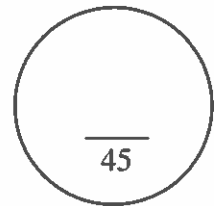


Student ID:	
Full Name:	Solutions

Signature:	
------------	--

Midterm



University of Toronto

*CSC418H
Computer Graphics*

Winter 2015

Instructor: Luke Moore

*Tuesday, February 24, 2015
Duration: 60 minutes*

This is a closed book exam (no calculators, PDAs, laptops or other examination aids)

Note: It is a violation of the University's Code of Behaviour on Academic Matters to cheat on an examination. Cheating includes copying another student's work, possessing unauthorized examination aids, having someone else take your examination, or knowingly allowing your work to be copied.

Question 1: Consider the 3D implicit surface defined by the following equation:

$$f(x, y, z) = 2x^2 + 3y^2 - xy + xz - 2yz + 2x + y - z + 9 = 0$$

0 or 1 mark

a) [1 marks] Find the point on the surface at $x=1$ and $y=1$.

$$f(1, 1, z) = 2 + 3 - 1 + z - 2z + 2 + 1 - z + 4$$

$$= -2z + 16$$

$$\text{When } f(1, 1, z) = 0, \quad z = 8.$$

Thus, the point is $(1, 1, 8)$

b) [4 marks] Compute the surface normal at the point in a). You do not have to express it as a unit vector.

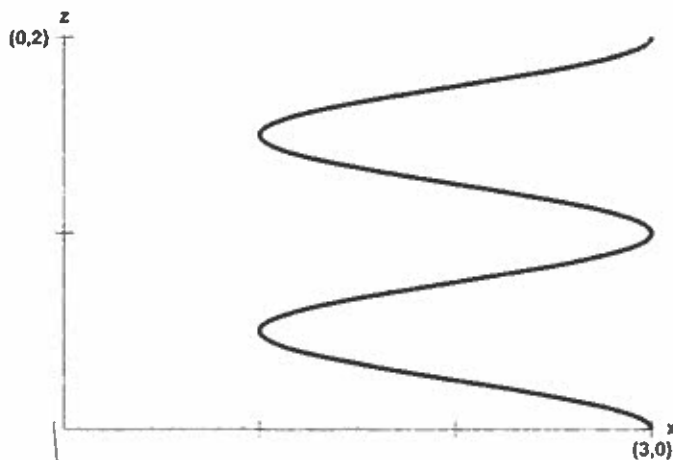
$$\nabla f(x, y, z) = (4x - y + z + 2, 6y - x - 2z + 1, x - 2y + 1)$$

$$\vec{n} = \nabla f(1, 1, 8) = (4 - 1 + 8 + 2, 6 - 1 - 16 + 1, 1 - 2 + 1)$$

$$= (13, -10, -2)$$

Question 2:

a) [4 marks] Give a parametric representation for the following 3D curve that lies on the x - z plane. State the domain of the curve parameter.



$$x(t) = \cos(2\pi t) + 2$$

$$y(t) = 0$$

$$z(t) = 2t$$

$$t \in [0, 1]$$

① for knowing generic surface of revolution formula

① for rotating about correct axis

① for correct formulas

① for specifying domains

- b) [4 marks] Give the parametric representation for the 3D surface formed by rotating the curve from a) about the z axis. State the domains of the two parameters.

$$x(t, \theta) = \cos \theta (\cos(2t) + 2)$$

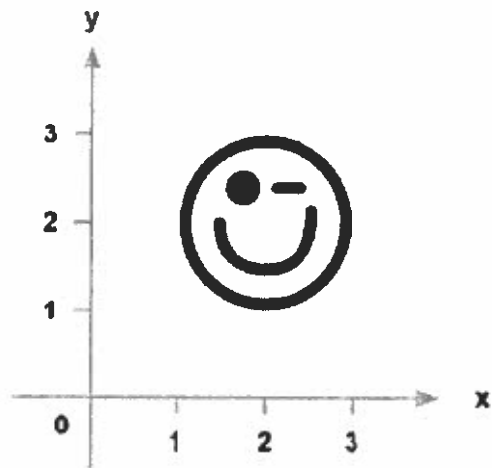
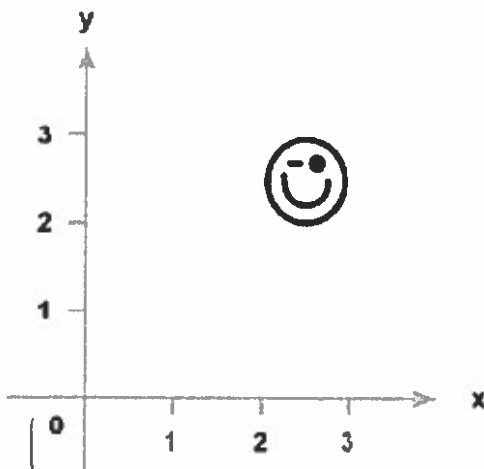
$$y(t, \theta) = \sin \theta (\cos(2t) + 2)$$

$$z(t, \theta) = 2t$$

$$t \in [0, 1]$$

$$\theta \in [0, 2\pi]$$

Question 3: [8 marks] Find the 3x3 homography matrix that transforms the 2D vertices of the smiley face as illustrated below. Express the result as a single matrix.



