CS559 Midterm Exam

October 28, 2014, 7:15pm-9:15 pm

This exam is closed book and closed notes.

You will have the entire period (until 9:15) to complete the exam, however it was designed to take less time.

Please write your name and NETID (your account you use on Wiscmail and moodle) on **every page**! (we may unstaple the exams for grading). Please do this first (and check to make sure you have all of the pages)

Write numerical answers in fractional form or use radicals (square root symbols) – we would prefer to see $\frac{\sqrt{3}}{2}$ than .866. You should not need a calculator for this exam, and you may not use one.

Unless otherwise noted, assume that everything is a right-handed coordinate system and that angles are measured counter clockwise. E.g. to find the direction of rotation, point your thumb along the axis and curl your fingers.

If you need extra space, use the back of a page, but clearly mark what everything is. We may look at your work to determine partial credit. Not providing an answer is not the same as giving a wrong answer. You are penalized for wrong answers on a True/False question.

If you feel a question is ambiguous or requires more information for you to make a good answer, you can state any additional assumptions you needed to arrive at your answer. However, you should probably avoid this: usually it is more a sign of a lack of thought about the question or understanding of the material.

Total : 100 points

Question 1: Lighting (10 parts, +1 for each correct, -1 for each wrong)

While we didn't mention it in class, you should know from the readings that a "directional light source" is one where all rays from the source are parallel (in the same direction). This is in contrast to a point light source, where all light rays come from a single point location.

Consider a sphere lit from above with a directional light source (so the light is pointing downwards). Consider using the Specular/Diffuse/Ambient lighting model (sometimes called the Phong model) as used by OpenGL or discussed in class and the readings.

You should also know that for a sphere, the normals always point outward (the normal at a point is the vector from the center to the point).

True or False (for each one):

A. **FALSE** If the sphere is a purely **diffuse** material, it will appear the same brightness everywhere.

B. **TRUE** If the sphere is a purely **diffuse** material, there will be a single point that will be the brightest.

C. **TRUE** For any specific point on the top half of the sphere, if the sphere is a purely **diffuse** material, that point will appear the same brightness no matter what direction you look from.

D. **FALSE** For any specific point on the top half of the sphere, if the sphere is a purely **specular** material, that point will appear the same brightness no matter what direction you look from.

E. **TRUE** If the sphere is a purely **specular** material, the brightest point depends on where you look from. **(pathology: if you look from the bottom directly upwards, the sphere will be all black, so the brightest point is not unique)**

F. **FALSE** If the sphere is a purely **diffuse** material, the brightest point depends on where you look from. **(pathology: you may not see the brightest point)**

G. **FALSE** If you raise the **ambient** term, the position of the brightest spot on the sphere may change (i.e. a different point on the sphere may be the brightest).
(if you explicitly said saturation could cause the whole sphere to be max brightness, we didn't mark it right or wrong – although, it's more wrong than right since the point will still be brightest, even though it's tied, and it's a big assumption on how max brightness is handled)

The following questions are about the lighting model, not just for this sphere:

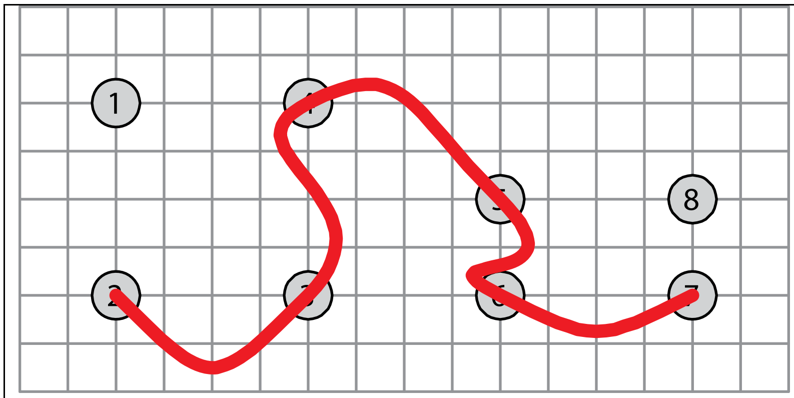
H. **FALSE** This lighting model includes shadows.

I. **FALSE** This lighting model includes reflections of other objects. (freebie)

J. **TRUE** This lighting model treats each light source separately, but we can combine them by adding their contributions together.

