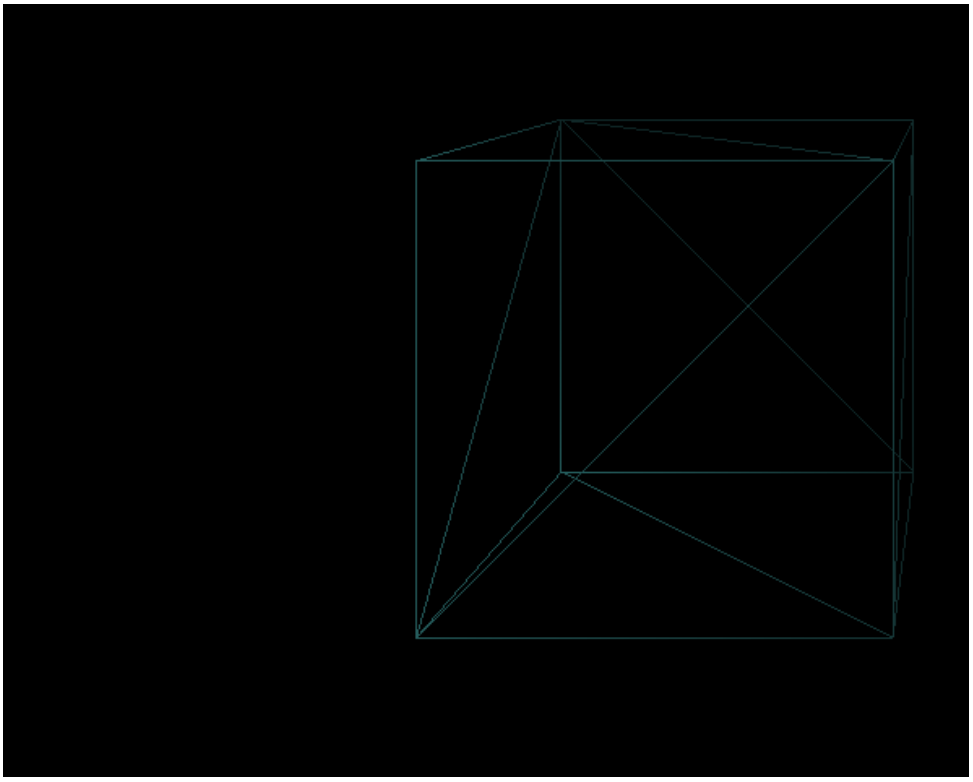
0.4 Half Edge to Mesh

see the code for explanation, which is quite straightforward

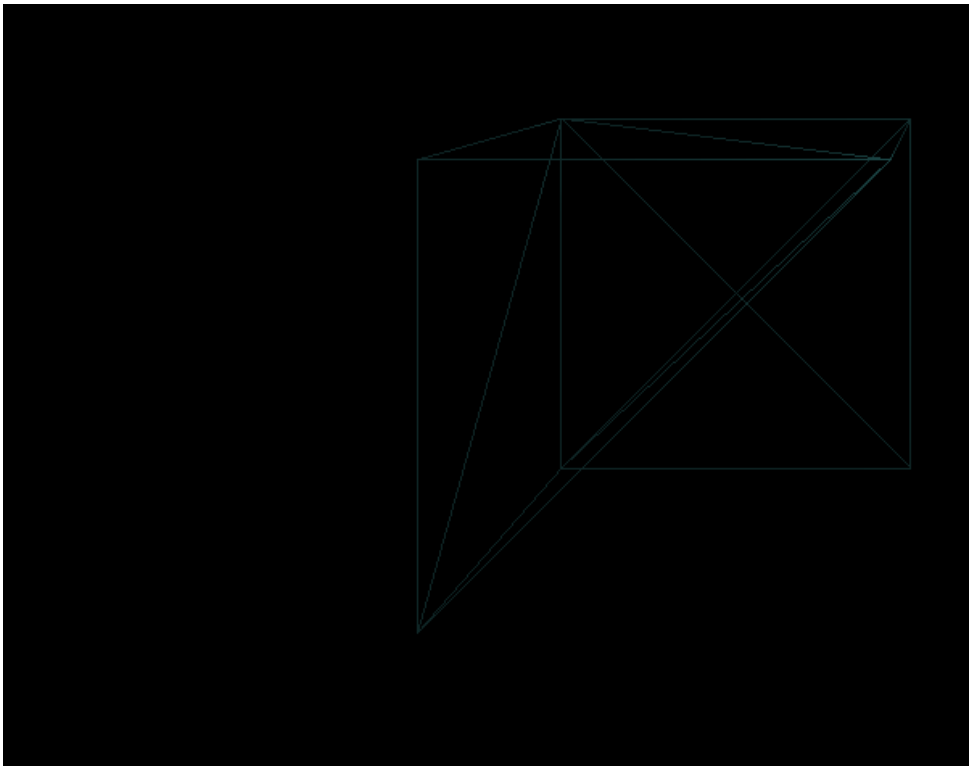
Result

0.5 Remove vertex

0.5.1 before

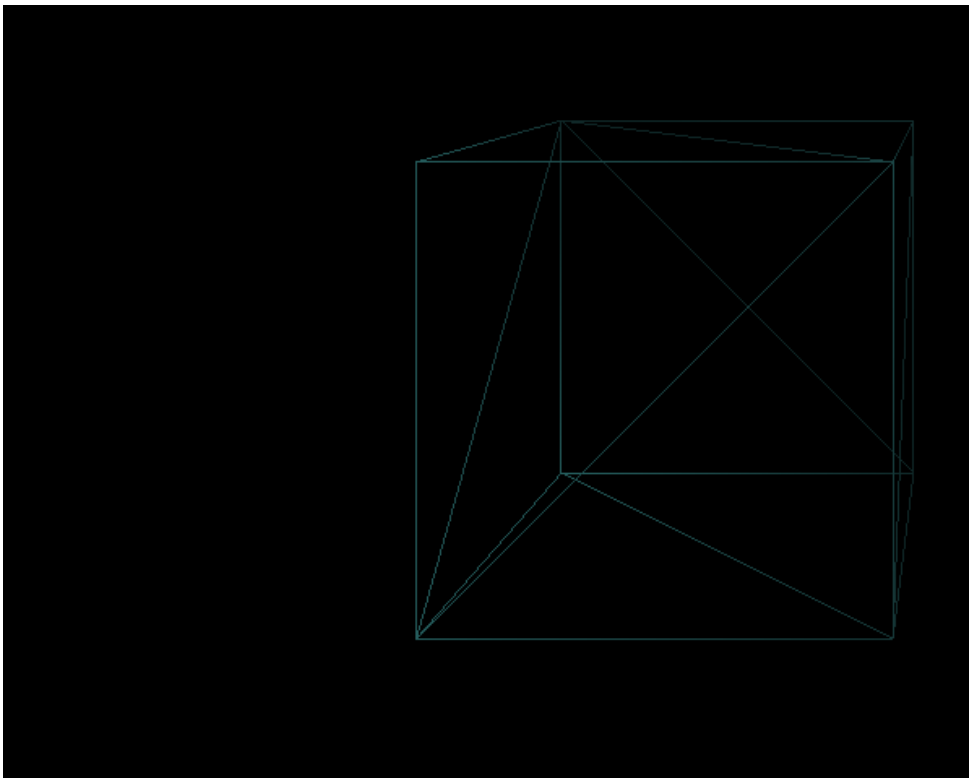


0.5.2 after

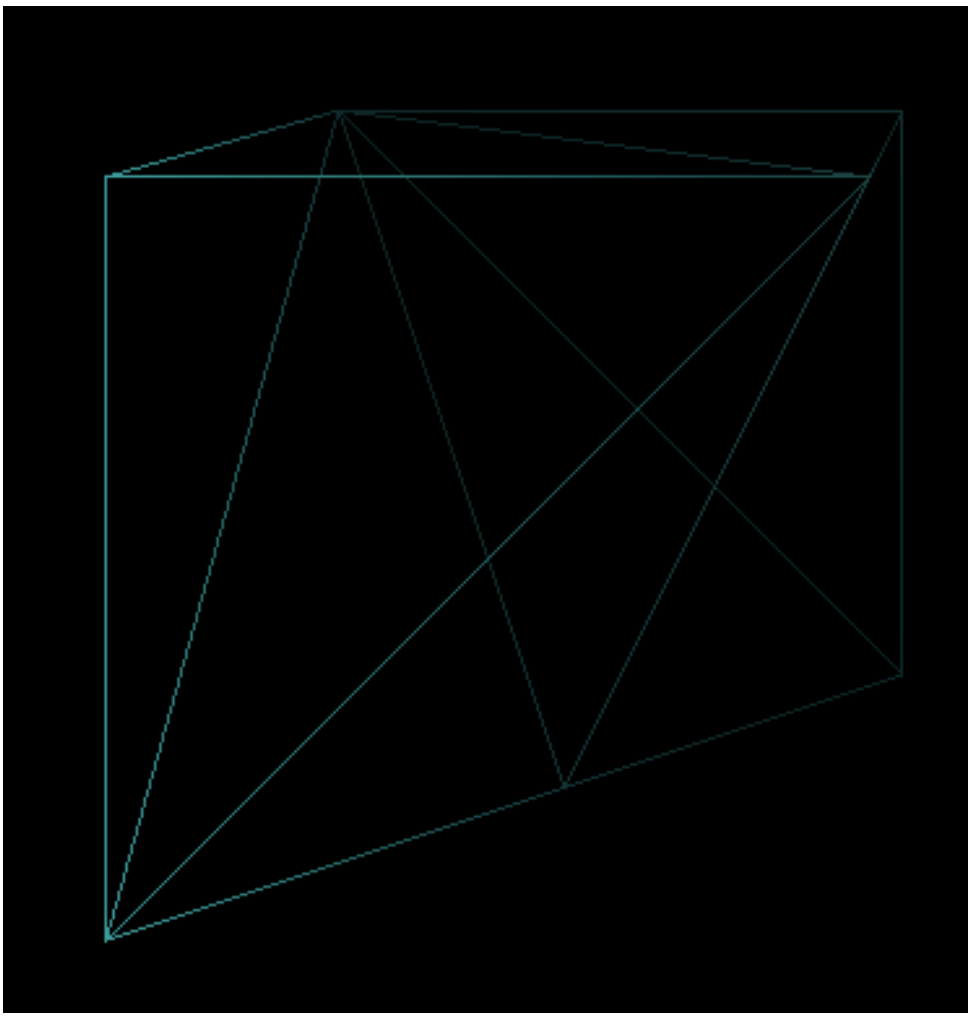


0.6 Collapse edge

0.6.1 before



0.6.2 after



Installiation

The following installation guide is copied from the official website of lwjgl: Eclipse supports Gradle/Maven projects and it is highly recommended to use them instead of a manual Eclipse project configuration. However, if you prefer setting up a native Eclipse project, follow these instructions (works with Eclipse Neon):

- Download the ZIP bundle from https://www.lwjgl.org/download
- When the download is complete, extract its contents to some file system directory, henceforth referred to as <lwjgl3>.
