If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

Cosc 363 Study Doc 2020

Facebook group

Discord Group

SENG301

SENG365

COSC363

COSC364

ENCE361

DATA301

COSC363 2019 - Be careful lol

someone made it so all of the text is actually a link to pornhub lmao

(Please add links to previous study docs if found - much easier than doing every year ourselves)

- **Try to keep all questions in black**
- Green answers are true and verified by others
- Orange answers are likely true but need confirmation
- Red answers are wrong but keep them around

Note: I ripped all the 2017 answers from the 2019 study doc, here's hoping they knew what they were doing.

So is the exam open book I'm confused about the format

Yea its explained in the slides iirc

Can we use google as a calculator since it's online? Bruhhhh

Can I just refer to this doc in the exam

Anyone able to give me a general explanation on what the geometry shader is exactly used for, and why it is used? There should be a question and answer (maybe 2018?) that explains it.

Can anyone tell me what Clip-coordinates are?

2019:

Q1

Question 1. [17 marks for the whole question] *Transformations*

For each of the questions (a)-(c) given below, write a small code segment containing only OpenGL transformation functions for generating the required output.

(a). (3 Marks) Refer to Fig. 1a. In the figure, 'A' refers to the initial display obtained by calling the function `glutSolidTeapot(1)`. Write the code segment required for producing the teapot 'B' located at a point on the *xy*-plane at an angle 30° from the *x*-axis. The distance of the point 'B' from the origin is 3 units (radius of the circular section shown in the figure)

Fig. 1a
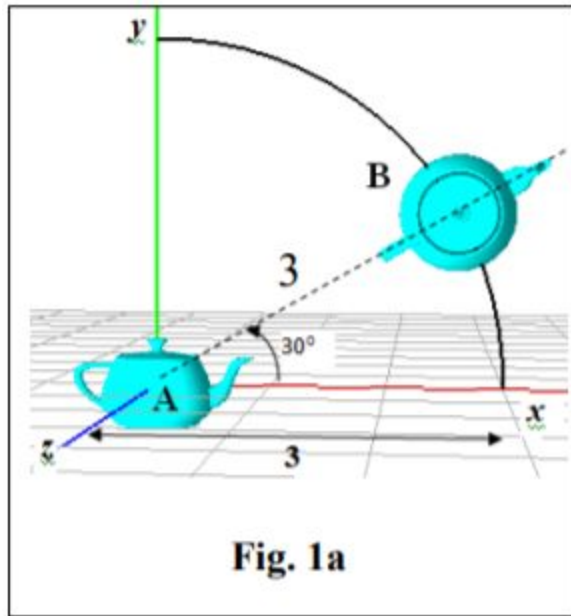
(a)
glPushMatrix();
  glTranslatef(3, 1.5, 0);
  glRotatef(30, 0, 0, 1);
  glRotatef(90, 1, 0, 0);
  glutSolidTeapot(1);
glPopMatrix();

The order of operations is reversed and the translation needs to happen in between the rotations. Like:

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

```
glRotatef(30.0, 0.0, 0.0, 1.0);
glTranslatef(3.0, 0.0, 0.0);
glRotatef(90.0, 1.0, 0.0, 0.0);
glutSolidTeapot(1.0);
```
✓ ✓ ✓ ✓ ✔ ✔ ✓

Yes, you r right, but our result looks the same, cause i move it to (3, 1.5, 0) first, and I check it by running code, they are the same.

Would it not be slightly different because, once rotated, it would not be at x=3 which is where you translate it to after the rotations?

It looks like a little different, I think you are right, so I remove my answer

I'd leave it maybe incase others come across the same issue… Sweet thanks… alright

Shouldn't the x-axis rotation be negative????????????????????
It should be 90, by right hand rule.
In teapot B, am I looking at the top or bottom side of the teapot? Cause I thought it was the bottom… I think it's the top, because the teapot B has a small circle in the center that looks like the top handle
Ahh okay, honest mistake. I'll remember to ask the exam supervisors if I'm unsure. Oh wait...

```
glPushMatrix();
   glTranslatef(3, 1.5, 0);
   glRotatef(30, 0, 0, 1);
   glRotatef(-90, 1, 0, 0);
   glutSolidTeapot(1);
glPopMatrix();
```
Rotation happens at the origin so I don't think you need to translate 1.5 towards y?

glTranslatef values are incorrect, btw. The values that you got means that you are positioning your teapot at x = 3 and y = 1.5. Also, why are you going negative -90 where the top of the teapot is facing toward us. -90 means that the bottom of the teapot is facing toward us.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

(b). (4 Marks) Refer to Fig. 1b. In the figure, 'A' refers to the initial display of an object obtained by calling a user-defined function drawArrow(). Write the code segment required for scaling the object along the x-axis by a factor of 2, and then rotating the scaled object about a pivot point (4, -2, 0) by 30° about the z-axis.