Question 2: Interpolating Cubics (13pts = 8 pts + 5*1 pts)

2A. Sketch a Catmull-Rom (e.g. Cardinal Cubic with tension 0) spline through the following sequence of points:



8 points

1. curve starts at 2 (not 1)
2. curve ends at 7 (not 8)
3. curve goes below 2,3 and 6,7
4. 6,7 hump is smaller than 2,3
5. curve crosses between 5 and 6
6. curve croses between 3 and 4
7. curve goes below 4 (on the right)
8. curve interpolates all 6 points

True or False +1 for correct, -1 for wrong:

TRUE If you were to move point 2 or 3 to the bottom edge of the grid, the curve would extend beyond the grid.

TRUE If you moved point 1 upwards enough, the curve would go off the grid between 2 and 3

FALSE If you moved point 1 upwards enough, the curve would go off the grid between 3 and 4 (freebie)

Imagine converting the curve segment between 2 and 3 to a cubic Bezier segment, True or False:

TRUE Two points of this segment would be lower (below on the diagram) than points 2 and 3

TRUE If the control points of this segment were on the grid, then the curve segment will be on the grid.

Question 3: Bezier Curves (7 pts)

A Cubic Bezier Curve has its control points at (0,0), (8,0) (8,12) (16,12).

What is the position of this cubic curve at parameter value u=.25?

DeCastlejau

(0,0) (8,0) (8,12) (16,12)

(2,0) (8,3) (10,12)

(3 ½, ¾) (8 ½, 5 ¼)

(4 ¾, 1 7/8) or (19/4, 15/8)

(I should have picked numbers that worked out evenly – so we gave lots of partial credit if we could tell you did the successive interpolations – but just made arithmetic errors)

NETID: ___________________________

Question 4 Bezier Curve properties (12 parts, 1pt each):

A curve is made from 2 connected cubic Bezier segments (call them A and B).
We'll name the control points A1, A2, A3, A4, B1, B2, B3, B4 (in order).
The curves are connected so that the end of the first segment connects to the end of the second segment. (that is A4=B1).
Other than the 2 points that are the same, the points are strictly increasing in the value of their X coordinate (so A1.x < A2.x < A3.x < A4.x = B1.x < B2.x < B3.x < B4.x).

The Control points are also arranged such that A3, A4, B1 and B2 are all on a line.

While the curves are in 3D, the Z values of all of the points are zero.

For each of the following statements determine if they are:

ALWAYS true (for any positioning of the points subject to the above)
NEVER true (for any positioning of the points subject to the above)
SOMETIMES true and sometimes false (depending on the positions of the points)

(write A, N or S)

A. **Always** The curve is C(0) continuous
B. **Always** The curve is G(1) continuous
C. **Sometimes** The curve is C(1) continuous
D. **Sometimes** The curve is C(2) continuous
E. **Sometimes** If we transform the control points to where they appear on the screen and draw a 2D Bezier curve through those points, it would be the equivalent to us drawing the curve in 3D and figuring out every point gets projected to on the screen, if we were using a standard camera transform and **perspective** projection.
F. **Always** If we transform the control points to where they appear on the screen and draw a 2D Bezier curve through those points, it would be the equivalent to us drawing the curve in 3D and figuring out every point gets projected to on the screen, if we were using a standard camera transform and **orthographic** projection.
G. **Always** The curve lies entirely in the plane Z=0.
H. **Always** The curve crosses the Y=0 plane 5 or less times.
I. **Sometimes** The curve crosses the Y=0 plane exactly 4 times.
J. **Never** The curve crosses the X=0 plane more than once. (freebie)
K. **Sometimes** There exists a quadratic Bezier curve that is the same as curve A.
L. **Always** There exists a quartic (degree 4, or 5 control points) Bezier curve that is the same as A.

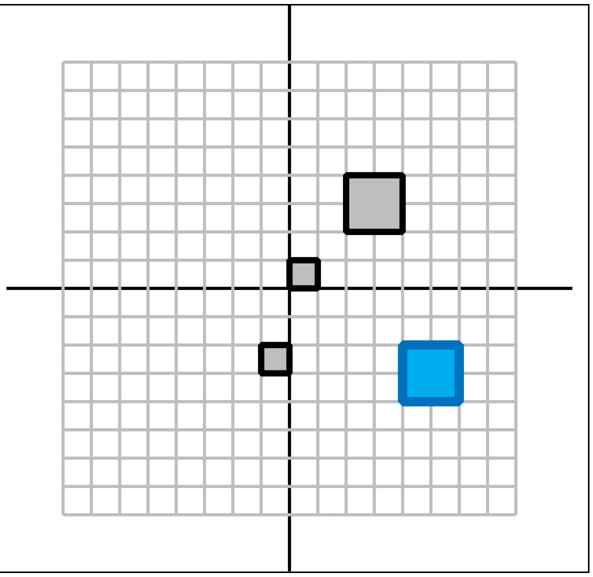
Question 5: Transformations (8 pts)

