

CPSC 414 Final Exam

April 23, 2004

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.

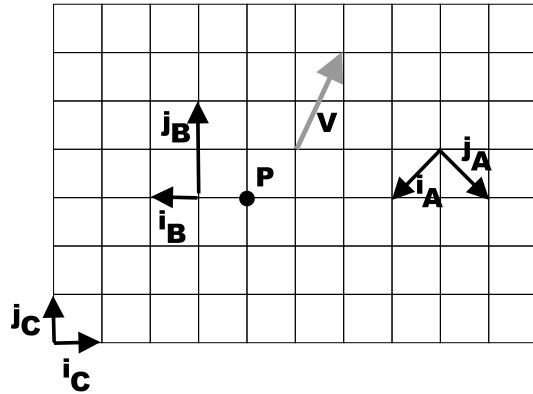
Name: _____

Student Number: _____

Question 1	/ 6
Question 2	/ 9
Question 3	/ 6
Question 4	/ 5
Question 5	/ 6
Question 6	/ 15
Question 7	/ 6
TOTAL	/ 53

This exam has 7 questions, for a total of 53 points.

1. Coordinate Frames



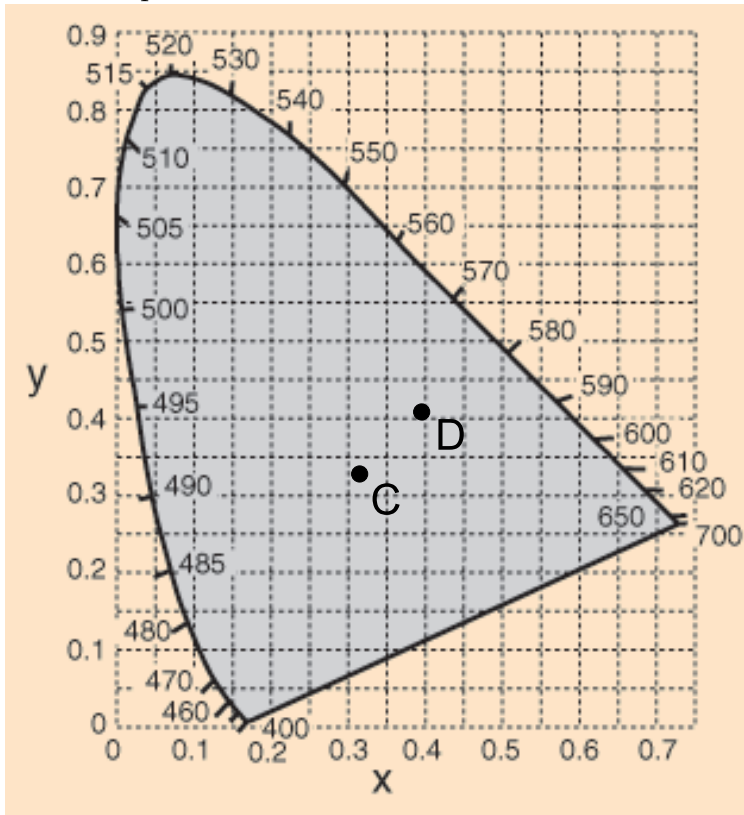
(a) (2 points) Express point P and vector V in each of the two coordinate frames.

(b) (2 points) Find the 3×3 homogeneous transformation matrix which takes a point from frame A coordinates and expresses it in terms of frame B coordinates. I.e., determine $M_{A \rightarrow B}$, where $P_B = M_{A \rightarrow B} P_A$.