Fig. 1b
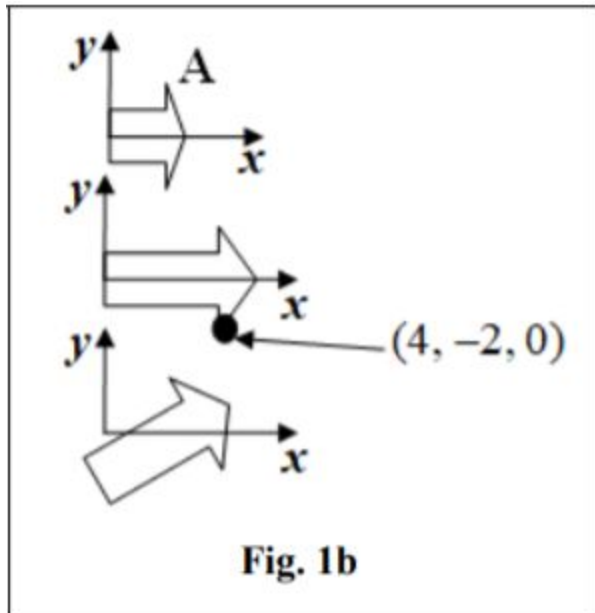
(b)
```
glPushMatrix();
   glScalef(2, 1, 1);
   glTranslatef(4, -2, 0);
   glRotatef(30, 0, 0, 1);
   glTranslatef(-4, 2, 0);
   drawArrow();
glPopMatrix();
```

Just moved the scale to before the translations happen:
Had something similar, but I had the two translates in opposite positions, its probably just me being dumb, but how come they are this way and not the other, I thought it would be glTranslatef(4,-2,0) first to move it to the pivot point, then rotate, then glTranslatef(-4, 2, 0) to move it back? (Going from bottom to top).

Because the point we want to rotate is at (4, -2, 0) and rotation happens at the origin, we first need to move the pivot point to the origin by translating by (-4, +2, 0). It is then rotated and moved back to its original position by translating (+4, -2, 0).
Makes sense now, thanks :)
No problem :)

What's the difference between putting scalef before glTranslatef() and after? In this question, I think the question requires us to scale the arrow first, then rotate it, am I right? Yeah

It results in different outputs. And in the diagram given they scale first. So in this case the scale is put at the end (Due to the reversed order of operations.) so that's why you put scale last, thx

```
glTranslatef(4, -2, 0);
glRotatef(30, 0, 0, 1);
glTranslatef(-4, 2, 0);
glScalef(2, 1, 1);
drawArrow();
```
✓ ✓ ✓ ✔ ✔ ✓ ✔

(c) (5 Marks) Refer to Fig. 1c. In the figure, 'A' refers to the initial display of an object obtained by calling a user-defined function `drawArrow()`. The length of the arrow shaped object is 3 units. Create another copy of the object and reflect it along the x-axis (horizontal flip), and move the objects along the x-axis to produce the display denoted by 'B'. Apply a single rotation by 30° about the z-axis to both objects to get the final output denoted as 'C'. (hint: Use glPushMatrix() and glPopMatrix())

Fig. 1c
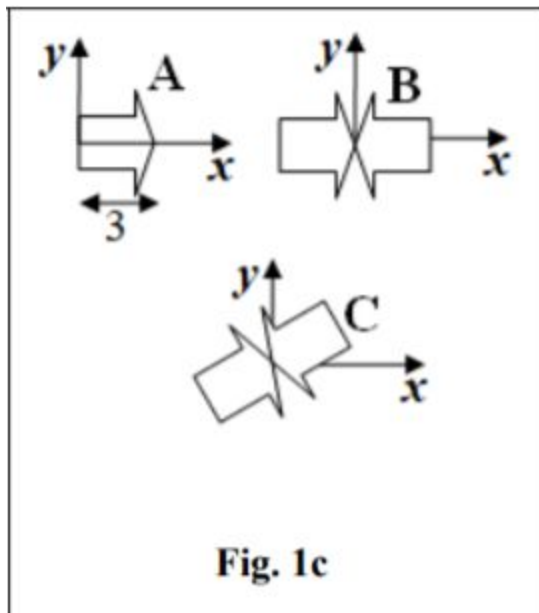
(c)
```
glPushMatrix();
   glRotatef(30, 0, 0, 1);
```