Imagine a simple OpenGL-like graphics toolkit. All of the transformation operations are like OpenGL in that there is a matrix stack and the command affect the top of it. Rotation is measured in degrees counterclockwise. For each of the little programs below, assume that the programs start out with the identity matrix on the stack.

The draw square command draws a unit square (0,1) in the 1st quadrant.

Here is an example program and its resulting picture:

```
Draw Square
PushMatrix
Scale(2,2)
Translate(2,1)
Scale(-1,1)
Draw Square
PopMatrix
Rotate(90)
Translate(-3,0)
Draw Square
```



A: If we forgot the "Push Matrix" and "PopMatrix" commands in the example program, the last square would appear in a different place. Draw it on the example above.

3 pts

B: If we had forgotten the "PushMatrix" and "Pop Matrix" commands (as in part A), and wanted to "reset" the transformation back to the identity at the end, we could add the following 3 lines of code. (in other words, if we added a DrawSquare command, it would place the square where the first line of the program does

Specify what the values of A, SX, SY, TX, TY should be:

Rotate A	A = -90 (freebie)
Scale SX, SY	SX = -1/2, SY = 1/2 (freebie)
Translate TX, TY	TX = -4, TY=4 (1 pt each – including freebies)

Question 6: Rotations (10 pts, 4+2+ 4*1/-1)

A) A 2D linear transformation is a **rotation** that transforms the point (0,5) to (3, 4). Give the 2x2 matrix for this rotation.

4/5 3/5 **-3/5 4/5**	This was best done as a geometric construction (and knowing there was a 3,4,5 triangle) -1 for getting it transposed (rotation in the wrong direction) -1 for getting the 3 and 4 mixed up +1 for having the right form of a rotation matrix +3 if you thought it was a 30 degree angle, and got everything else about it right (including the negative rotation)

B) Suppose that instead of a rotation, we wanted a 2x2 linear transform that preserves distance (and still maps (0,5) to (3, 4)). Is there an answer other than the one in Part A? If so, give it.

-4/5 3/5 **3/5 4/5**	1 point for realizing there is a mirror reflection (not a rotation since it doesn't preserve handedness) 1 point for figuring out what it is

Matrix Sudoku – In each case, you are given a 3x3 matrix with only some of the values filled in. You know that it's a 3x3 rotation matrix.
True or False (+1 for correct, -1 for wrong):

C) If you know all the numbers in one **row**, and 2 of the numbers on any other **row**, you can figure out the other 4 numbers.

TRUE

D) If you know one **column** of a 3x3 matrix, and you know 1 of the other numbers is 1, you can figure out the rest.

TRUE (freebie)

~~E) If you know the 3 numbers above or below the diagonal, you can figure out the rest.~~

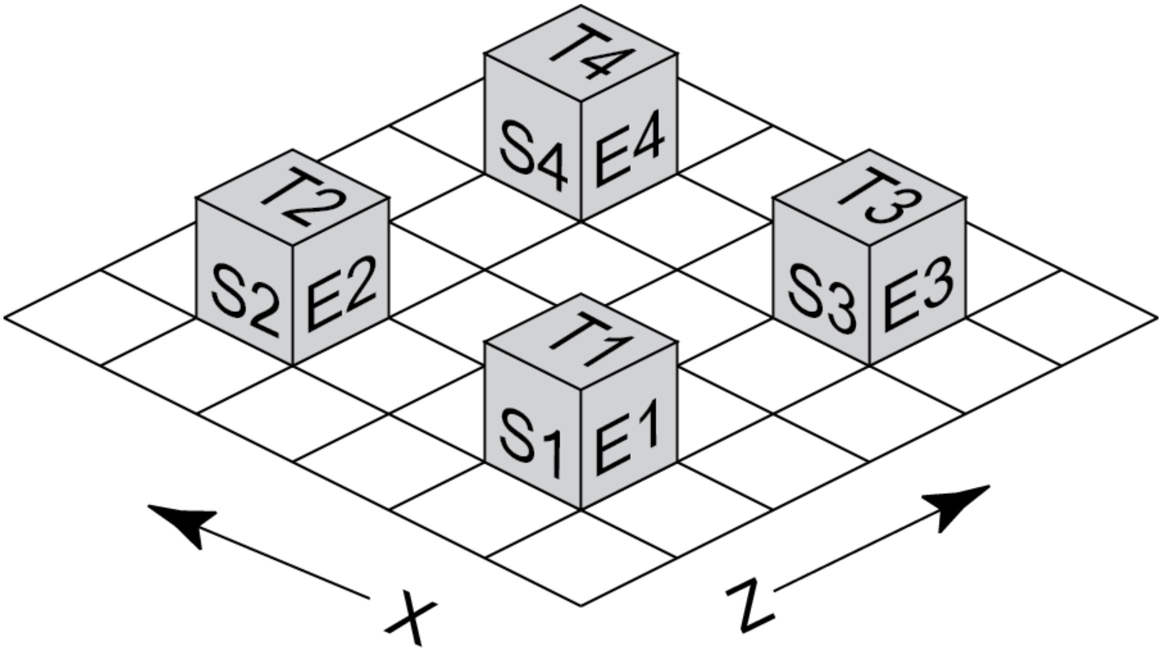
~~TRUE~~ (turns out there is at least one case where this is not true, so we are discarding the question – everyone gets this one correct).

