

Lab04 Mesh Manipulation

Task 1: Construct and render a plane (x-z).

Task 1.1: construct a plane (x-z)

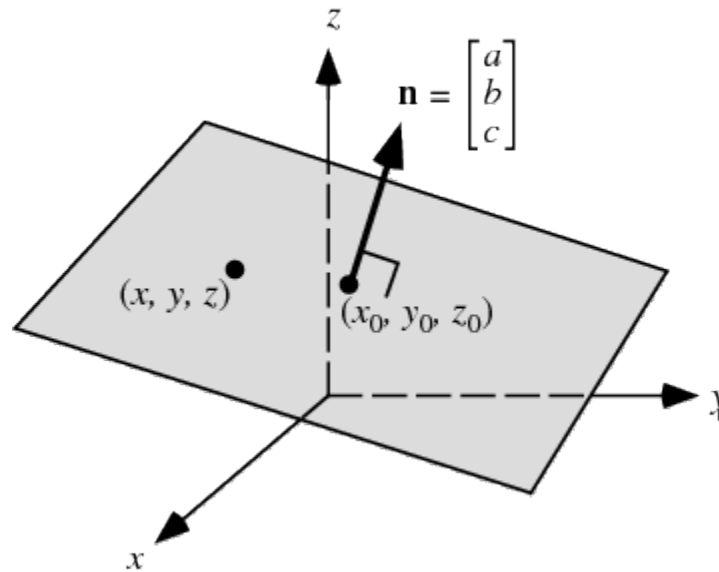
Task 1.2: render the plane

Task 2: Translate and rotate the plane using the keyboard.

Task 2.1: translate the plane in $\pm y$ direction using the keyboard

Task 2.2: rotate the plane around the z-direction

Task 3: Assign different colors to the model vertices based on their position with respect to the plane (below or above).



The equation of a plane with nonzero normal vector $\mathbf{n} = (a, b, c)$ through the point $\mathbf{x}_0 = (x_0, y_0, z_0)$ is

$$\mathbf{n} \cdot (\mathbf{x} - \mathbf{x}_0) = 0, \quad (1)$$

where $\mathbf{x} = (x, y, z)$. Plugging this point into general equation of a plane gives

$$a x + b y + c z + d = 0, \quad (2)$$

where

$$d \equiv -a x_0 - b y_0 - c z_0.$$

To check if a vertex is above or below the plane, evaluate Eq. (2) and if the result is greater than zero then the point is above the plane, else it is below.

Task 3.1: Calculate the coefficients (a, b, c, d) of the plane.

Task 3.2: Change the color of the fragments according to the position of the vertex with respect to the plane.

- a) Task 3.2a VS: propagate vertex coordinates (world space) to fragment shader
- b) Task 3.2b VS: calculate vertex position in world space
- c) Task 3.2c FS: get vertex position from VS
- d) Task 3.3d FS: get the coefficients of the plane from main program (uniform vec4)
- e) Task 3.2e FS: find on which side of the plane is the vertex and apply different colors (red above 0.02, blue below -0.02 else blue)