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.
put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

```
glPushMatrix();
   glScalef(-1, 1, 1);
   glTranslatef(3, 0, 0); I think it should be (3, 0, 0)
   drawArrow();
glPopMatrix();

glPushMatrix();
   glTranslatef(-3, 0, 0);
   drawArrow();
glPopMatrix();
glPopMatrix();
```
✓

I got this but I think it's the same result as above.
```
glRotatef(30, 0, 0, 1);
glPushMatrix();
      glTranslatef(-3, 0, 0);
      drawArrow();
glPopMatrix();

glPushMatrix();
      glTranslatef(3, 0, 0)
      glScalef(-1, 1, 1);
      drawArrow();
glPopMatrix();
```
I agree with you, finally it rotates the z-axis ✓ ✓ ✓ ✔ ✔ √

I got this but not sure if right
```
glRotatef(30, 0, 0, 1);
glPushMatrix();
      glTranslatef(3, 0, 0);
      glRotatef(180, 0, 0, 1);
      drawArrow();
glPopMatrix();

glPushMatrix();
      glTranslatef(-3, 0, 0);
      drawArrow();
glPopMatrix();
```

```
glRotatef(30, 0, 0, 1);
glTranslatef(-3, 0, 0);
glPushMatrix();
  glTranslatef(3, 0, 0);
  glScalef(-1, 1, 1);
  drawArrow();
glPopMatrix();
drawArrow();
```

```
glRotatef(30, 0, 0, 1);
glPushMatrix();                    <--would this work?
  translate(3, 0, 0);
  glScalef(-1, 1, 1);
  drawArrow();
glPopMatrix();
glTranslatef(-3, 0, 0);
drawArrow();
```

```
glPushMatrix();
        glRotatef(30, 0, 0, 1);
        glPushmatrix();
                glScalef(-1, 1, 1);
                glTranslate(-3, 0, 0);
                drawArrow();
        glPushMatrix();
        Draw Arrow();
glPopMatrix();
```
Doesnt this one have one of the arrows not translated?

(d) (4 Marks) A point (x, y, z) is transformed using a set of matrices as follows:

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Please describe the sequence of transformations (including the types of transformations, parameters, and the order of transformations) represented by the above matrix expression.

(d)
glRotatef(arcsin(1), 1, 0, 0);
glScalef(4, 3, 2);
glTranslatef(1, 2, 1);
✓ doesn't this have to be the other way around? Technically the operations are not commutative so we have to rotate first, then scale, then translate. -its asking for the sequence of transformations(right to left).

$$R_x(\theta) = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & \cos\theta & \sin\theta & 0.0 \\ 0.0 & -\sin\theta & \cos\theta & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

How did you get arcsin(1)?ahh i see...

How did you determine it was 90degress of rotation? so the angle theta is -90?

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In exam rotation is specified as   so theta is 90 NOT -90 and rounded from 0.98 to one and 0.16 to 0. -- check this. <- this is what happens when you don't work in degrees. Else, sin will be 1 and cos will be 0.
Or

Step 1: Rotate by 90 degrees along the X axis
Step 2: Scale X by 4, Scale Y by 3, Scale Z by 2
Step 3: Translate X by 1, Translate Y by 2, Translate Z by 1
✓ ✓ ✓ ✔ ✔

(e) (1 Mark) Based on your answer to the previous question (d), what would be the output if the above transformation was applied to the point (0, 0, 0, 1)? You are not required to perform any numerical computation or matrix multiplication.

(e)
glRotatef() and glScalef() will be invalid, but glTranslatef() will still change the coordinate of the point to (1, 2, 1);✔

Or

The rotation and scaling will have no effect as the point is at the origin (0, 0, 0). However the translation (Which happens last) will still move the point by (+1, +2, +1).✔ ✓ ✔ ✔

Would we need to specify the new point as (1, 2, 1, 1)?

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

The following code, for example, creates a rotation matrix that rotates vertices 45° around an arbitrary axis specified by $(1,1,1)$, as illustrated in Figure 4.10.

```
vmath::mat4 rotation_matrix = vmath::rotate(45.0, 1.0, 1.0, 1.0);
```

Different notation from what we've used but we retrieve a vec4. So yes but no if you're specifying how the xyz is translated. I guess it depends on what the marker wants.s

But do you remember the transformation order is from the bottom to the top in the matrix block function ,is it possible that the output with(4,0,2)? No because the order is rotate scale then translate, so xyz will be 0 until translated. -in matrix it's right to left, as the leftmost matrix is the first pushed onto the heap. Hence, the first line of code.

Q2

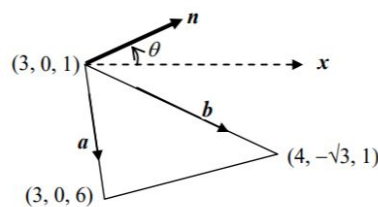**Question 2. [10 marks for the whole question]** *Mathematical Preliminaries*

Fig. 2.