F) If you know any 3 numbers off the diagonal, you can figure out the rest.

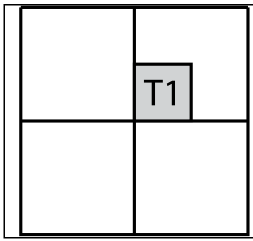
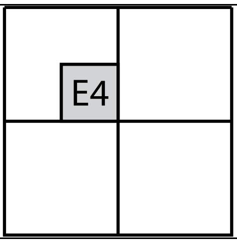
FALSE

Question 7 (10pts)

Four unit size (1 unit cube) blocks are placed on a table. The table top is the Y=0 plane. The origin is the point at the bottom of the picture. So block 1 is at X=1, Z=1. Each side of each block is labeled with a compass direction (N for north, S for south, T for Top) and a number.



Give Lookfrom / Lookat / VUp to make the following pictures (the lines are the X=0 and Y=0 of the picture plane)

	Lookfrom: **2,5,1 (freebie)** Look at: **2,1,1** V-Up: **0,0,1**		Lookfrom: **2,0,5** Look at: **4,0,5** V-Up: **0,1,0**

Note: there are other right answers (anyway along the line containing the answers above)

2pts for each look from and look at (1 pt if one of your numbers is wrong, but reasonable)

1 pt for each Vup

Question 8 (10 pts)

Describe in words what each of these 4x4 homogeneous transformation matrices does to objects (when they get brought back into 3D):

A) $\begin{matrix} 1 & 0 & 0 & 3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$ **Translate by 3 in X**

B) $\begin{matrix} 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 3 \end{matrix}$ **Identity (does nothing) - freebie**

C) $\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{matrix}$ **Scales by 1/3 (makes things smaller) – 1/2 pt for not saying what**

D)

$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{matrix}$	**Perspective Transformation (divides x,y,z by z) – but not the one you're used to since there's a 1 in the lower right (it really divides by z+w). It's still a kind of perspective.** **It is a skew in 4D (but technically not in 3D), so we allow that as a partial credit answer. (full credit if you explain what happens when it returns to 3D**

E)

$\begin{matrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$	**Skew in the X direction based on Z** (OK to say skew in X) Part credit for saying Skew More credit for saying skew Z by X (which is the wrong way)

NETID: ___________________________

Question 9: (4 Points)

For each of the properties, mark P if it is true of a perspective projection, O if it is true of an orthographic projection, B if it is true of both orthographic and projective, and N if it is true of neither. (use P or O only if the property is only true of one type of transformation).

_**B**_ Straight lines are mapped to straight lines

_**B**_ An object can be viewed from its front, side or top

_**O**_ Far away objects appear the same size as closer ones

if you added a note saying an infinite focal length P gives this, we'll give you credit
but, technically this is not a perspective projection – it's orthographic

_**P or B**_ Requires homogeneous coordinates in order for it to be encoded into a linear transformation.

technicality: some orthographic projections require translation, so they need homogeneous

Question 10: Composition (2 Points)

Given a rotation matrix **R** and a translation matrix **T**, the composition $\mathbf{R}^{-1}\mathbf{TR}$ is equivalent to:

a) A rotation
b) A translation
c) A shear
d) A scale
e) None of the above

Question 11: Z-buffer (6 points)

Besides performance (which is a tricky thing, since so many different factors influence it) list two situations where the order that triangles are drawn matter when using a Z-Buffer.

