CPSC 314
Assignment 4

due: Friday Dec, 2nd 2016, in class

Answer the questions in the spaces provided on the question sheets. If you run out of room for an answer, continue on the back of the page.

Name: ______________________________

Student Number: ______________________________

Question 1	/ 5
Question 2	/ 5
Question 3	/ 5
TOTAL	/ 15

1. (5 points) Ray Intersections with an Ellipse: Given a ray originating at $C = (1, 1)$ with a direction $v = (-1, -1)$, determine whether it will intersect an ellipse given by the implicit equation $x^2 + \frac{y^2}{4} = 1$. If it does intersect the ellipse, find the coordinates of the intersection(s) and the normal(s) to the ellipse at these points. If there are two intersections identify the one closest to the origin of the ray. Reminder: use parametric ray equation: $\mathbf{P}(t) = \mathbf{C} + t\mathbf{v}, t \geq 0$ *Solution.* Let's start by finding intersection points. Writing out the parametric ray equation for our point and direction, and the ellipsoid equation, we get a system of equations and an inequality:

$$\begin{cases} x = 1 - t \\ y = 1 - t \\ t > 0 \\ x^2 + \frac{y^2}{4} = 1 \end{cases}$$

By substituting first two equations into the third one, we get:

$$(1 - t)^2 + (1 - t)^2/4 = 1,$$

which can be simplified as

$$5t^2 - 10t + 1 = 0.$$

The two roots of this equation are

$$t_1 = 1 + \frac{2}{\sqrt{5}}, t_2 = 1 - \frac{2}{\sqrt{5}}$$

Those two roots correspond to the intersections of a **line** passing through the ellipse; in our case it's a ray, and the root with the smaller t value is the intersection closest to the origin of the ray, in this case t_2.

To find the coordinates, plug in the values of t back into the parametric equations for x and y. The coordinates for t_1 and t_2 respectively are then:

$$(-\frac{2}{\sqrt{5}}, -\frac{2}{\sqrt{5}}), (\frac{2}{\sqrt{5}}, \frac{2}{\sqrt{5}})$$

To find the normals at that points, let's compute gradient of the ellipse equation:

$$\begin{pmatrix} n_x \\ n_y \end{pmatrix} = \begin{pmatrix} \frac{\partial F}{\partial x} \\ \frac{\partial F}{\partial y} \end{pmatrix} = \begin{pmatrix} 2x \\ y/2 \end{pmatrix}$$

Plugging the coordinates we obtained form the previous step we get the normals at t_1 and t_2 as:

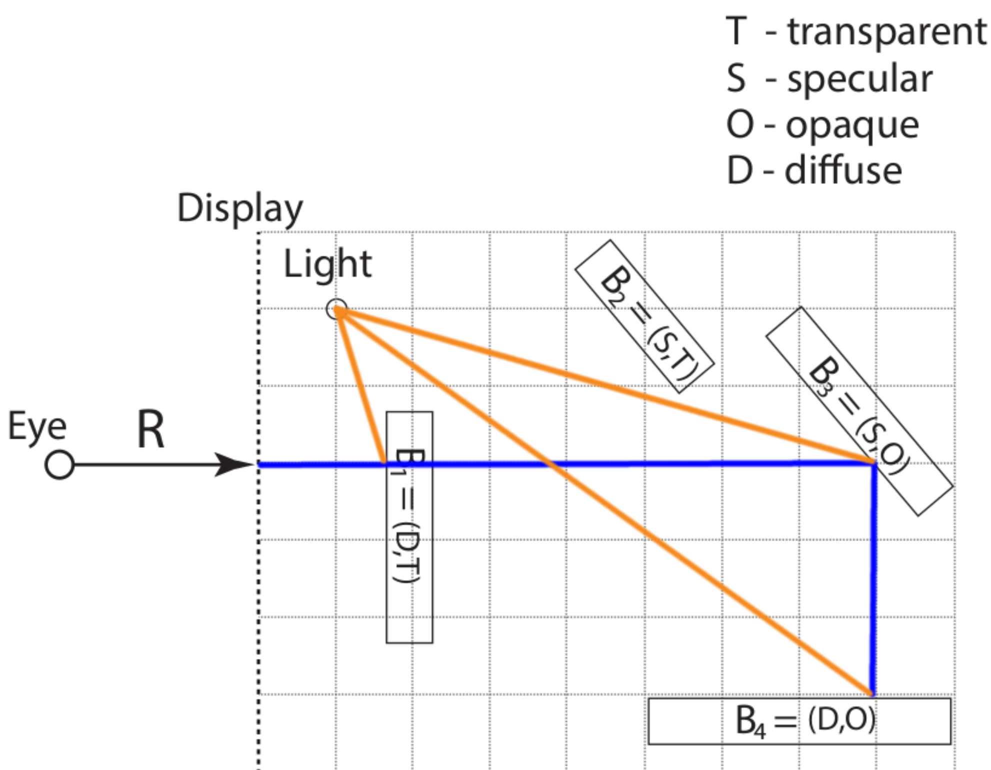
$$\begin{pmatrix} \frac{4}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{pmatrix}$$

