GLSL shaders

Due date: Monday, May 16

In this assignment you will implement a vertex and a fragment shader. A basic framework for loading and rendering a mesh using shaders is provided on the class web page. You need to install the `GLEW` library to compile this code.

1 What your program should do

Vertex shader. Write a shader that (a) computes all necessary information for the bump-mapping fragment shader (below); creates a periodic blend between the original surface and the surface displaced in the normal direction by an amount depending on the surface point position, $c \sin(x) \sin(y) \sin(z)$, where c is a scale you choose based on the mesh size (1/100-1/20 of the mesh boundinx box size is reasonable).

Fragment shader. Implement a bump map; rather than loading the bump map in the texture, implement a procedural generation of the normal variation following the brick texture in the GLSL example; your normals should coincide with the surface normals on the part of the surface corresponding to brick, and create an appearance of a V-shaped groove in mortar areas.

2 What to turn in

The source code (C and shaders) and a working executable of your program.