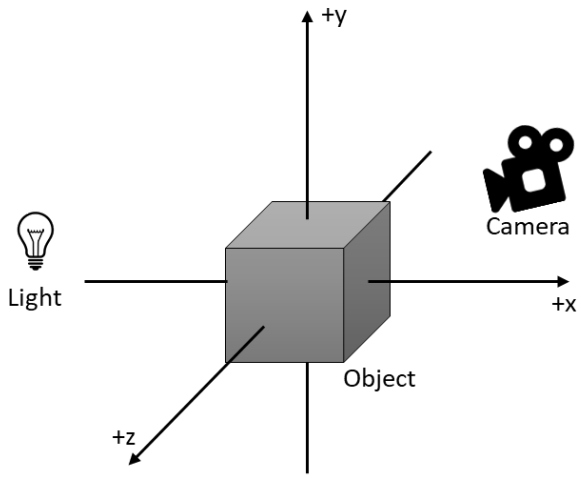
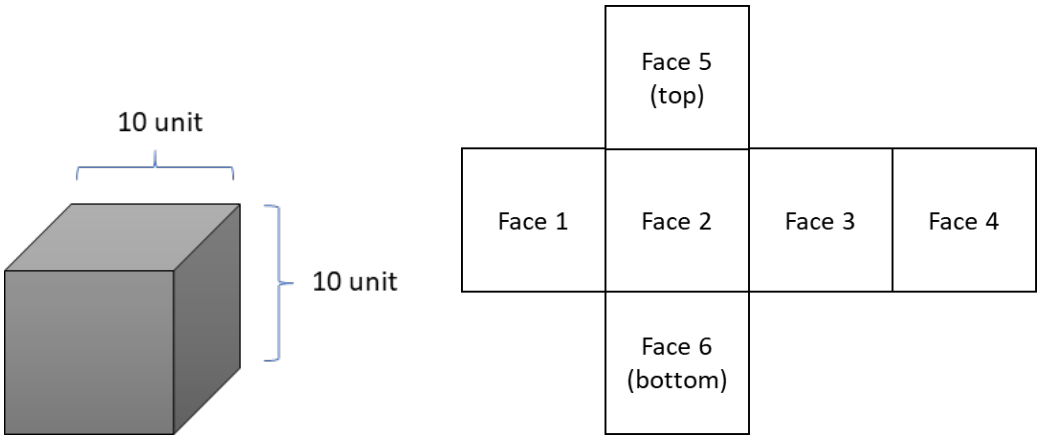
CSCI 4550 Homework 5

In this assignment, you will experiment OpenGL lights and texture mapping. You will start with designing a special cubic object, whose center is positioned at the origin of the coordinate space.



The length of each side the cube will be 10 unit long. This object has 6 faces. Each face (labeled with corresponding ID below) has a different material and texture property. Here is the unfolded image of the object and label of each face.

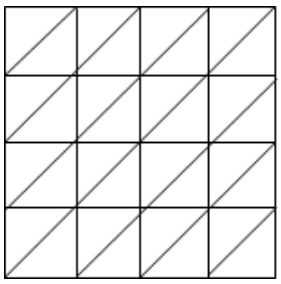


Object Materials: Below are the material and texture properties for each face on the object:

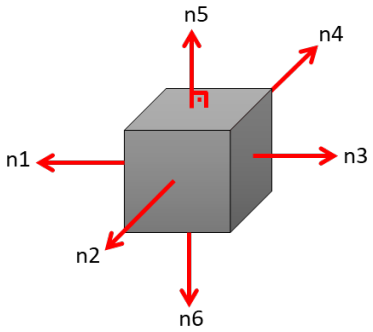
	Ambient	Diffuse	Specular	Texture
Face 1	[0.2, 0.2, 0.2]	[1.0, 1.0, 1.0]	[1.0, 1.0, 1.0]	Image1
Face 2	[0.2, 0.2, 0.2]	[1.0, 1.0, 1.0]	[1.0, 1.0, 1.0]	Image2
Face 3	[0.2, 0.2, 0.2]	[1.0, 1.0, 1.0]	[0.0, 0.0, 0.0]	Image3
Face 4	[0.2, 0.2, 0.2]	[1.0, 1.0, 1.0]	[0.0, 0.0, 0.0]	Image4
Face 5 (top)	[0.2, 0.2, 0.2]	[0.8, 0.0, 0.0]	[1.0, 1.0, 1.0]	N/A
Face 6 (bottom)	[0.2, 0.2, 0.2]	[0.0, 0.0, 0.0]	[0.0, 0.0, 0.0]	N/A

You are free to pick 4 texture images in bitmap (.bmp) format. And you will map them to the corresponding faces given above table. Please make sure your images have 500x500 pixel resolution.

Surface Triangulation: Although square faces can be created by only four vertices, I would like you to create a dense version with more vertices for each square face to achieve proper speculation effects. Each face will be formed from a grid of 5x5=25 vertices and 32 triangular surfaces.



Surface Normals: A normal vector should be assigned to each vertex. Normals of the vertices on faces are directly towards outside of the surface (left image). Right table shows the actual normal vector values you are expected to use for each face.



	Normal Vector	
Face 1	n1	[-1,0,0]
Face 2	n2	[0,0,1]
Face 3	n3	[1,0,0]
Face 4	n4	[0,0,-1]
Face 5	n5	[0,1,0]
Face 6	n6	[0,-1,0]

Light: There will be one light source in the scene. It will be a bright white light located at (-20, 20, 20) in the coordinate space. Light properties are given below:

	Ambient	**Diffuse**	**Specular**
Light 0	[1.0, 1.0, 1.0]	[1.0, 1.0, 1.0]	[1.0, 1.0, 1.0]

Camera: The camera is at fixed location (20, 20, 20). A perspective projection model (such as `gluPerspective(90.0, (float)w / (float)h, 1.0, 100.0)`) can be used.

Animation: When the program starts, the object will rotate with a fixed speed around itself with respect to +y axis (vector [0, 1, 0]). Following keys will be assigned to their given functions.

- Input “space” will pause object rotation.
- Input “a/A” turns on/off the first light.