Optionally you can normalize the normal, but that was not required.

2. Ray-Tracing

   (a) (3 points) Draw the ray tree for the ray R shown below. Assume index of refraction c_1 for air is 1 and index of refraction for all the transparent objects in the scene is $c_2 = 1.2$. Use Snell's law to obtain (approximate) refraction angles.

   Solution First we draw the path of the eye ray. The ray passes through B1 a transparent object at an orthogonal angle so that Snell's law gives 0 degrees as the angle of refraction. Next, the ray bounces off the opaque object B3 and finally hits B4. To complete the drawing we draw the light rays from the light source to each point hit by the eye ray.



   (b) (2 points) Assume the transparency coefficient α for the transparent objects is .5, the light intensity is $I_p = (1, 1, 1)$ (no other lights), and the diffuse/specular coefficients for the objects are $k_d^1 = (0, 0, 1), k_s^1 = (0, 0, 0), k_d^2 = (0, 0, 0), k_s^2 = (1, 1, 1), k_d^3 = (0, 0, 0), k_s^3 = (1, 1, 1), k_d^4 = (1, 0, 0), k_s^4 = (0, 0, 0)$. What is the color returned by the ray tracing algorithm for ray R?

   Solution Let L_i and N_i denote the light direction and normal direction on the surface at the hit point for object B_i. To get the color returned by the ray tracing algorithm, we work out way backwards starting with the contribution from color at the hit point on object B_4.

$$color_{B_4} = k_d^4 * I_p * (L_4 \cdot N_4) = (1, 0, 0)(1, 1, 1)(L_4 \cdot N_4) = (arctan(\frac{7}{5}), 0, 0) = (0.58, 0, 0)$$

Tracing the ray backwards to the hit point at B_3, we get:

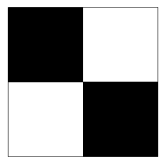
$$color_{B_3} = k_s^3 * color_{B_4} = (1,1,1)(0.58,0,0) = (0.58,0,0)$$

Estimating the angle between $L_1 and N_1$ to around 70 degrees . The final color is then:

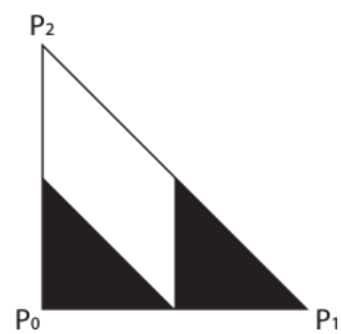
$$color_{final} = (1-\alpha) * color_{B_3} + \alpha * k_d^1 (L_1 \cdot N_1) \approx$$
$$0.5 * (0.58,0,0) + 0.5 * (0,0,1) * (1,1,1) * 0.3 \approx (0.3,0,0.15)$$

3. Texture Mapping.

   (a) (3 points) Given the following texture defined over a unit u,v square.



draw the textured triangle with the following vertex and texture coordinates
$P_0 = (0, 0, 0); UV_0 = (0, 0);$
$P_1 = (2, 0, 0); UV_1 = (1, 1);$
$P_2 = (0, 2, 0); UV_2 = (0, 1);$
Solution
Taking the top left corner as having UV coordinates (0,0), we get:



The colors are inverted if we denote the bottom left corner as having UV coordinates (0,0).

(b) (2 points) The texture below is stored in a 4×4 "texel" array. How will this texture look when mapped to a square of 3×3 pixels with no mipmaping or other antialising techniques used? Draw and explain.

Solution

We pick the color according to the center of the pixel - there is no anti-aliasing so we do not blend the colors. For squares where the center lies on the boundary of white and black, we pick a consistent way to tie-break. If we always take the left top corner color, we get: