Practice Final
CMSC 427
Distributed Tuesday, December 11, 2007
Review Session, Monday, December 17, 5:00pm, 4424 AV Williams
Final: 10:30 AM Wednesday, December 19, 2007

General Guidelines: The final will focus on topics that have been discussed in class. I will not ask questions about material from the text book that has not also been discussed in class or in a problem set. However, there will be no questions about OpenGL or programming problems. You will be expected to know some basic equations, but nothing very obscure. I'll try to indicate below the key equations you should know.

The final will be closed book, with no notes or calculators. This means that problems should not require the use of a calculator; if you feel you need one when you are doing a problem this probably means there is an easier way to do the problem.

The final will be comprehensive. It will focus more (about 3/4 emphasis) on the material covered since the midterm, but you are still responsible for knowing material from the first half of the semester.

Topics and things to know about them:

1. Geometry (needed for problems on the next two topics)
    a. How to take an inner product, and what this does.
    b. How to take a cross product, and what this does.
    c. What a unit vector is, and how to find the magnitude of a vector.
    d. How to write the equations for a 3D line, or for a plane in 3D, or for a sphere.
2. Transformations
    a. How to write rotation about the x, y, or z axis as a matrix.
    b. How to write translation as a matrix.
    c. How to combine these. For example, rotate about a point not at the origin.
    d. How to rotate or translate a line.
    e. How to relate the rows of a rotation matrix to the axes of a new coordinate system
        This includes writing the coordinates of points relative to a new coordinate system.
        Given a position, lookat vector, an upvector, you should be able to write a matrix that transforms the world so that the viewer is at the origin, with the lookat direction in the z direction, and the upvector in the y direction.
3. Projection
    a. Perspective projection
        i. How to find the image coordinates of a 3D point
        ii. Some consequences of that, such as properties of vanishing points, knowing when a line in 3D does not project to a line in the image, …..
    b. Orthographic projection
        i. How to find the image coordinates of a 3D point.

        ii. Consequences of orthographic projection, such as that it preserves parallelism and ratios of areas.

4. Intersections and Relations between lines and planes and points.
    a. How to use the equation of a line or plane to tell which side of the line/plane a point is on.

        Example: Given a plane and two points, are the points separated by the plane, or on the same side of it?
    b. How to find the intersections of spheres, axial rectanguloids, triangles.
    c. How to intersect a line with a plane, triangle, or sphere
5. Discretization
    a. How to represent a line with pixels.
    b. How to fill a polygon.
        i. How to tell if a point is inside a polygon.
            1. Implicit function test (testing that it is on the appropriate side of each line).
            2. Crossing number test.
        ii. Flood fill algorithm.
6. Sampling and Aliasing
    a. Creating an image by sampling
    b. Aliasing – what it means, what causes it.
        i. Spatial
        ii. Temporal
    c. Anti-aliasing
        i. Smoothing
        ii. Supersampling and smoothing
7. Interpolation
    a. Linear, bilinear
    b. Cubic
    c. Bezier curves -- Given three control points, you should know how to construct points on a Bezier curve, and know some of its basic properties.
8. Texture
    a. Perlin noise algorithm – General idea
9. Morphing and interpolation
    a. Morphing algorithm that you implemented.
    b. Describing points in a new coordinate system.
10. Color
    a. Superposition
    b. Color Spaces – CIE, RGB, CMYK, HSV.
11. Visibility
    a. Culling – Basic idea
    b. Painter's algorithm
        i. For example, how to tell if a triangle 1 is in front of, or behind, the plane that contains triangle 2, and how knowing this might help you in the painter's algorithm.
    c. Z buffer.

