CS 130
Assignment 2

"Ray-tracer smooth or flat triangle renderer"

Your program will render a polygonized, triangle-based model read in from an input file like the first project. Test models provided from Assn 1, but your program should be general to work with other triangle-based models and will be tested to show that it can work on other models during grading. Your project is to interpret the data and render the model parsed as follows.

The rendering tool will use two modes to draw the model. The starting/default mode is a flat shaded render and the second is smooth shaded, using Phong smoothing to render. The flat and smooth ray-tracer modes will follow from the lecture description. The illumination should follow the Phong lighting model discussed in class and should contain diffuse, specular highlighting, and ambient components.

The ray tracing algorithm must include shadows that happen based on one polygon shadowing another. The shadow follows along with the ray-tracing as described in the lectures.

Implementation:

You should start from the OpenGL test code provided in lab. To be clear, you will be use OpenGL only to turn pixels on using glVertex. This is the only OpenGL call your code should make to render.

My suggestion is that your algorithm follow the pseudo-code given in class for ray-tracing. You may "hard-code" information about the light source(s) as in position(s) and intensities. A default material can also be hard-coded.

A simple background color (maybe, black or white?) should be set when a ray doesn't hit any objects in the scene. To help you pick your camera angle (scene set-up), we provide test objects (Assn 1) of standard sizes that span the volume(s) expected for the other objects.

For the keyboard interface, a simple switch for the render modes is controlled by the "x" button press.

HINTS: This program may run slowly. To speed up the debug process, test a simple model (a few triangles) and use a small test image (like 50x50 pixels!) until you are certain the basic parts work. Then make the test image larger (now 100x100) and add the shadows, etc. Finally, make the image more reasonably sized (500x500) to make/set the scene and don't do the full blown image size until the very end. Of course, *do* test the full resolution before you turn it in and **submit a snapshot of the image along with your code.**

Credit will be broken down as follows, rendering the polygon model flat shaded with phong (specular, diffuse, ambient) lighting is 45 pts, and adding shadows (15 pts), smooth shading (30 pts). And the last 10 points is for neat code and documentation. Your program should run on any of the machines in lab, using the commandline string: "render modelname.txt" where modelname.obj is the model to be rendered.

Want more?
Okay, you will be given 20 extra points for adding a bump map over the polygons. For more detail, see the TA

or instructor. This is not for the weak of heart and will require some extra parsing or reorganization of the model. It will be graded all or nothing. More details are available if you want, just ask.

Good Luck!