(c) (2 points) Suppose that coordinate frame C as shown on the figure is the current coordinate frame. The following OpenGL transformations are then applied. Sketch the new coordinate frame on the figure and label it as coordinate frame D .

```
glTranslatef(3,1,0);
glRotatef(90,0,0,1);
glTranslatef(1,0,0);
```

2. Colour Representation



- (a) (2 points) Sketch a typical RGB monitor gamut on the CIE chromaticity diagram shown above. Label each of the R, G, and B locations.
- (b) (1 point) Label the area in the CIE chromaticity diagram that contains “non-spectral” colours.
- (c) (2 points) What is the dominant wavelength of the colour D on the chromaticity diagram? Which spectral colour is complementary to D?
- (d) (1 point) Compute and illustrate where the colour $(X, Y, Z) = (10, 20, 10)$ would be located on the CIE chromaticity diagram.
- (e) (1 point) Suppose a colour sent to a printer has $(C, M, Y, K) = (0.7, 0.5, 0.6, 0.2)$. Compute the approximate RGB values corresponding to this colour. Use the simple RGB-CMYK conversion model discussed in class.

- (f) (1 point) What are metamers?
- (g) (1 point) Describe how one could take an RGB colour on one monitor and reproduce it on another RGB monitor that has slightly different phosphors.

3. (6 points) Local Illumination

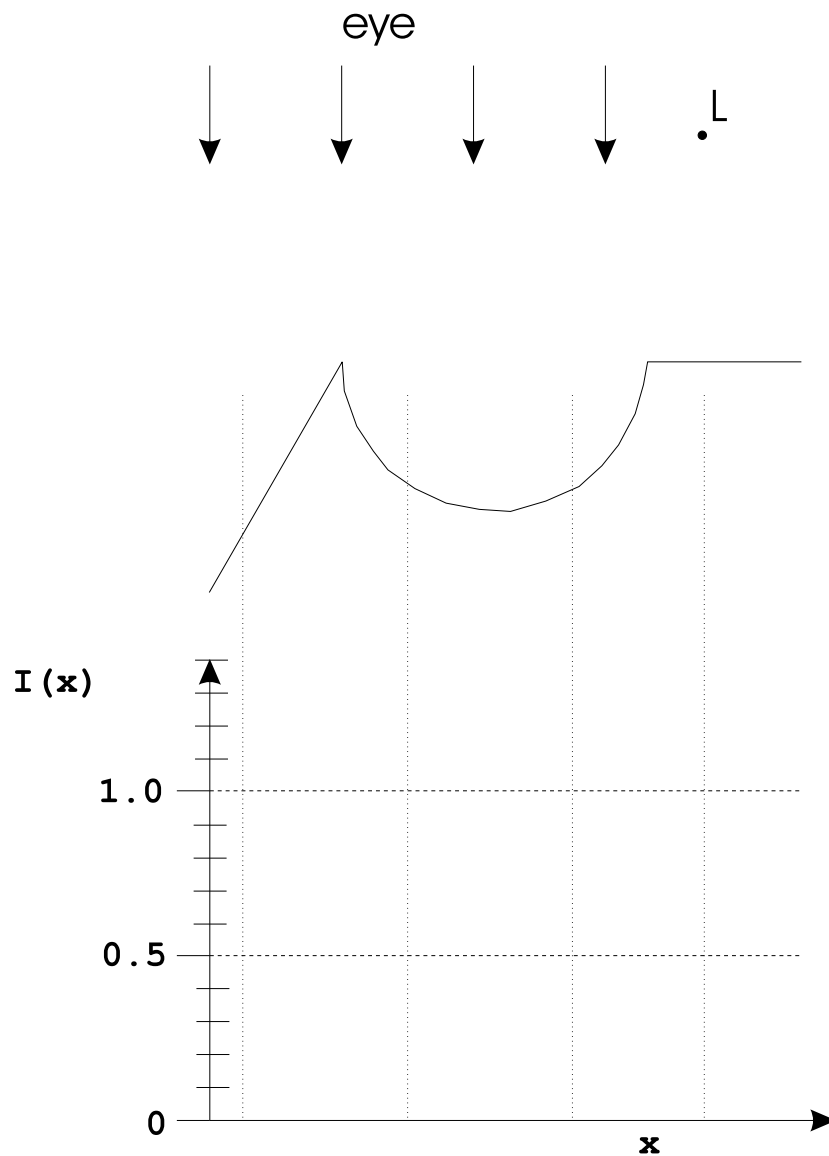
Sketch the illumination that would be computed for the following scene using the Phong illumination model. The scene is viewed from above using an orthographic projection and is lit by the single light source L . Draw 4 sketches, one for each of ambient, diffuse, specular, and total illumination.