        i. Given the vertices of a 3D triangle, you should be able to figure out where they appear in an image (perspective projection), which 2D points are inside the triangle, and what their depth is.
   d. BSP Trees – The basic idea of the algorithm. See the problem below.
12. Lighting – Optics and Reflectance
    a. Ambient – What does this mean?
    b. Lambertian -- Know the equation for Lambertian reflectance.
    c. Specular –
        i. Know the equation for Phong reflectance.
        ii. Know how to figure out the direction of mirror reflection.
13. Shading – Know what the following mean.
    a. Gouraud
    b. Phong
14. Ray Tracing
    a. Shadows
    b. Finding rays from focal point to pixels.
    c. Given the intersection point between a ray and an object, know in which directions you need to cast additional rays.
15. Shadows
    a. Z-buffer algorithm.
    b. Casting a shadow with a directional or point source onto a plane.
16. Radiosity and photon mapping – Basic idea
17. Modeling
    a. Implicit shapes – Don't memorize which shapes have which equations (except you should know the equation for a sphere).
    b. Constructive solid geometry -- You need to anyway know how to tell whether a point is in a simple geometric object (eg., cube, sphere), and where a ray intersects it. You should understand how to use this to tell whether a point is inside, or a ray intersects, a more complex object made up of simple ones.
    c. Fractals -- Just the basic idea; no math (eg., don't worry about complex numbers).

Sample Problems (Note, some of these problems may be a bit more involved than ones I'd ask on a time-limited exam).

1. Suppose the camera's focal point is at (1,2,3), you are looking in the direction (1,1,1), and the focal length is 1. Give a ray from the focal point through the center of the image.

The ray begins at (1,2,3), and goes in the direction (1,1,1).

2. Suppose the camera has a focal point at (0,0,0), and you cast a ray through a pixel whose center is at (1/2, 0, 1). Does this ray intersect a sphere centered at (3, 2, 12) with a radius of 3?

We can describe the line the ray is on with the equations $x = z/2$, $y = 0$. The sphere has the equation:

$$(x-3)^2 + (y-2)^2 + (z-12)^2 = 9$$ (^2 means squared)

Combining these equations we get:

$x^2 - 6x + 9 + 4 + 4x^2 - 48x + 144 = 9$
$5x^2 - 54x + 148 = 0$
The solution to this quadratic equation is:
$x = (54 \pm \sqrt{-44})/10$. That means that both solutions are imaginary, so there is no true intersection point.

3. Suppose the camera has a focal point at (0,0,0), and you cast a ray through a pixel whose center is at (1/2, 0, 1). This ray definitely intersects a sphere centered at (3, 2, 12) with a radius of 5. What are the coordinates and surface normal of the first point that the ray intersects?

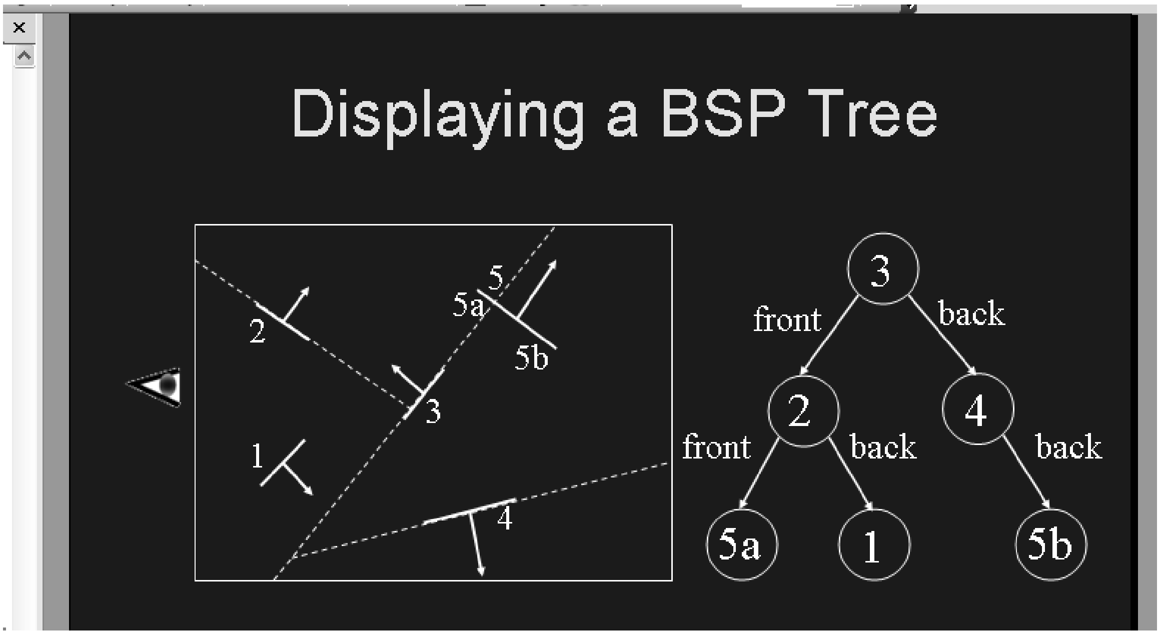
We get the same equation for a line, and for a sphere we get:

$$(x-3)^2 + (y-2)^2 + (z-12)^2 = 25$$ (^2 means squared)