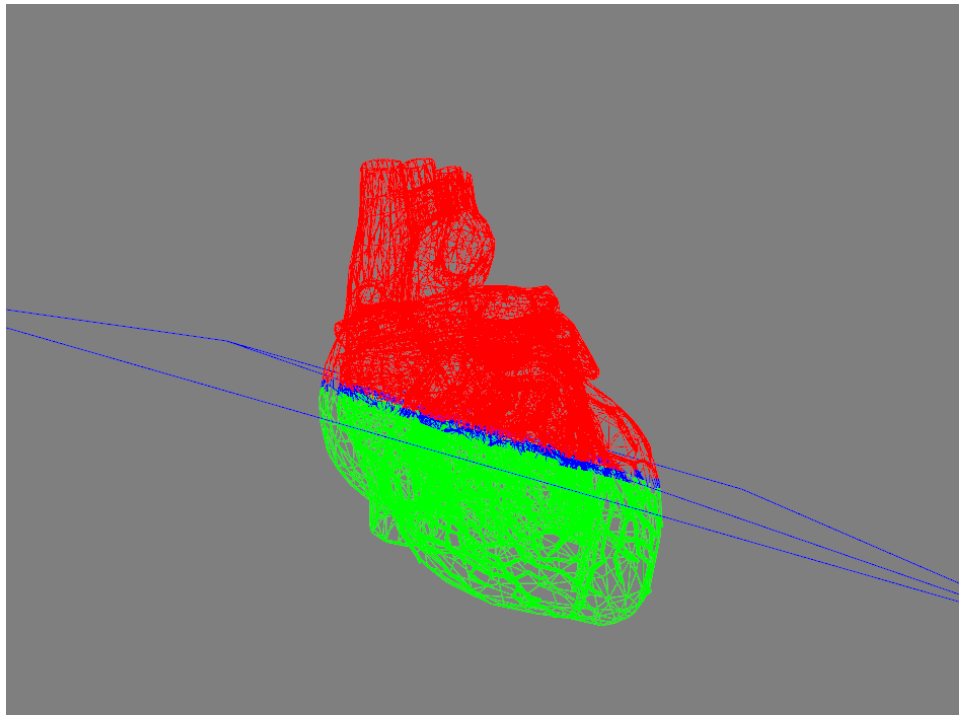


Figure 1 Task 3.2

Task 3.3 FS: color the model based on the vertex position and make the lower part vanish partly (use alpha).

```
// Task 3.3: blend must be enabled
glEnable(GL_BLEND);
glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA);
```

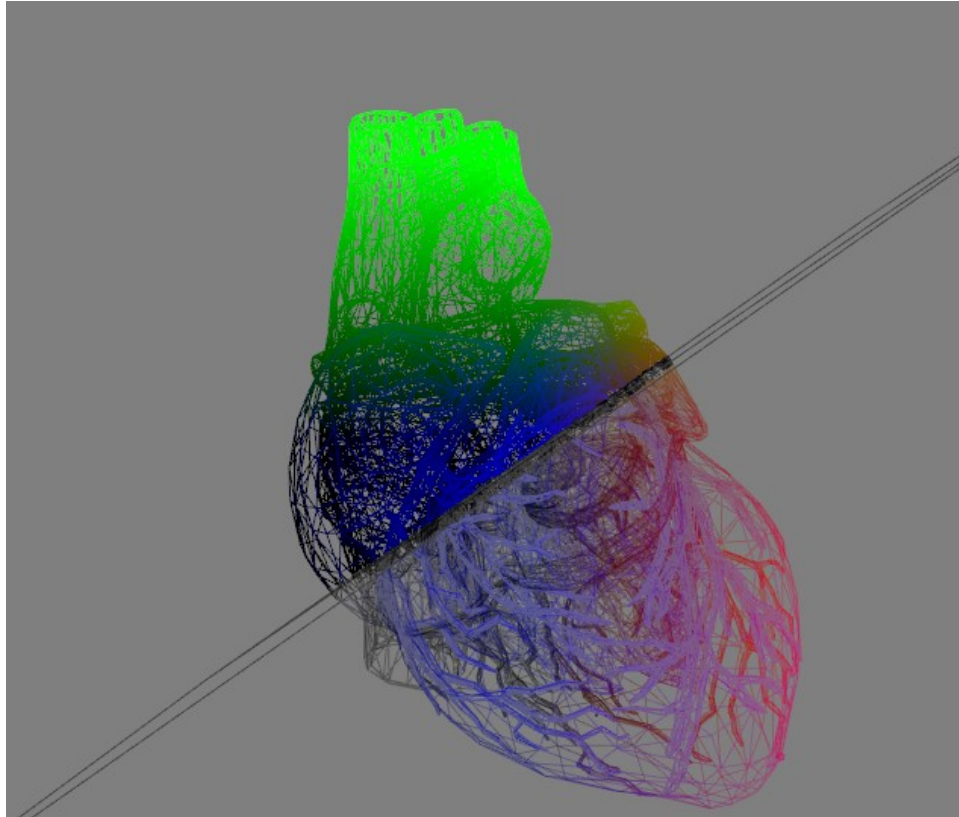


Figure 2 Task 3.3

Task 3.4: add a wireframe toggle action that will check the current polygon mode and change to either FILL or LINE.

```
GLint polygonMode[2];  
glGetIntegerv(GL_POLYGON_MODE, &polygonMode[0]);  
if (polygonMode[0] == GL_LINE) ...
```