Consider the triangle shown in Fig. 2. The vertex coordinates of the triangle are given in the figure.

(a). (5 Marks) Compute the vectors *a, b*, indicated in the figure, and use them to compute the components of the normal vector *n* of the triangle.

(b). (2 Marks) Normalize the vector *n* to a unit vector.

(c). (3 Marks) Compute the angle θ between the normal vector *n* and the *x*-axis.

(a)
a = (3, 0, 6) - (3, 0, 1) = (0, 0, 5)
b = (4, -sqrt(3), 1) - (3, 0, 1) = (1, -sqrt(3), 0)

n = a*b = (0*0-(-sqrt(3))*5, 5*1-0*0, 0*(-sqrt(3))-1*0) = (5*sqrt(3), 5, 0)
n = (8.66, 5, 0)
✓✔ ✓✔✔✔

(b)
magnitude = sqrt((5*sqrt(3))^2, 5^2) = sqrt(100) = 10
Unit vector n = (5*sqrt(3)/10, 5/10, 0) = (sqrt(3)/2, 1/2, 0)
✓✔ ✓✔✔

(c)
unit b = (1/2, -sqrt(3)/2, 0) - 2 is magnitude
   cos(2*theta) = ub*un = (1/2, -sqrt(3)/2, 0) * (sqrt(3)/2, 1/2, 0) = 0
Isn't the result of this (0, 0, 0.5) not just 0? The z value, should be x1*y2 - x2*y1 (½ * ½ -
(-sqrt(3)/2 * sqrt(3)/2).
we should use Vector Dot Product to calculate cos() like lecture 6 p4.
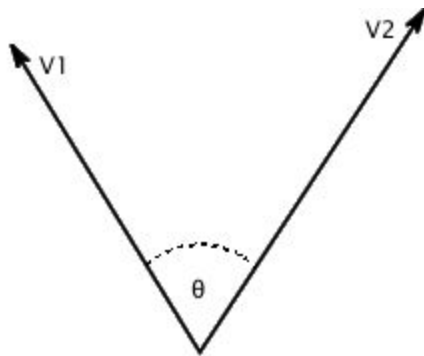Ah yep, my bad.
   2*theta = 90°
   Theta = 45°

Figure 4.2: The dot product — cosine of the angle between two vectors

Mathematically, the dot product of two vectors $v1$ and $v2$ is calculated as

$$v1 \times v2 = v1.x \times v2.x + v1.y \times v2.y + v1.z \times v2.z$$

the value returned falls between -1.0 and 1.0 and is equal to the cosine of
the angle between them. Of course, to get the actual angle between the

vectors, you'd need to take the inverse cosine (or arc-cosine) of this value.

E.g cos^-1(dot product)
= 30 deg

Shouldn't you just calculate the dot product between the normalized normal vector and the x
axis (1,0,0)? How come the vector b is used in the above calculations? We just following the
process described on Lec 6, page 4 of the lec notes. Think you guys have calculated the wrong
angle. What I did was calculate the angle between n and b, and assume that x was halfway so divide
it by two, that may be wrong. I may be being dumb, but how do you know x is (1,0,0) .Isn't (1,0,0)
just the unit vector in the x direction (and therefore can be used for x axis)? ← I agree on this one
✓

My bad i was using 5 root(3) instead of 5 x root(3)

I got theta = 30degrees.
n = (sqrt(3)/2, 0.5, 0)
x = (1, 0, 0)
n.x = sqrt(3)/2 * 1 + 0.5 * 0 + 0 * 0

= sqrt(3) / 2

Theta = arccos(n.x)
    = arccos(sqrt(3)/2)
    = pi/6 rads = 30degrees ✓ ✔✔✔✔

 --why pi/6? Just ignore that, if your calculator is in degree then it just gives 30 anyway. Are you able to explain just out of curiosity? I understand its converting but why 6.. I thought pi/180 gave degrees from radians.

The answer to arccos(sqrt(3)/2) on google gave pi/6 rads which I converted to degrees.
Ah okay, thanks!
✓✓

Q3

Question 3. [10 marks for the whole question] *Illumination Models*

The following figure (Fig. 3) gives the components of unit vectors used for lighting calculations at a point P on a sphere, and also the ambient, diffuse, specular properties of the light L and the sphere material (l = light source vector, n = normal vector, v = view vector, r = reflection vector). Assume that L is the only light source in the scene.

f (Phong's constant) = 10.