Combining these equations we get:

$x^2 - 6x + 9 + 4 + 4x^2 - 48x + 144 = 25$
$5x^2 - 54x + 132 = 0$
The solution to this quadratic equation is:
$x = (54 \pm \sqrt{276})/10 = 7.1, 3.7$. These produce the points (7.1, 0,14.2), (3.7, 0, 7.4). The ray starts at the origin, and x and z increase as we go along, so it first reaches the point with the smallest values of x and z, which is (3.7,0,7.4). The normal of this point is a unit vector in the direction from the center of the sphere to this point, which is (3.7,0,7.4) – (3,2,12) = (.7, 2, -4.6)/5. (There's a bit of round-off error here).

4. Consider the BSP tree discussed in class:



a. Suppose we look at this scene from the right side, instead of the left. Use the BSP tree to determine the order in which the triangles should be rendered.

From the right, we're behind 3, so we want to first render things in front of 3. We're in front of 2, so we first render things behind it, which is 1. Then we render 2, then anything in front of it, which is 5a. Then 3. Then we are behind 4, so we first render things in front of it. But there is nothing there, we render 4, then finally 5b.

b. Construct a valid BSP tree starting with triangle 1 instead of triangle 3.

Starting with 1 as the root, 3, 4 and 5 are in front, and 2 is behind. We could pick any of 3, 4 or 5 to be the child of 1. Let's pick 4. Then 3 and 5 are in back of it. Pick its child to be 5, and 3 is the final child, behind 5.

5. Suppose we construct a model by taking a sphere centered at (10,0,10) with a radius of 2, and subtracting a cylinder. The cylinder is centered also at (10,0,10). It has a circular cross-section with a radius 1 when we slice it with a plane that has a constant z value (eg., if we intersect the cylinder with the z = 10 plane, we'll have a circle centered at (10,0,10) with a radius of 1). The cylinder has a length of 5. Suppose we cast a ray from (0,0,0) in the direction (5,0,4). Will it intersect the model? If so, where?

We can describe an infinite cylinder that includes ours with the equation
(x-10)^2 + y^2 = 1. We will see whether the ray intersects this infinite cylinder in its actual, finite extent. The line the ray is on has the equation:
(x,y,z) = (0,0,0) + t(5,0,4) = (5t,0,4t). Substituting, we get:
(5t-10)^2 = 1. So 5t - 10 = +-1 and t = 9/5 or 11/5. So the ray intersects the cylinder at:

(9, 0, 36/5) and (11, 0, 44/5). The cylinder's z values range from 7.5 to 12.5. This means the ray enters the infinite cylinder before it strikes the finite cylinder, and exits the finite cylinder at (11,0,44/5). So the ray enters the cylinder when z = 7.5, which is at the point (75/8, 0, 15/2). So now, the question is, does the ray intersect the sphere outside of the range (75/8, 0, 15/2) to (11,0,44/5)?

The ray, with equations y = 0, 4x = 5z intersects the sphere, with equation (x-10)^2 + (z-10)^2 = 4, at the point (10,0,8) and (11.95, 0 9.56). So the ray stays in the sphere after it has left the cylinder. That is, at the point (11,0,44/5) the ray is leaving the cylinder but still in the sphere, so it intersects the object there.

6. Suppose we construct a Bezier curve, using the control points (0,0), (1,1), (2,0).
   a. Give four points that lie on the curve.

The curve starts at (0,0), and ends at (2,0). The midpoint can be found first by averaging (0,0) and (1,1) to get (1/2,1/2). Then we average (1,1) and (2,0) to get (3/2,1/2). Then we average these to get (1,1/2). If we want a point ¼ of the way along the contour we can take weighted averages obtaining first the points (1/4,1/4) and (5/4,3/4), and then averaging these we get (1/2, 3/8).

   b. What is the tangent of the curve at its beginning and ending points?