The Phong illumination model is given by:

$$I = k_a I_a + k_d I_d (N \cdot L) + k_s I_s (R \cdot V)^n$$

and the values of the various parameters are:

$$I_a = I_d = I_s = 1.0, k_a = 0.2, k_d = 0.8, k_s = 0.7, n = 200.$$

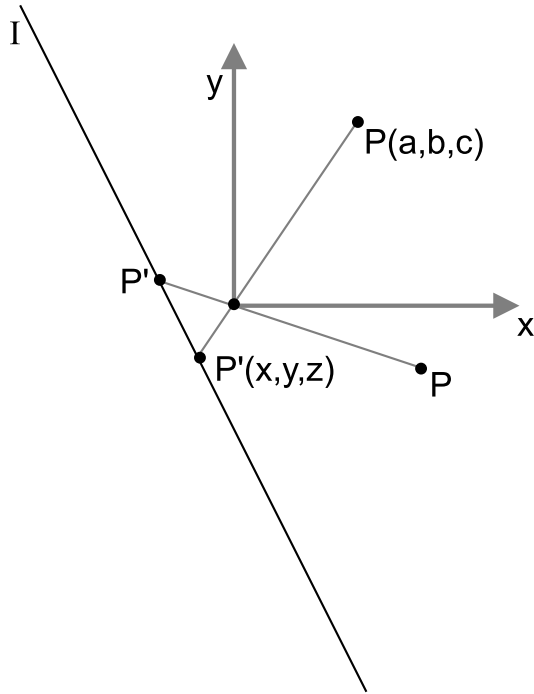


4. (5 points) Projection matrices

Suppose we wish to perform a projection of 3D points onto the image-plane I , as shown below. Assume that the plane equation for I is known and is given by

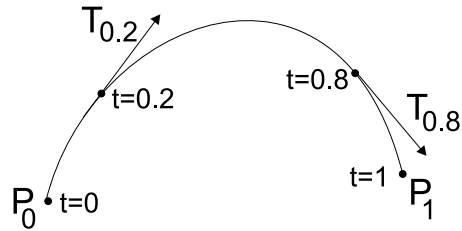
$$Ax + By + Cz + D = 0.$$

The center of projection is the origin. Show how to compute the projected point, $P'(x, y, z)$ for any point $P(a, b, c)$. Give the 4×4 homogeneous projection matrix that could be used to perform this transformation.



5. Parametric Curves

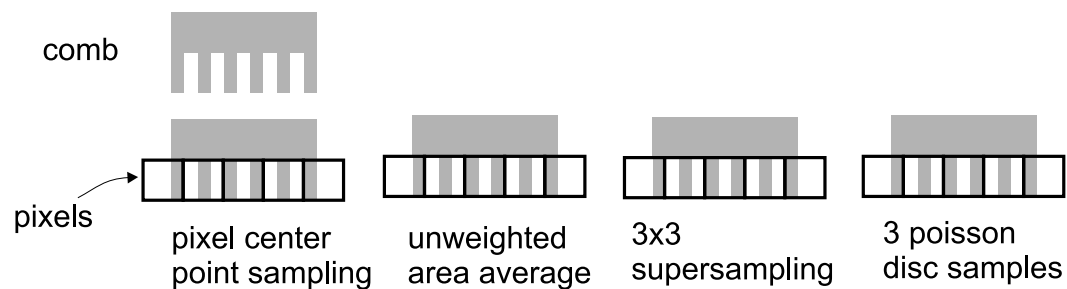
- (a) (4 points) Derive the basis matrix M for the following cubic parametric curve, which is specified using two points and two tangent vectors: $P_0, T_{0.2}, T_{0.8}$ and P_1 . Use this same order when specifying your geometry vector. Do not bother with numerically inverting matrices.



- (b) (2 points) For a given curve and an efficient implementation, how many floating point multiplies and additions are needed to compute points on the curve for 1000 different values of t .

