

Homework 1: Read and Draw a 3D Surface Mesh

Write an OpenGL program to read and display a 3D mesh (in .obj format).

1. Create a “Windows Console” project, include “HW1_Starter.cpp”.
2. Implement the “`bool ReadOBJFile(const char filename[])`” function, to read in the obj file and store all the elements in a list of vertices and a list of faces.
3. Implement the “`ComputeBoundingBox()`” function, here,
 - a. Compute the mass center $M(x, y, z)$, which is the average position of x , y , and z coordinates of all vertices. Move the object center to the origin, by subtracting M from every vertex’s coordinates.
 - b. Compute the bounding box $B = [x_{min}, x_{max}] \times [y_{min}, y_{max}] \times [z_{min}, z_{max}]$ of this 3D object. Compute its diagonal axis length l .
 - c. Now you can set the camera position to $(0,0,1.5l)$.
4. Implement the “`Render_Mesh()`” function to finish the OpenGL rendering. Here you just need to go through all the triangles, and draw them one by one.
5. Finish the rotation (mouse left button), panning (mouse middle button), and zooming (mouse right button) functions.

Note 1: I have attached two “OBJ” meshes and an “hw1.exe” program. Your program should work like that.

Note 2: Please write all your codes in one “hw1.cpp” file. Test and make sure it compiles and renders correctly, then upload your “hw1.cpp” file only. I will compile and run it on my computer to see your result.

Note 3: You only need to consider the simplest OBJ format in this homework. It contains a list of vertices (each row starts with a keyword “v”, e.g., `v x y z`), and a list of faces (each row starts with a keyword “f”, e.g., `f vInd1 vInd2 vInd3`). Note that the vertex index `vInd` starts at 1. So if you have stored vertex positions in an array, the first vertex is at position 0, and each vertex’s location is `vInd-1`.