The tangent of the first (last) point is towards (away from) the second point. That is, in the direction (1,1) and (1,-1).

7. Suppose we construct a curve in the following way. We create a half circle starting at (0,0), passing through (1,1) and ending at (2.0), with a radius of 1. Then we connect this to a half-circle starting at (2,0), passing through (3,-3), and ending at (4,0), with a radius of 1. How smooth is this curve?

At the place where the half-circles join, the tangents are the same, but the curvatures are not. This is one way to define a level of smoothness, by how many derivatives exist.

8. Suppose you want to model the reflectance properties of a blue car.
   a. What reflectance model might you choose, and what parameters might you use? Why? You might want to limit yourself to using the models we've discussed in class (eg., Phong, Lambertian).

You could use a combination of Phong and Lambertian. The Phong exponent should be very large, so there is a mirror-like reflection. But the hood is also part Lambertian, because we don't just see the world reflected in it, we also see the color of the hood.

   b. Suppose you have a flat piece of blue car hood. You shine a light on the car from an angle of pi/4. Then you take a few pictures of the car from different angles. How might you use these images to determine what parameters you should use to model its reflectance? How many images do you think you would need?

If you take a picture in any direction other than that of the mirror reflection, you'll just get the diffuse component of reflection. In the mirror direction you get lambertian plus mirror reflection. If you subtract the first image from the second, you get the pure mirror part of the reflection. This gives both Lambertian and mirror separately. Note that a real car hood reflects light in a way that is more complicated than Lambertian plus mirror; this is just a simple model.

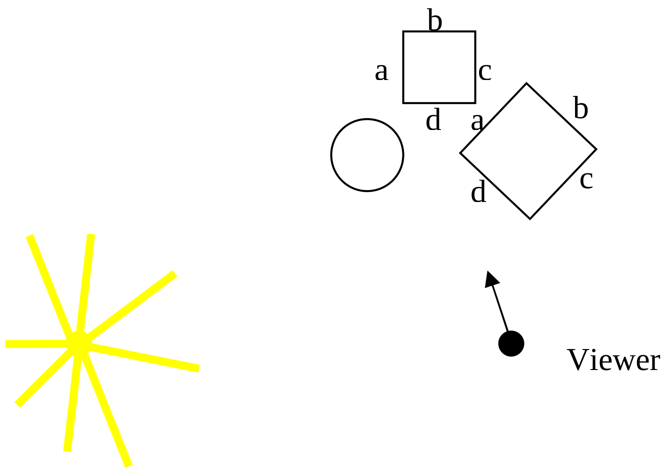
9. Create a simple scene with a viewer looking in a specific direction, one or two simple objects, and a point source of light. Then work out how to render or ray trace this scene. Figuring out what kind of scene will create a reasonable set of problems will also be helpful to your studying. This is an Uber-problem. If you can do this, you have studied much of what is needed.

   a. Transformations
      i. Create a matrix that will transform the world so that the viewer is at the origin and the viewing direction is in the z direction.
      ii. Apply this matrix to an object in the scene to determine its coordinates after this transformation.
      iii. Apply the equations of perspective projection to determine the image location of the object.
   b. Ray Tracing.
      i. Find the equation for the image plane.
      ii. Determine a ray that goes from the viewer to a vertex in the scene, and figure out where it intersects the image plane.
      iii. Determine a ray that goes through the center of the image, and figure out which object in the scene it first intersects.
      iv. Suppose a ray strikes a surface that is a bit shiny (that is, it reflects light with some Phong and some Lambertian reflectance), but it is not at all transparent. If we are allowing for multiple bounces and shadows, what rays will we cast from the point where the original rays hits the surface. Give the direction of each ray.
   c. Lighting
      i. Find a ray from the viewer that intersects an object in the scene. Suppose the object is Lambertian and white. Determine the intensity it will produce in the image.
      ii. Now do the same, assuming the object has Phong reflectance.
   d. Shading
      i. The surface normal of a vertex isn't really well-defined. One way to define it is to take the average of the normals of the sides of the object that form the corner. Using this approach, find the normals for the vertices of a polygon in the scene.

ii. As in (c), find the intensity that each of these vertices will produce in the image. Then determine the intensity that you will get in the middle of the polygon using Gouraud or Phong shading.

