# EECS 487 HW 2

*6 problems totaling 76 points plus points from Pop Quizzes.*
Due: 9:40 Monday 24 November 2014, at the *start* of lecture

Please review the grading policy page of the course web site.

**Sources.** You are allowed to consult both online and offline sources, including humans, to help solve the problems; but if you do consult any outside sources, you **MUST** cite them. You do not need to cite the teaching staff, the textbooks, nor the lecture notes; these are the only exceptions. Your submission must be your, individual work. Classmates can give an idea on how to approach a problem, but cannot give any solution, even in part, to these problems. If you find an online solution to any of these problems, you are allowed to consult it, but not to use it verbatim. You must phrase your solution in such a way that shows you have understood the problem and its solution. Failure to comply with any part of the above policy is a *strict violation* of the Honor Code.

**Notation.** In your answers, you are **required** to represent vectors as lowercase letters with an arrow above ($\vec{v}$), matrices as uppercase letters with hats ($\hat{T}$), points as lowercase letters with tildes ($\tilde{q}$), scalars as lowercase letters ($b$), angles as lowercase greek letters ($\theta$), and colors as lowercase letters with subscripts ($c_d$). Do not forget to use $\oplus$, $\otimes$, and $\ominus$ when necessary (only for colors).

**Write legibly.** If you turn in a handwritten solution, please write legibly; **illegible scribble will earn zero points**. If the grader cannot read what you've written, it will be immediately marked wrong. If you lose points because part of your answer could not be read, you will *not* have the opportunity to explain what it says. It is best to write a *rough draft* where you can scribble and make a mess, and then submit a clean, carefully-written version.

**Partial Credit.** Even if your solution to a problem is far from correct, you may still earn up to half of the total credit on any given part if the effort is substantial and direction is correct. Do not leave any question blank! However, notice that the true/false questions deduct points for incorrect answers, so those are the only ones that you may want to leave without solution if you are unsure of your answer.

1. **True/False with Justification**
**(2) points for a correct answer with justification, (0) points for no answer or a correct answer with incorrect/poor justification, and (-1) points for a wrong answer.**

   (a) ________ **Shadow Volume** In computing shadow volume, the stencil buffer is used solely as a counter.

   (b) ________ **Surface Shading** A normal map always requires more memory to store than a bump map.

   (c) ________ **Anisotropic Filtering** The use of anisotropic texture filtering is not necessary when a scene is rendered orthographically.

   (d) ________ **Texture** The differences between `GL_LINEAR` and `GL_NEAREST` is only noticable for textures larger than 2x2.

   (e) ________ **Buffer Objects** Vertex Buffer Object, Pixel Buffer Object, and Framebuffer Object all serve similar purposes, allowing data to be sent from the CPU or a file directly to the GPU by bypassing the RAM.

2. **Environment Mapping and Textures.** (10 pts).

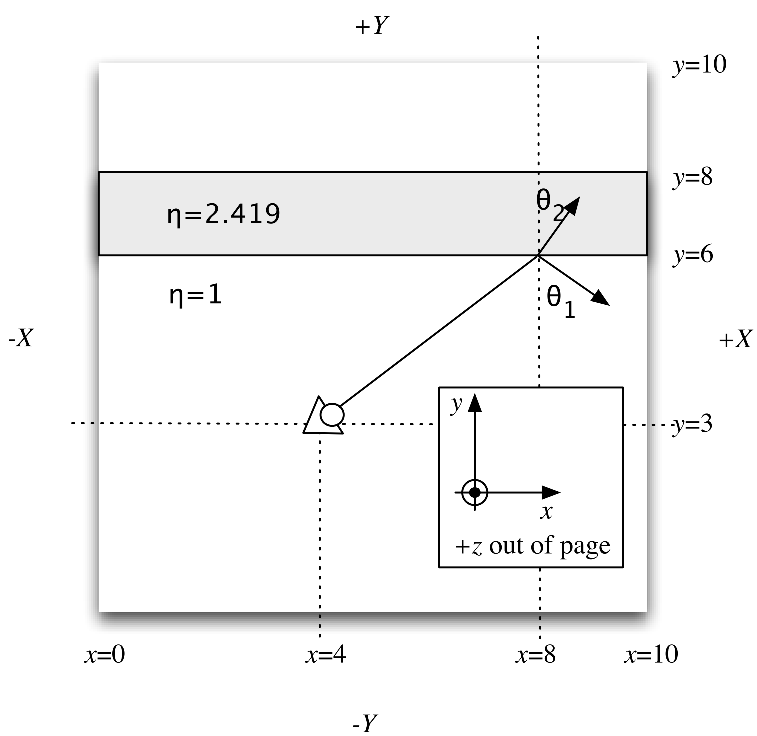



Figure 1: A slice of the cubical room at $z = 5$

You are in a cubical room of side 10 ft. All the descriptions below happen at $z = 5$ ft. Fig. 1 shows the slice of the room through this $z$-plane. On this plane, the eye is at $(4, 3, 5)$. A diamond wall spans the room from floor to ceiling, between planes $y = 6$ ft. and $y = 8$ ft.

