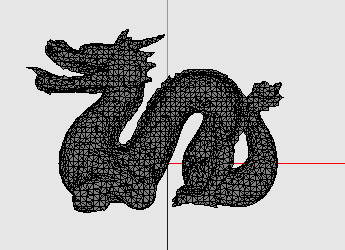
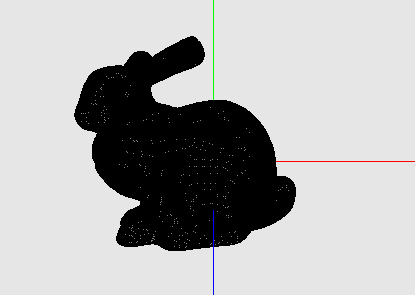
Loading a Shape File

**Description:**

You’ve learned that we can draw polygons to the screen by plotting them out one at a time, or even using mathematical functions to draw nice curves and geometric patterns. Often times to create more complex shapes, 3D objects will be scanned from the real world by lasers or modeled in 3D software. This data is then output in a 3D file format that can be read.

**Your Task:**

* You will be loading a file known as the PLY format. This format consists of a, header, vertex list, and a face list.
* Start by looking at the headers given to familiarize yourself with the functions provided. You will be implementing the loadgeometry() member function for the shape class located in the shape.cpp file.
* Center the object at the origin, and scale the object to the unit size.

**Files Given:**

main.cpp – Main program interface (you do not have to modify this)

shape.cpp – The shape class that loads the PLY model

shape.h – The shape header file with data (start here)

geometry.h – Some data structures to store geometry information (you do not have to modify this).

dragon and bunny.ply – These are the files you will be loading. Take a moment to load it and familiarize yourself with this text-based 3D file format.

**Compiling: (On the Mac)**

g++ -Wall -Wextra main.cpp ply.cpp -I/Library/Frameworks/GLUI.framework/Headers/ -framework OpenGL -framework GLUT -framework GLUI -o ply

**Running:**

./ply

**Provide:**

comp 175 \_\_\_\_\_ list out all files

**C++ Refresh -- Helper functions:**

Opening a File

1. #include <fstream>
2. ifstream myfile (“filepath”);
3. if (myfile.is\_open()){
4. string line;  
    while(getline(myfile,line)){
5. // some code
6. }
7. }

Parsing a line

1. #include <stdio.h>
2. char\* delimeter\_pointer;
3. delimeter\_pointer = strok(line,” “)
4. while(delimeter\_pointer != NULL){
5. cout << delimeter\_pointer;
6. delimeter\_pointer = strtok(NULL,” “);
7. }

**Example PLY File** [1]:

ply

format ascii 1.0 { ascii/binary, format version number }

comment made by Greg Turk { comments keyword specified, like all lines }

comment this file is a cube

element vertex 8 { define "vertex" element, 8 of them in file }

property float x { vertex contains float "x" coordinate }

property float y { y coordinate is also a vertex property }

property float z { z coordinate, too }

element face 6 { there are 6 "face" elements in the file }

property list uchar int vertex\_index { "vertex\_indices" is a list of ints }

end\_header { delimits the end of the header }

0 0 0 { start of vertex list }

0 0 1

0 1 1

0 1 0

1 0 0

1 0 1

1 1 1

1 1 0

4 0 1 2 3 { start of face list }

4 7 6 5 4

4 0 4 5 1

4 1 5 6 2

4 2 6 7 3

4 3 7 4 0

**Going Further:**

Did you enjoy this in class assignment? Take a look at some other 3D formats such as .md2(includes basic animation data). Look at some popular 3D modeling software like Maya, Blender3d (download for free!), and check out there file formats.

**References:**

[1] http://paulbourke.net/dataformats/ply/