Light's properties:
$L_a = (0.2, 0.2, 0.2)$
$L_d = (1, 1, 1)$
$L_s = (1, 1, 1)$

Material properties:
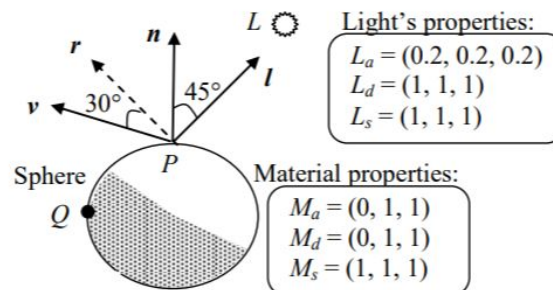$M_a = (0, 1, 1)$
$M_d = (0, 1, 1)$
$M_s = (1, 1, 1)$

Fig. 3.

(a) (2 Marks) Write the colour components of the net reflection from the vertex Q.

(b) (3 Marks) Write the mathematical expression for the colour of the diffuse reflection at vertex P, and compute the numerical values of its components.

(c) (3 Marks) Write the mathematical expression for the colour of specular reflection (including the shininess term) at P along the view direction v, and compute the numerical values of the colour components.

(d) (2 Marks) Write the expression for the half-way vector at P, and also the modified formula for computing the specular reflection using the half-way vector. You are not required to perform any numerical calculation for this question.

(a) La*Ma = (0, 0.2, 0.2) ✓✓✓✔

(b) Ld*Md * max(l.n, 0) = (1, 1, 1)*(0,1,1) * cos(45) = (0, 0.707, 0.707) - think this is it correct me if wrong agree ✓✓✔ ✓✔ Can someone explain how they got this equation?? Nvm found it in lecture note

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

(c)

(d) s! Lec 7 p5-p12

(e) Ls*Ms*(max(r.v, 0))^f = (1, 1, 1)*(1,1,1) * (cos(30))^10 = (0.237, 0.237, 0.237)✓ ✓

Where did you get f = 10 ?

It's "Phong's Constant" next to the diagram - oh i see thanks :)

- think this is it correct me if wrong agree ✓✔ I thought because (1, 1, 1) * (1, 1, 1) is (0, 0, 0), the answer is (0, 0, 0)? Cross product of (1,1,1) and (1,1,1) is (0,0,0) but this is just the multiplication of the vectors right ?? how does 1 * 1 = 0...? Isn't cross product supposed to be the multiplication of two vectors?

AHHH i see what you mean. But this is in relation to the reflection vector r and the view. Cross product calculates the unit vector perpendicular to V1 and V2 or r and v in this case. But we want to calculate the dot product between the view and reflection in this question. Which is given by 30 degress. cos(30) returns this to dot product form. (1,1,1) and (1,1,1) are just the material and lighting color properties. (please discuss if you think i'm wrong)

Where does the vector come from if dot product returns a scalar

- The vectors r and v are already defined from point P.
- We only need to know the angle between these two vertices to alter the color values in respect to the viewpoint (v).

Well we know the material and light properties at p. So as before, we get the net reflection by multiplying these together. And we also know the angle between the reflection and viewpoint so we can alter the color values in relation to the viewpoint with these. instead of calculating any vertices.

- The intensity of specular highlight reduces as the viewer moves away from **r**.
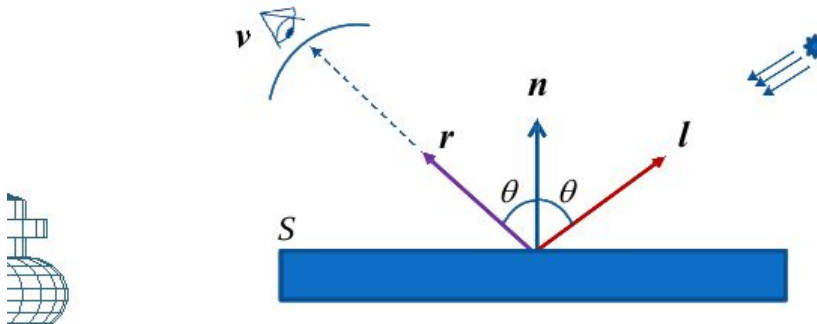
Similar to diffuse reflection, we can write, $I_s = L_s M_s \cos\phi$, where ϕ is the angle between the view vector **v** and the reflection vector **r**.
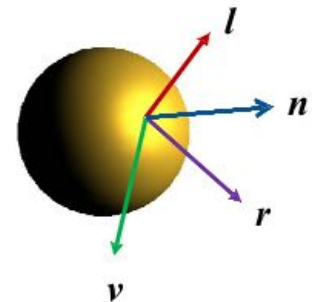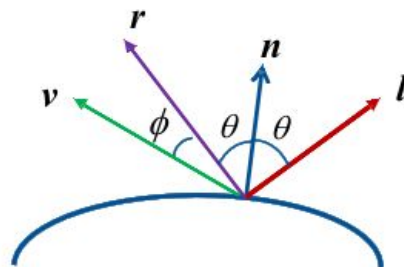
$$I_s = L_s M_s \max(\boldsymbol{r \cdot v}, 0)$$

L_s: Light's specular color
M_s: Material's specular color

Note: **r, v** are unit vectors.