Assume a single texel is mapped to each of the four walls such that if you take the frontal view of a wall and its associated texel, the lower left corner of the texel is mapped to the lower left corner of the wall, and the upper right corner of the texel is mapped to the upper right corner of the wall. Each wall may have a different texel mapped to it. So:

**for wall $-X$, the coordinate mapping scheme is:**
$(s = 0, t = 0) \mapsto (0, 0, 0)\,; (0, 1) \mapsto (0, 0, 10)\,; (1, 0) \mapsto (0, 10, 0)$; and $(1, 1) \mapsto (0, 10, 10)$,

**for wall $+Y$, the coordinate mapping scheme is:**
$(s = 0, t = 0) \mapsto (0, 10, 0)\,; (1, 1) \mapsto (10, 10, 10)$,

**for wall $+X$, the coordinate mapping scheme is:**
$(s = 0, t = 0) \mapsto (10, 10, 0)\,; (1, 1) \mapsto (10, 0, 10)$,

**for wall $-Y$, the coordinate mapping scheme is:**
$(s = 0, t = 0) \mapsto (10, 0, 0)\,; (1, 1) \mapsto (0, 0, 10)$.

We wish to determine the color at point $(8, 6, 5)$ on the "front" surface of the glass wall by cube-mapping. The viewing ray incident on this point is 50% reflected and 50% transmitted such that the color at this point is 50% contributed by the reflected texel and the other 50% by the texel seen through, and refracted by, the glass wall.

(a) **Reflection** (5 pts). Which cube wall, and what is the texel coordinate for that wall, contributes to the 50% of the color on the glass wall at location $(8, 6, 5)$ due to the reflection? Show your work.

(b) **Transmission** (5 pts). The viewing ray continues through the glass wall. Using Snell's law of refraction,

$$\eta_1 \sin\theta_1 = \eta_2 \sin\theta_2, \tag{1}$$

assuming refractive index for air=1, diamond=2.419, determine the cube wall and the texel coordinate seen through the diamond wall that contributes to the other 50% of the color at location $(8, 6, 5)$.

3. **Short Answers** (16 pts).

(a) **Bump Mapping** (2 pts). A scene contains a strawberry rendered using bump mapping. How would you be able to tell that this strawberry has been bump mapped? Name one method that can be used to make the strawberry look more realistic.

(b) **Shadow Mapping** (4 pts). Consider a scene in a game with three different light sources: A ceiling light high above the player, a miner's headlamp attached to the player's head, and a flashlight an enemy is shining towards the player from across the room. Which light would most benefit from using multiple shadow map slices with different resolution? Is it necessary to use multiple slices for all the lights? Justify both of your answers.

(c) **GPU Effects** (6 pts). The main concepts/tools for use in GPU effects (*not* raytracing) are the various buffers constituting the framebuffer, environment mapping, random sampling, and image space computation with perhaps one bounce. For each of the following effects, explain how two (2) of the listed tools could be used to achieve it:

i. **Shadows** (2 pts).
ii. **Reflections** (2 pts).
iii. **Depth of Field** (2 pts).

(d) **Light an Image** (4 pts). A common feature of image editing packages is to add lighting to an image. This is done by using one of the red, green, or blue channels to compute the desired illusion of 3D roughness. An example is shown below (the green channel was used, but it does not matter since the picture has little color).

Explain how using OpenGL you might produce the image on the right, if all you have is the one on the left.

4. **Splines** (16 pts). Answer the following questions about splines.

   (a) **Degree** (2 pts). Why are the most popular types of splines cubic? Why not quadratic or higher order?

   (b) **Interpolation** (2 pts). Identify the main problem with interpolating a spline through the control points instead of approximating a spline near the points.

   (c) **Ideal Spline** (4 pts). I have a curve representation with the following properties:
   1. The curve is a cubic
   2. The curve interpolates its control points
   3. The curve has local control
   4. The curve has $C^2$ continuity

