

## CS 450/550 -- Fall Quarter 2020

### Project #3

100 Points

Due: October 21

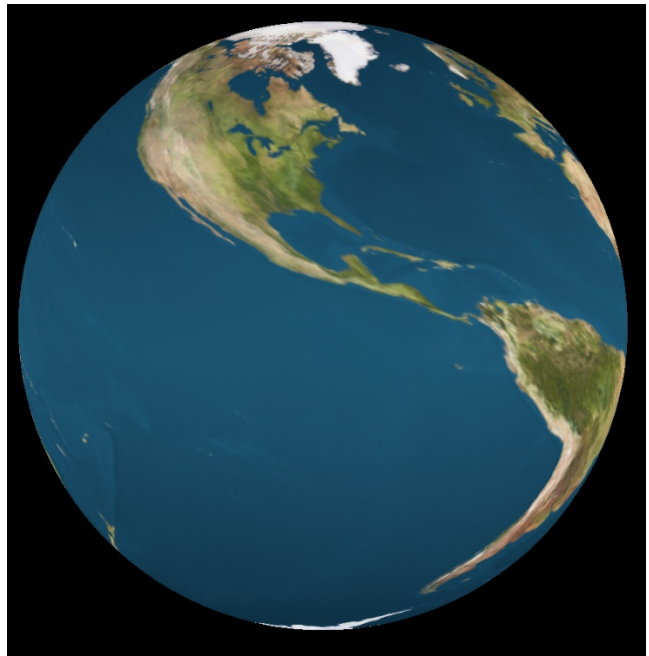
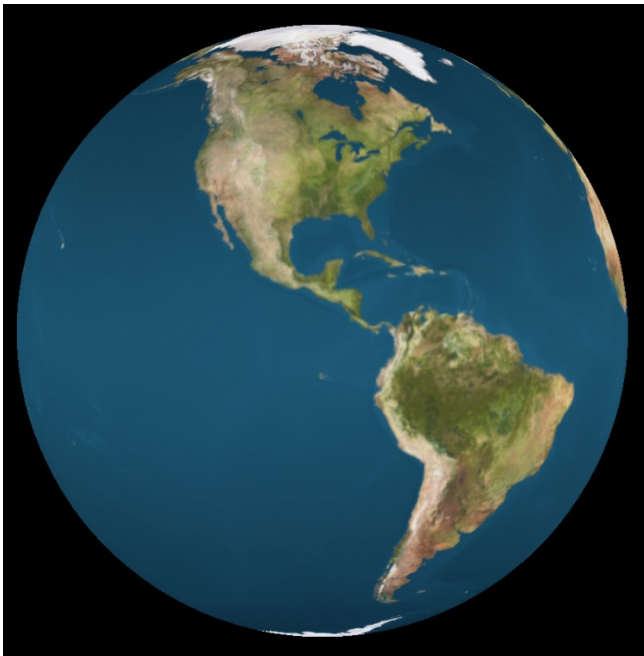
### Texture Mapping

---

*This page was last updated: July 26, 2020*

---

#### Introduction



The goal of this project is to map a texture (your choice) to an object (your choice) and then distort it in some way. What you do here is up to you, but make it more than trivial. But, whatever you do, make it so that you could not do the same thing with a texture transformation. That is, no rigid body rotates, scales, or translates.

## Requirements:

1. Read in a texture image. The choice of image is up to you. You can use the **BmpToTexture( )** function if you want, but you don't have to. Remember that some graphics cards require the dimensions of this image to be powers of two.
2. Create a 3D object that has texture coordinates (s,t) at the vertices, specified by **glTexCoord2f( )** calls before each call to **glVertex3f( )**. Since you will be modifying the creation of the (s,t) texture coordinates, this needs to be an object that you have source-code control over. A cube would work, but is boring. Whatever you did for Project #1 might work. A sphere would work (you are being given the code below).
3. Under control of your GLUT Idle Function, animate the texture distortion. (See below for some advice on doing this.)
4. Under control of a right-mouse-button menu option, allow the object to be displayed:
  1. With no texture -- just an unlit blob-ish color.
  2. With your texture image GL\_REPLACE'ed on it.
  3. With some distortion of the texture-image GL\_REPLACE'ed on it.
5. The distortion should be done by re-drawing the object with a distorted texture coordinate pattern.

## Turn-in:

Use the [Teach system](#) to turn in your:

1. .cpp file
2. A one-page PDF with a title, your name, your email address, a nice screen shot from your program, and the link to the [Kaltura video](#) demonstrating that your project does what the requirements ask for. Narrate your video so that you can tell us what it is doing.

**Be sure that your video is flagged as *unlisted*.**

## Bonus Days:

Each of you has been granted five total Bonus Days, which are no-questions-asked one-day project extensions, but no more than **2** Bonus Days may be applied to any one project. Hint: Bonus Days will likely be worth a lot more to you in the second half of the quarter than they are worth to you in the first half!

## Files You Might Want

[Click here to get the \*\*BmpToTexture\( \)\*\* function.](#)

[Click here to get the \*\*worldtex.bmp\*\* file.](#)

[Click here to get a sphere-drawing function.](#)

## Timing Your Scene Animation

In computer graphics animation, it is best to deliberately time the animation. Here is a good way to do that. Set a constant called something like MS\_PER\_CYCLE that specifies the number of milliseconds per animation cycle. Then, in your Idle Function, query the number of milliseconds since your program started and turn that into a floating point number between 0. and 1. that indicates how far through the animation cycle you are. So, in Animate, you might say:

```
int ms = glutGet( GLUT_ELAPSED_TIME );  
ms %= MS_PER_CYCLE;  
Time = (float)ms / (float)MS_PER_CYCLE;           // [0.,1.)
```

and then in Display, you might use that 0.-1. number something like this:

```
glRotatef( 360.*Time, 0., 1., 0. );
```

### Something to Notice

If you do use the sphere with the Earth texture, so that it looks like a globe, you will find that orthographic looks nicer than perspective. Try it. Why is it this way?

### Grading:

Item	Points
Correctly draw the blob-ish object	20
Correctly draw the normal-texture object	30
Correctly draw the distorted-texture object	50
Potential Total	100