The cross product of two three-dimensional vectors $v1$ and $v2$ can be calculated as

$$\begin{bmatrix} v3.x \\ v3.y \\ v3.z \end{bmatrix} = \begin{bmatrix} v1.y \cdot v2.z - v1.z \cdot v2.y \\ v1.z \cdot v2.x - v1.x \cdot v2.z \\ v1.x \cdot v2.y - v1.y \cdot v2.x \end{bmatrix}$$
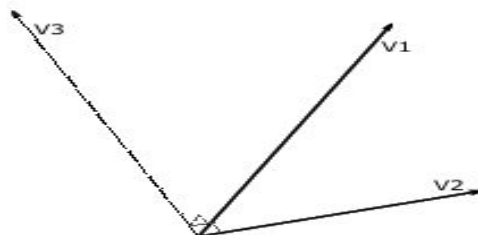
Figure 4.3: A cross product returns a vector perpendicular to its parameters

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

(f) H = (L + V)

Is = Ls*Ms*(max(h*n, 0)^f) ✓✔✔ wrong?with specular reflections are we not dealing with the angle between the view and r?....nothing this explains it.

Is = Ls*Ms*(max(cos(15), 0)^f) <--explain? ✓ theta = 15 ,nl = rn = 45degrees ,rv = 30 ,total = 120 ,half = 60 ,hl = 60 ,hn = hl - nl = 60 - 45 = 15 **This feels excessive for a 2 mark question, that green one is probably better**

thank you!
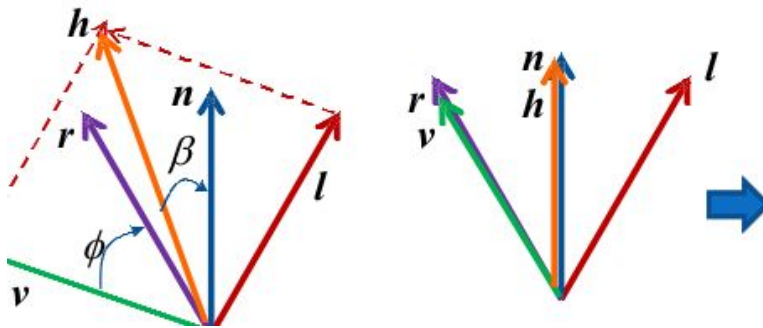
β varies with ϕ, and could be used as a substitute for ϕ.

Q4

Question 4. [12 marks for the whole question] *Texture Mapping*

The polygonal object shown in Fig.4a is created using a quad-strip {A_1, A_2, B_1, B_2, C_1, C_2, D_1, D_2}. The top side consists of two triangles A_2B_2P, C_2D_2Q and a quad PB_2C_2Q. Consider the texture image of size $N \times N$ pixels given in Fig. 4b.
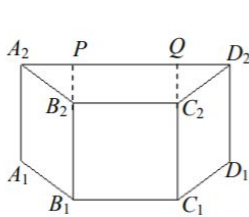

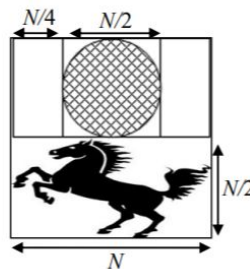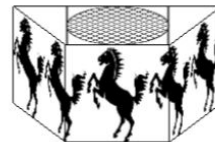Fig. 4a.   Fig. 4b.   Fig. 4c.

(a) (8 Marks) Write the texture coordinates (s, t) that must be assigned to the vertices, so that the quad-strip displays 5 repetitions of the horse image distributed evenly across the three segments; and the quad PB_2C_2Q on the top displays the disc (Fig. 4c). Assume that the texture wrap mode is set to GL_REPEAT along s and t. Please provide your answer in the answer-booklet in the following format:

(a)

Agree with all these

I also agree

Do these not need to be scaled by N?