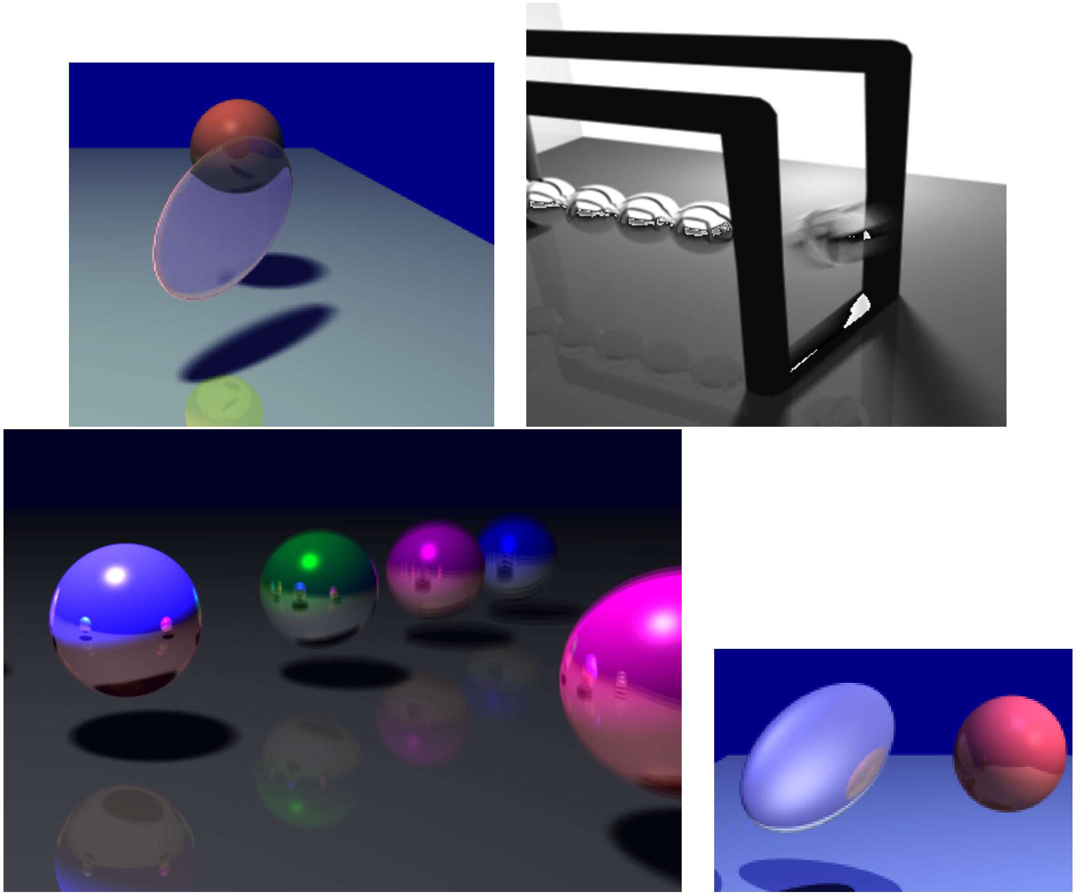
   I want to use this curve for rendering an animation of a face. Would the animation look realistic? Why or why not?

   (d) **Matrices** (4 pts). Derive the constraint and basis matrices for a natural cubic spline whose control points are:

$$p_0 = f(0)$$
$$p_1 = f'(1)$$
$$p_2 = f''(1)$$
$$p_3 = f(1)$$

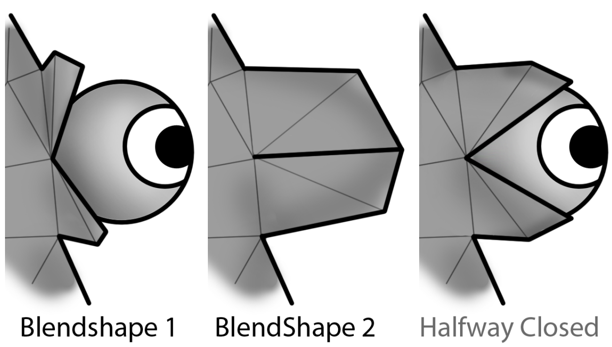
   (e) **Smooth Interpolation** (4 pts). Given $n \geq 4$ control points, we use cubic Catmull-Rom to interpolate all but the first and last control points. To interpolate the segment between the first two control points, we will use quadratic splines. What should the basis matrix for the segment between the first two control points be to ensure $C^1$ continuity at the second control point where the quadratic spline meets the cubic Catmull-Rom spline? Show how you derive the basis matrix.

5. **Distributed ray tracing** (16 pts). For each image in the figure below, identify one special effect **in addition to** anti-aliasing and soft-shadows present (i.e., you cannot give anti-aliasing nor soft-shadows as the answer) that requires the use of distributed ray tracing, i.e., ray tracing alone cannot generate the special effect.



For each effect, draw a ray (or rays) that shows how the effect is achieved using ray tracing. Be sure to give relevant formulas, if needed. For example, if you were to show anti-aliasing with jittered sampling, you would draw a pixel divided into subpixels, with a ray going through a random point in each subpixel.

6. **Blendshapes** (8 pts). An animator is creating an animation with a chameleon and using blendshapes (morph targets) for the face. Consider the blendshapes below.




The first is for the eyelids open, and the second, for them closed. To generate the third image, a programmer proposes a linear combination of the vertices in blendshapes 1 ($\vec{v}_1$) and 2 ($\vec{v}_2$). He can then specify a weight $w$, with the position of vertex $\vec{v}$ in the resulting mesh given by $\vec{v} = w\vec{v}_1 + (1-w)\vec{v}_2$. Then, for the third image, he suggests $w = 0.5$. (a) Does this setup and algorithm produce the desired third image? Justify your answer by briefly describing the steps taken by the proposed algorithm (2 pts). (b) Describe *two* other algorithms to animate the eyelids to create the desired third image.

Describe the algorithms without writing a program or pseudocode. Your algorithms may, or may not, involve blendshapes. Briefly describe how your proposed algorithms work to create the third image. (6 pts).