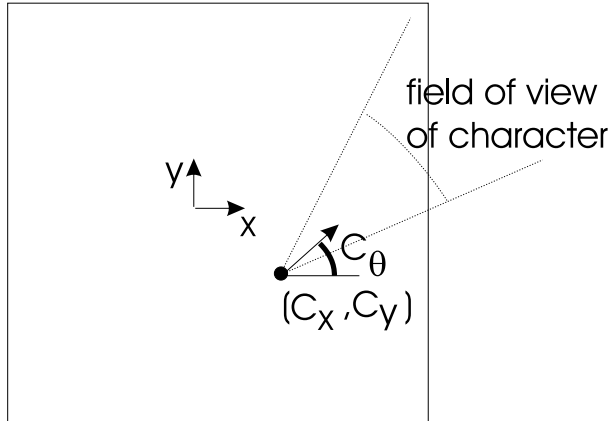
6. General Questions

- (a) (1 point) Give 3 applications of particle systems.
- (b) (1 point) True or False: Decreasing the field of view in a perspective view frustum will produce a zooming (magnification) effect in the resulting image.
- (c) (2 points) Give implicit, explicit, and parametric equations for a circle of radius r centred at (x_0, y_0) .
- (d) (1 point) True or False: Normal vectors can be rotated and translated by using the same rotation and translation matrices as used for points, but using $h = 0$ instead of $h = 1$.
- (e) (4 points) Give the approximate result that would be produced by rendering a black comb ($I = 0$) on a white background ($I = 1$) by applying each of the following schemes to the following row of pixels. Indicate your sample points on the diagram, and indicate the final pixel intensity I that would be assigned for each pixel. Make reasonable assumptions as needed.



- (f) (3 points) Describe in general terms how shadows are (or are not) computed using each of (A) projective rendering with local illumination; (B) radiosity; and (C) raytracing.
- (g) (1 point) True or False: Two BSP trees for the same scene will always produce the same back-to-front ordering from a given eye viewpoint.
- (h) (1 point) True or False: The Phong illumination model can be used to accurately model the real physical reflection properties of nearly all materials.
- (i) (1 point) Ray tracing a scene with only diffuse objects will produce the same resulting image as projective rendering using OpenGL, assuming that the same local illumination parameters are used. Is this true? Why or why not?

7. (6 points) You are developing a game where the player controls a character that is walking through the forest. The game is played from a “first person” perspective, meaning that the game player sees what the eyes of the character would see. The character has a known position C_x, C_y and a known facing angle C_{theta} as shown in the figure below. The game-play is controlled using four keys that make the character take $1m$ steps forwards and backwards, and make the character turn left and right in 10 degree increments. The forest and the animals that inhabit it can be drawn using a call to `DrawWorld()`, which draws all the trees and animals in the world according to their world coordinates. There are moving animals in the forest and they can update their position with a call to the `UpdateWorld()` function. The game is written using OpenGL and glut.



Your job is to write the code for the 7 functions given on the following page. You can assume that the window has already been opened and that the projection matrix has already been setup. The list of functions that you will need is given below. You can assume that C_x, C_y, C_θ are available as globals and you can create other globals as needed.

List of functions:

```
glClear( GL_COLOR_BUFFER_BIT )
glLoadIdentity()
glMatrixMode(GL_PROJECTION)
glMatrixMode(GL_MODELVIEW)
glutSwapBuffers()
glTranslatef(x,y,z)
glRotatef(theta,0,0,1)
glutPostRedisplay()
glPushMatrix()
glPopMatrix()
gluLookAt(x_eye, y_eye, z_eye, x_lookat, y_lookat, z_lookat, x_up, y_up, z_up)
DrawWorld()           // draws the world
UpdateWorld()         // updates animal positions in the world
```

```
init() { // initialization, called once at the start

}

idle() { // glut Idle callback

}

display() { // glut display callback

}

keyForw() { // glut call back for "move forward 1m" key press

}

keyBack() { // glut call back for "move backward 1m" key press

}

keyLeft() { // glut call back for "turn left 10 degrees" key press

}

keyRight() { // glut call back for "turn right 10 degrees" key press

}
```