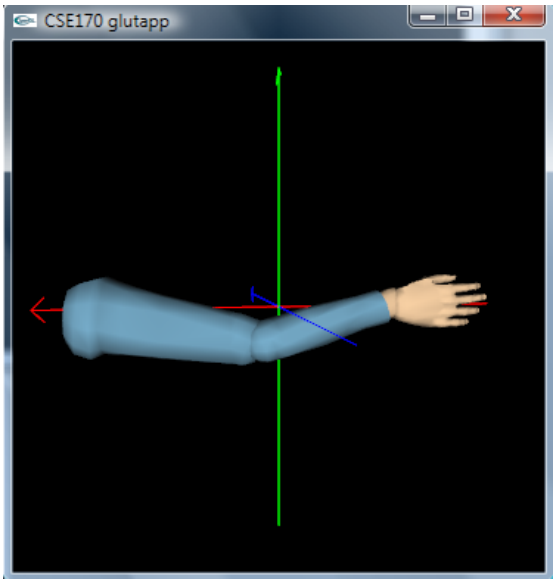
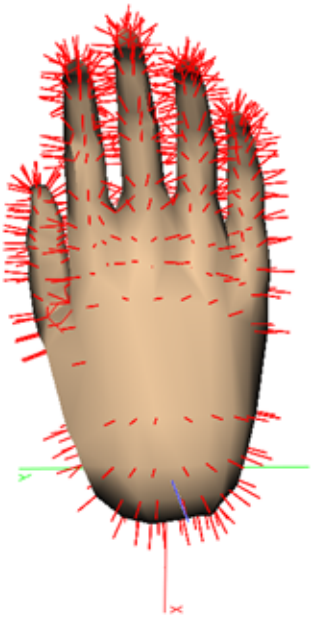
Programing Assignment #5

In this assignment you will load and draw an arm model in wireframe, flat, and smooth shading. The arm consists of 3 parts: the hand, the lower arm, and the upper arm. Each of these parts has a corresponding model file (.m) that represents the geometry of the part. The model files are provided to you along with support code that can load the information stored in the files.



The Three Models of the Arm Correctly Positioned



Hand Model Smooth Shading Normals

Hand Model Flat Shading Normals

The structure of the model files is very simple. It contains a list of vertices, faces, colors and normal vectors. You are not required to understand the structure of the .m file, however, you will have to understand the Model class provided as support code in order to accomplish your tasks.

Download armmodel.7z from the Support Code folder. Inside the package you will find the following files: rhand.m, rlowerarm.m, rupperam.m, model.h and model.cpp. Put all the files in the directory of your source files and add the .h/.cpp files to your visual studio project.

For this assignment you have to accomplish the following:

Requirement 1 (20%) – Correct Display and Positioning of the Arm Models

Create a SoModel class in order to load and manage the OpenGL arrays for each model you will load. To render the entire arm correctly you will have to draw three models. Therefore you will need three instances of SoModel, one for each arm part to be loaded. Each object was modeled with respect to its own local frame. Use your own transformation matrices to place the parts in the correct locations such that they look like a real human arm, as you can see in the left-most picture above.

Requirement 2 (30%) – Rotations and Controls

After the models are correctly positioned at the right locations to form the entire arm, you will then apply rotations at the joints. Compute and apply transformations such that you achieve correct rotations at each arm articulation:

'q' and 'a' - will rotate the shoulder articulation up and down
'w' and 's' - will rotate the elbow articulation up and down
'e' and 'd' - will rotate the wrist articulation up and down

Note that your rotations have to lead to a correct and meaningful control of the arm; for example, when the shoulder is rotated, the entire arm correctly moves together with the shoulder. To achieve this you will need to combine rotations and translations in the correct order, and correctly accumulate them with respect to each model.

Requirement 3 (40%) – Rendering Options

The final requirement is to render your models in three modes: flat shading, smooth shading, and wireframe. When rendering in flat shading or smooth shading, you will also need to have the option to display the normal vectors sent to OpenGL or not.

As before, leave the arrow keys to rotate the entire model for visualization and use the keys indicated below to change the visualization options:

'z' – Wireframe rendering	(5%)
'x' - Flat Shading rendering	(10%)
'c' - Smooth Shading rendering	(10%)
'v' – Normals on/off visualization	(15%)

When you draw the normals, be sure to draw in the following way:
a) Flat shading: place each normal vector at the center of each face, pointing outwards.
b) Smooth shading: place each normal vector at the vertices (it should appear pointing outwards)

Note that you will only need to compute normals for flat shading because the normals for smooth shading are already provided by the .m files.

For achieving wireframe rendering, you may just make a call (before drawing your objects) to glPolygonMode with parameter GL_LINE. For wireframe rendering you can choose to use flat or smooth normals; each will achieve different results (flat or smooth) but both will correctly render in wireframe mode.

Overall PA quality (10%)

Just do a nice project with a well organized and well implemented code.