Consider the following scene.

- Imagine the ground is at y = 0.
- **Cube1:** There is a cube with corners at (20,0,20), (18,0, 18), (16, 0, 20), (18,0,22), (20,2,20), (18,2,18), (16,2,20), (18,2,22).
- **Cube2:** There is another cube with corners at (15, 0, 21), (17,0,21), (17,0,23), (15,0,23), (15, 2, 21), (17,2,21), (17,2,23), (15,2,23).
- **Sphere:** There is a sphere with a radius of 1, centered at (14, 1, 20).
- **Viewer:** There is a viewer located at (18,0,10), who is looking directly at the left corner of the first cube, that is, at the point (16,0,20). The camera has a focal length of 1.
- **Light:** There is a point light source located at (4,0,10).



1. Transformations

a. Create a matrix that will transform the world so that the viewer is at the origin.

We just need to translate the viewer. We can do this with:

$$\begin{pmatrix} 1 & 0 & 0 & -18 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -10 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

b. Create a matrix that will transform the world so that the viewer is at the origin and the viewing direction is in the z direction.

The viewing direction is a vector from the viewer to the point you're looking at. That is, (16,0,20) – (18,0,10) = (-2, 0 10). We want to make this a unit vector, so we divide it by sqrt(104). Then we want to transform the world so that this is the new z direction, by putting this in the third row of a transformation matrix. So we get:

$$\begin{pmatrix} \frac{10}{\sqrt{104}} & 0 & \frac{2}{\sqrt{104}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{-2}{\sqrt{104}} & 0 & \frac{10}{\sqrt{104}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & -18 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -10 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Notice that I had to choose the first and second rows to be orthonormal with the third. There are many choices here. I used guesswork. When one vector looks like (a, 0, b), it is easy to see that (0,1,0) will be orthogonal to it and a unit vector, and then we know that (b, 0, -a) must have the same magnitude, but be orthogonal.

To check this, we can apply the resulting matrix to the point (16,0,20).

$$\begin{pmatrix} \frac{10}{\sqrt{104}} & 0 & \frac{2}{\sqrt{104}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{-2}{\sqrt{104}} & 0 & \frac{10}{\sqrt{104}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & -18 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -10 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 16 \\ 0 \\ 20 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{10}{\sqrt{104}} & 0 & \frac{2}{\sqrt{104}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{-2}{\sqrt{104}} & 0 & \frac{10}{\sqrt{104}} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} -2 \\ 0 \\ 10 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \sqrt{104} \\ 1 \end{pmatrix}$$

This is right. The point we are looking at has x=0, y= 0, and z gives the distance from the viewpoint to this point.

c. Apply this matrix to Cube1 to determine its coordinates after this transformation.

Just do the same as above.

d. Apply the equations of perspective projection to determine the image location of the eight corners of the cube after this transformation has been applied. Make a drawing to show these.

We have transformed the world so that the viewer is at the origin and the image plane is the z = 1 plane. So we just need to take coordinates (x/z, y/z). For the point we're looking at, the coordinates will be (0,0).

2. **Ray Tracing.** In this problem, imagine that we are rendering the scene using ray tracing.

a. Keep in mind that the focal length of the camera is 1, and that we know what direction we are looking at. Find the equation for the image plane.

As stated before, the viewing direction is (-2,0,10)/sqrt(104). Call this vector n, because it is normal to the image plane. Then, an equation for the image plane is <n,(x,y,z)> = <n, (18,0,10)> + 1 = 64/sqrt(104) + 1. We can write this as:
-2x/sqrt(104) + 10z/sqrt(104) = 64/sqrt(104) + 1. That is, the image plane is the set of points that are all the same distance from the origin in the direction n, and this distance is the distance to the viewer, plus one (for the focal length).

b. Suppose we cast a ray towards the lower corner of Cube1, (18,0, 18). Where would this ray intersect the image plane?