--length of N = 1. <-just add that statement at the start

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

A1 (0, 0)
A2 (0, 0.5)
B1 (1.67, 0) Where did we get this value of 1.67? O is it 5/3? Cause 5 is the number of times we want to repeat and 3 for the number of quadst. Yep
B2 (1.67, 0.5)
C1 (3.33, 0)
C2 (3.33, 0.5)
D1 (5, 0)
D2 (5, 0.5)
P (0.25, 1)
B2 (0.25, 0.5)
C2 (0.75, 0.5)
Q (0.75, 1)✓ ✓ ✔

- If the wrap parameter for a texture axis is set to GL_REPEAT, then the integer part of the texture coordinate along that axis is ignored. (eg. A value 1.3 is treated as 0.3). This results in the tiling of the image along that axis. [Default]

(b) (4 Marks) Briefly describe (in 3 or 4 sentences) what is meant by "mipmapping", and its usefulness in texture mapping. You are not required to give any equations.

(b)Mipmaps are smaller, pre-filtered versions of a texture image, representing different levels of detail (LOD) of the texture. They are often stored in sequences of progressively smaller textures called mipmap chains with each level half as small as the previous one.
Instead of sampling a single texture, the application can be set up to switch between any of the lower resolution mipmaps in the chain depending on the distance from the camera. (I copy this from internet, not sure if that is correct) ✓ ✓ //

- MIP = Multum In Parvo = "Much in a small place"
- A mipmap is a set of prefiltered versions of the same image at different scales (resolutions)
- The problem of disappearing lines when a texture is mapped to a small region can be solved by using a mipmap.
- Mipmapping requires additional processing, and 33% extra texture storage space.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

Q5

Question 5. [10 marks for the whole question] *View Transformation*
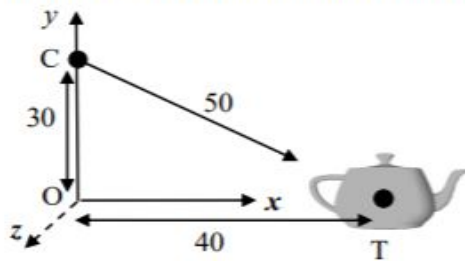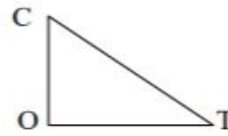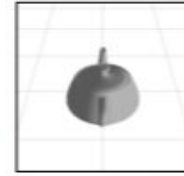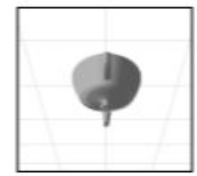

Fig. 5a

Fig. 5b

Fig. 5c

Fig. 5d

A teapot is placed at position T on the *x*-axis at a distance of 40 units from the origin, as shown in Fig. 5a. The camera is placed at a fixed position C along the *y*-axis at a height of 30 units.

(a) (2 Marks) Write the parameter values of the camera function $gluLookAt(e_x, e_y, e_z, l_x, l_y, l_z, u_x, u_y, u_z)$, that places the teapot along the view axis of the camera and generates an output similar to that given in Fig. 5c.

(b) (3 Marks) Redraw the triangle in Fig 5b in your answer booklet and mark the directions of the camera axes x_e, y_e, z_e (eye coordinate frame) at C, where the view axis of the camera is along CT. Indicate an out-of-plane axis with a dashed line.

(c) (2 Marks) The camera is rotated by 180 degrees at position C, keeping the view direction fixed, to generate the output given in Fig. 5d. Write the parameter values of the function $gluLookAt(\ldots)$ for this configuration.

(d) (3 Marks). The teapot is at a distance of 50 units from the camera. Write the coordinates of T and C in both the world coordinate frame and the eye coordinate frame for the camera configuration given in Q.5(a).

(a) gluLookAt(0, 30, 0, 40, 0, 0, 0, 1, 0); agree ✓✔✓✓✔
(b) Wil not include in this exam
Anyone know how to do this one? I thought -Ze is in the direction of CT, Xe is coming out of page towards you and Ye is perpendicular to Ze, but i'm not sure. ✓ ✓

(c) gluLookAt(0, 30, 0, 40, 0, 0, 0, -1, 0); agree ✓ ✔✓✓✔
Would gluLookAt(0, 30, 0, 40, 0, 0, -1, 0, 0); turn the camera around? If -1 was in the y position it would be upside down, in the x position it might turn it around? Cheers, was just curious.

(d) Eye: c(0, 0, 0) t(0, 0, -50) agree✔ Why is the z-value -50? Should it be t(40, -30 0)?
Have a read of lec 7, slide 22. Ok lol thanks why is c(0,0,0)????camera is the origin in eye coordinates

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

World: c(0, 30, 0) t(40, 0, 0) agree ✓✓✓✔✔

Soz but can someone explain question d to my dumbass :(Have a read of lec7 p22, in this question, C is the origin in eye coordinate, and Z axis goes through C and T. ✓

Q6

Question 6. [10 marks for the whole question] *Projection Transformation*

A view frustum is specified using the following statement:

```
glFrustum(-50.0,  50., -30., 30.,  50.0, 100.0);
```

(a) (6 Marks) Draw a sketch of the view frustum, clearly showing what each of the six parameters of the above function represents. The figure should also show the camera position, camera coordinate axes directions, and the view axis direction.