① for knowing form of translation matrix

① for knowing form of scale matrix

① for knowing how to flip/mirror in y -axis

③ for any sequence of operations which produces the result

② for multiplying the matrices correctly and getting the correct final result

$$\begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -2.5 \\ 0 & 1 & -2.5 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -2 & 0 & 2 \\ 0 & 2 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -2.5 \\ 0 & 1 & -2.5 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -2 & 0 & 7 \\ 0 & 2 & -3 \\ 0 & 0 & 1 \end{bmatrix}$$

Question 4: Consider the following projection matrix:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix}$$

- a) [2 marks] If f is 0.5, what are the 2D Cartesian coordinates of the 3D point $(-12, 24, -3)$ after projecting it with the above matrix?

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \begin{bmatrix} -12 \\ 24 \\ -3 \\ 1 \end{bmatrix} = \begin{bmatrix} -12 \\ 24 \\ -3 \\ -6 \end{bmatrix} \approx \begin{bmatrix} 2 \\ 4 \\ 1/2 \\ 1 \end{bmatrix} \quad \text{The 2D coordinates are } (2, 4).$$

- b) [3 marks] What is pseudo-depth and how does OpenGL use it?

It is the z value in canonical view space.
OpenGL uses it to compute visibility on a pixel by pixel basis.

- c) [3 marks] What pseudo-depth values does the above projection matrix produce? Why isn't this matrix suitable for use with a z -buffer algorithm?

All points will have a pseudo depth of $1/f$.
It can't be used to distinguish which projected pixel values are closer to the camera.

- d) [4 marks] Modify the matrix so it produces a range of pseudo-depth values.

Change the matrix to:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1/f & 0 \end{bmatrix}$$

Then (x_1, y_1, z_1) projects to

$$\begin{bmatrix} x \\ y \\ z+1 \\ \frac{z}{f} \end{bmatrix} \approx \begin{bmatrix} \frac{xf}{z} \\ \frac{yf}{z} \\ \frac{fz+1}{z} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{xf}{z} \\ \frac{yf}{z} \\ f + \frac{1}{z} \\ 1 \end{bmatrix}$$

Question 5: [12 marks] Consider the following three-link, two-dimensional excavator arm determined by arm lengths l_1 , l_2 , and l_3 and angles θ_1 , θ_2 , and θ_3 :

- Its base is affixed to the origin of the world coordinate system.
- Its first link has length l_1 and can rotate about the angle θ_1 as in the diagram.
- Its second link has length l_2 and can rotate about the angle θ_2 .
- Its third link has length l_3 and can rotate about the angle θ_3 .

① for the general idea of trans, rot, trans, rot, trans, then rot

② for expressing the last operation first in the matrix product

③ for correct rotations involving θ_2 and θ_3

④ for correct rotation involving θ_1

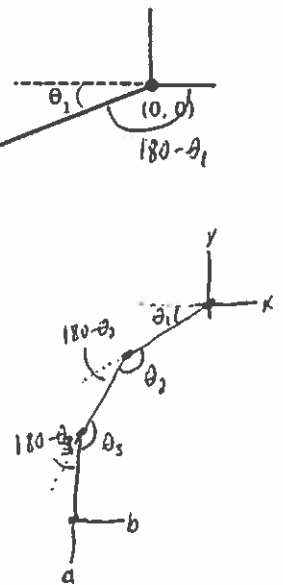
⑤ for proper form of rotation matrices

⑥ for translating in x

⑦ for translating by the proper amount and direction

⑧ for the correct order of transformations involving l_1, l_2, l_3

⑨ for overall correct final matrix product



Suppose we attach a local coordinate system with basis vectors (a, b) to the end of the bucket, as shown in the figure above. Suppose also that point p is expressed in coordinates with respect to this local coordinate system. Determine the transformation that maps local coordinates to world coordinates, and express it as a product of elementary transformation matrices. You do not need to perform any matrix multiplications.

Think of the series of transformations required to transform the (a, b) axes to the world's (x, y) axes. Observe that to transform an axis we need to transform the point by the opposite transformation.

The series of transformations for point p is then:

translate by l_3 in x , rotate by $180^\circ - \theta_3$, translate by l_2 , rotate by $180^\circ - \theta_2$, translate by l_1 , then rotate $\theta_1 - 180^\circ$

The desired matrix product is:

$$\begin{bmatrix} \cos(\theta_1 - 180^\circ) & -\sin(\theta_1 - 180^\circ) & 0 \\ \sin(\theta_1 - 180^\circ) & \cos(\theta_1 - 180^\circ) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & l_1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(180^\circ - \theta_2) & -\sin(180^\circ - \theta_2) & 0 \\ \sin(180^\circ - \theta_2) & \cos(180^\circ - \theta_2) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & l_2 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(180^\circ - \theta_3) & -\sin(180^\circ - \theta_3) & 0 \\ \sin(180^\circ - \theta_3) & \cos(180^\circ - \theta_3) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & l_3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$