- In Eclipse go to menu "Window" > "Preferences" and in the tree view to the left search for 'Java' > 'Build Path' > 'User Libraries' and Click "New..." in the 'User Libraries' dialog. In the opened modal dialog "New User Library" write "LWJGL3" in the 'User library name:' text field and click 'OK'. The newly created library "LWJGL3" should show up in the list 'Defined user libraries:'.
- Now, select this item in the list and click 'Add External JARs...'. This opens a standard OS file selection dialog allowing you to select *.jar files which get added to the class-

path/buildpath of all projects using the LWJGL3 User Library. Go to <lwjgl3> and select all *.jar files which do not have -javadoc or -sources in their names. Make sure you don't forget the lwjgl-natives-<os>.jar file, and click 'Open'. This will populate the LWJGL3 user library in the list with respective entries for all selected jar files. You could leave it at that now in order to use LWJGL 3.

- However, if you want to have Sources and JavaDocs, you will have to select each of the entries, click on 'Source attachment: (None)' and on 'Edit...'. This will open the "Source Attachment Configuration" dialog. Here you could select 'External location' and 'External File...' to select the appropriate *-sources.jar file.
- In order to actually use the LWJGL3 User Library in one of your projects, go to the Build Path settings of your project and select the 'Libraries' tab. Here, click 'Add Library...', select 'User Library' and mark the "LWJGL3" User Library.
- Now you are set up to use LWJGL 3 in your project.