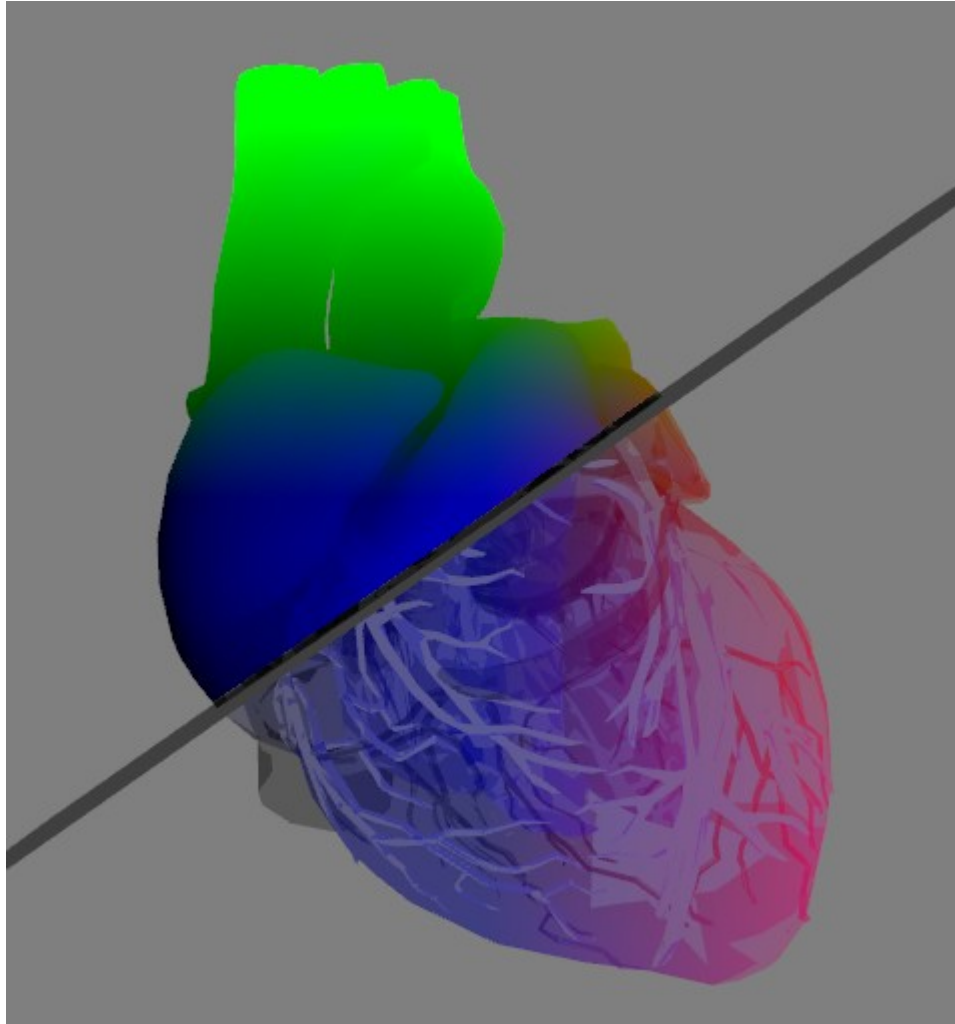


Figure 3 Task 3.4

Task 4: separate the object into two parts using the position of the plane and adjust the displacement of the two halves using the keyboard. The detachment offset will be adjusted from the keyboard. The vertex coordinates must be transformed so that the coordinates are moved away from the plane. This transformation will cause undesired stretching of the intermediate edges. Solve this problem by discarding the fragments that are between the two halves.