1. Z-fighting (objects with the same / similar Z-values)
2. Transparency

There were some other answers people gave that were partially right (although they tended to be variants of one of the above) – we usually were generous

NETID: ___________________________

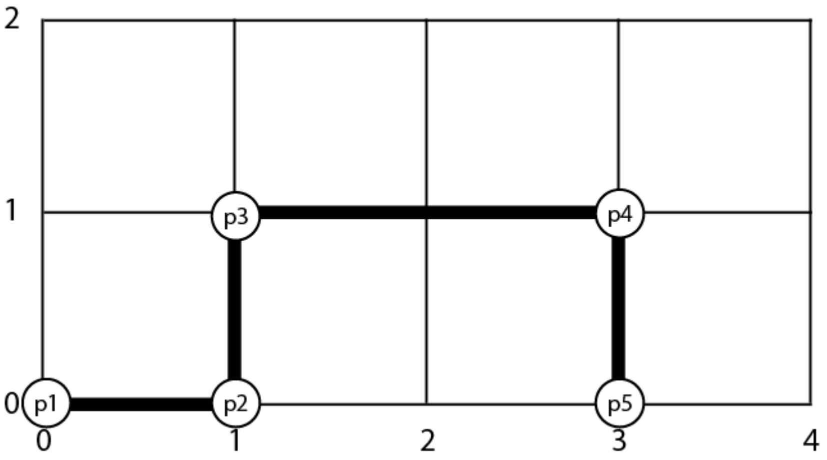
Question 12: Arc-length parameters and blending functions (2+2+4 pts)

Here is a curve.

It is piecewise linear, and interpolates points p1,p2,p3,p4 and p5.

It is arclength parameterized.

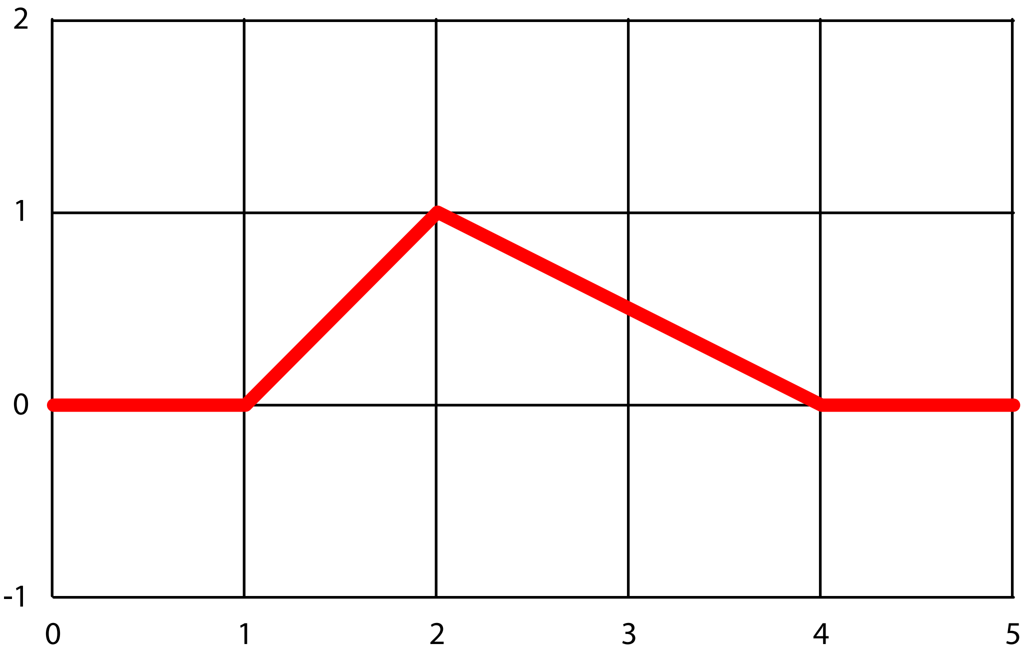
Its parameter range is from 0 to 5.



What is the parameter value when the curve is at (2,1)? **3**

When the parameter has value 1.5, what is the curve's position? **1, ½**

Sketch the blending function for P3:


2
1
0
-1
0
1
2
3
4
5