This ray is going from the focal point to (18,0,18), so it’s in the direction: (18,0,18)-(18,0,10) = (0,0,8). The equation for this ray is:

(18,0,10) + t(0,0,1) = (x,y,z). If we substitute into the equation for the image plane, we get:

-36/sqrt(104) + 10(10+t)/sqrt(104) = 64/sqrt(104) + 1.
-36 + 100 + 10t = 64 + sqrt(104)
10t = sqrt(104)
t = sqrt(104)/10
(x,y,z) = (18, 0, 10 + sqrt(104)/10)

c. Suppose we cast a ray from the focal point in the direction (-.2, 0, .98). Which surface will this ray intersect? What will be the location of the intersection point? Where will this ray intersect the image plane?

First, let’s get a sense of what’s going on. This ray is almost entirely in the z direction. Let’s look at some points on it. It starts at (18,0,10). The lowest point of the cubes is z = 18, so let’s see what happens when we go +8 in the direction of the ray. We wind up at the point (16.4, 0, 17.84). So by the time the ray reaches the lowest point in Cube 1, it is to the left of side C. Going a bit further, the ray will get to (16, 0, 19.8). So it is also going to miss side D of cube 1. Looks like it’ll hit side D of cube 2. To check this, we must see what the x value is when z = 21. Since we have (x,y,z) = (18,0,10) + t(-.2, 0, .98), when z = 21, 11 = t*.98, so t = 11.2245. So when z = 21 on the ray, x = 18 - .2*11.2245 = 15.75. Side D of cube 2 goes from (15,0,21) to (17,0,21), so there is an intersection.

d. Suppose cube1 is a bit shiny (that is, it reflects light with some Phong and some Lambertian reflectance), but it is not at all transparent. We cast a ray from the camera’s focal point which hits sideD of cube1 at the point (17,1,19). If we are allowing for multiple bounces and shadows, what rays will we cast from (17,1,19)? Give the direction of each ray.

First, the shadow ray will go from the intersection point towards the light. That is, its direction will be (4,0,10) – (17,1,19) = (-13, -1, -9). We make this a unit vector by dividing by sqrt(251).

We also need to cast a ray in the direction of mirror reflection. First, we need to know the ray in the direction to the viewer. This is (18,0,10) – (17,1,19) = (1,1,-9). The unit vector is (1,1,-9)/sqrt(83). Because the cube is at a 45 degree angle, its normal vector is (-1,0,-1)/sqrt(2). Now we can use the equation (v + r) = 2*n*<n,v>. This gives us:
r + (1,1,-9)/sqrt(83) = 2*8/sqrt(166)*(-1,0,-1)/sqrt(2).

3. **Lighting**

a. Suppose the two cubes are Lambertian and white (they have an albedo of 1). What will be the intensity of the mid-point of each side of each cube? I mean the center points, which have a y value of 1. For example, the midpoint of Cube1-sideD is (17,1,19). Take into account shadows and the fact that light falls of as $1/d^2$. To get started, we suppose that the center point of sideD in cube1 has an intensity of 100.

This is a bit complicated. If the point is in shadow, it's intensity is zero. First of all, one can see that most of the cube sides are facing away from the light, and so they will be attached shadows with intensity of 0. Only sides a and d are facing towards the light. We know from the problem that side D of cube 1 isn't in shadow. So we have to check on the others. Specifically, is the sphere shadowing any of them? You'll have to calculate the midpoints, and see whether a line from the midpoint to the light intersects the sphere.

For the ones that aren't in shadows, side D cube 1 allows us to calculate the intensity of the light. Then we need to compute the normal of each side, the inner product between this and the direction to the light (don't forget to make it a unit vector) and scale this by the light intensity and 1/d^2.

4. **Shading**

a. The surface normal of a cube at its corner isn't really well-defined. One way to define it is to take the average of the normals of the three sides of the cube that form the corner. Using this approach, find the normals for the four corners of sideD of cube1.
b. Suppose cube1 is a white, Lambertian object, and the light has intensity of 1. Ignoring any cast shadows from the sphere, determine the intensity of light reflected at each of these four corners.
c. Suppose we use Gouraud shading. What will be the intensity of the point (16.5, .5, 19.5), which is on sideD of cube1, in the image?
d. What would be the answer to problem (c) if we use Phong shading?