- a) Task 4.1a: change the detachment coefficient using U, O keys
- b) Task 4.1b: calculate and transmit the detachment offset to the GPU
- c) Task 4.1c: displace the coordinates above by the detachmentDisplacement and the coordinates below by -detachmentDisplacement
- d) Task 4.1d: discard the fragments that are between the two halves

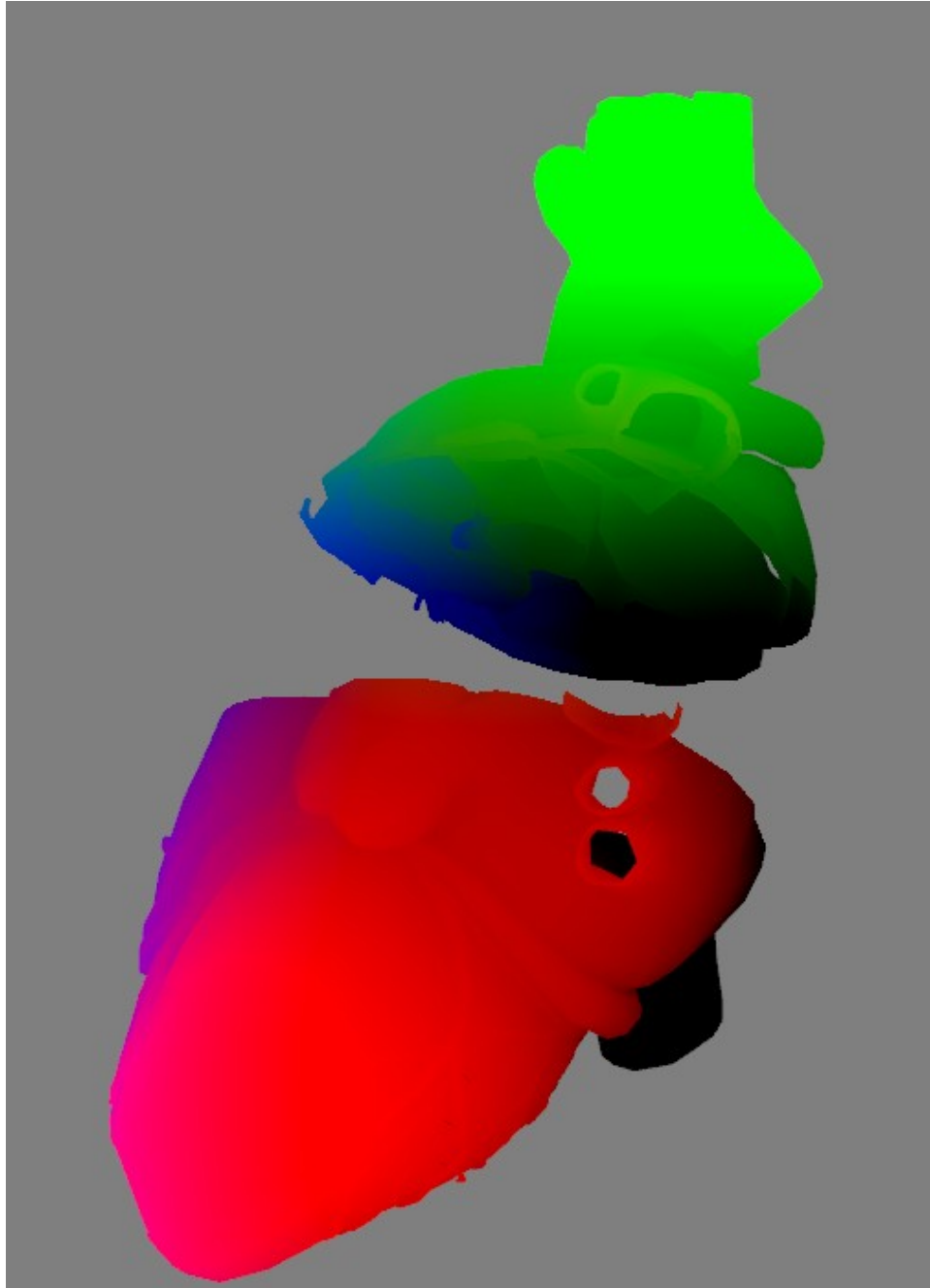


Figure 4 Task 4

Homework:

1. Explain Tasks 1-4.
2. Assign color based on distance from the plane.
3. Tomography - Separate the object in seven parts.
The planes should function just like the plane in Task 4.