(b) (4 Marks) Using the parameters of the function, compute the aspect ratio of the near plane, and the field of view along the *y*-direction.

(a) Open to critique, sorry it's kinda messy
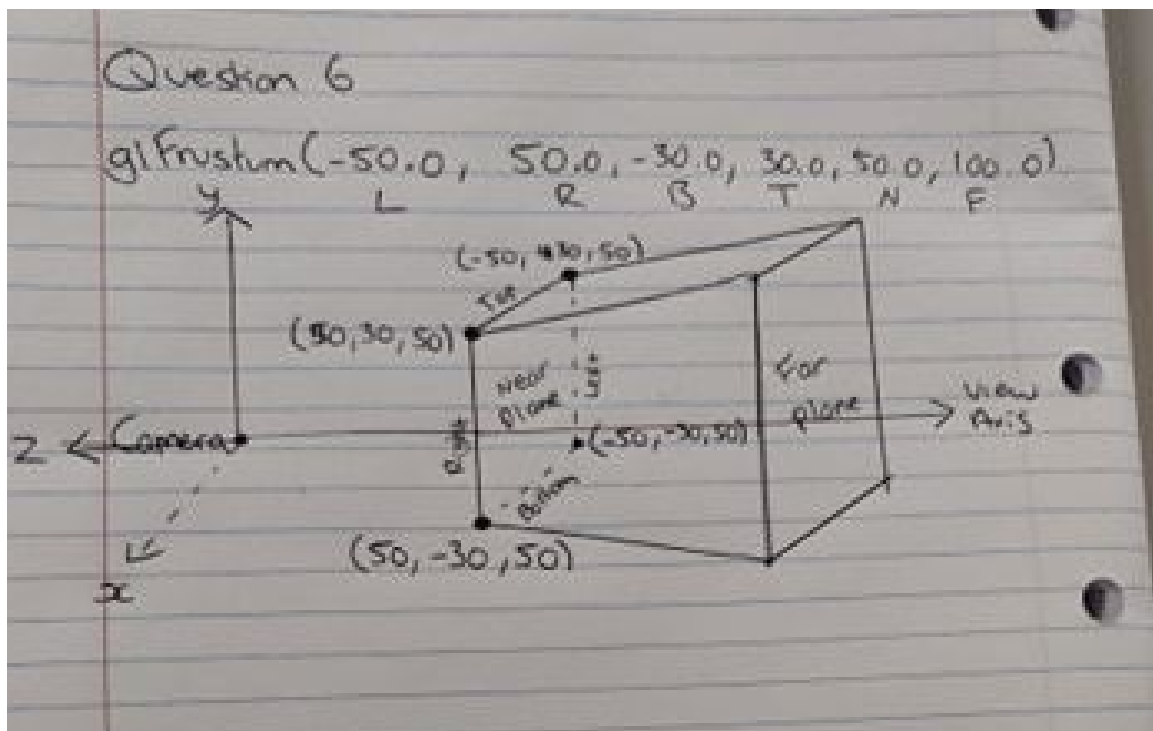
Shouldn't z be negative?? I agree. The view axis is -z and the opposite direction is positive z

(b) Aspect ratio = width/height = 50*2/30*2 = 1.67 agree ✓✓ why do we times 2 here? I thought it would be width/height = (R-L)/(T-B) which is the same thing but

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

wondering why we times 2 width is doubled since it goes 50 in both directions the same goes for height.

FOV = 2*arctan(h/2n) = 2*arctan(0.6) = 61.93deg ✓ ✓ ✓ what's n and where does this equation come from Lecture 6 slide 28 I'm not sure how to use bi-interpolation works in this can someone explain I think they meant lecture 7, slide 28

...how on earth did you get 0.6?

Ive got h = 60 N = 50? Have i done something wrong here?...it must be my N cause i know T -B = 30 - (-30) = 60. And N is the near parameter no?

So 2*tan^-1(60/2.50) = ...ahhh i got it -> i didnt hit equals on the calculator hahah

$$2 \times (\tan_0^{-1}(60 \div (2 \times 50))) =$$

61.927513064147042834215359681673

Without brackets around 2*50 youll get 88.4.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

Q7

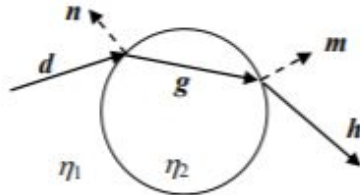**Question 7. [16 marks for the whole question]** *Ray Tracing*
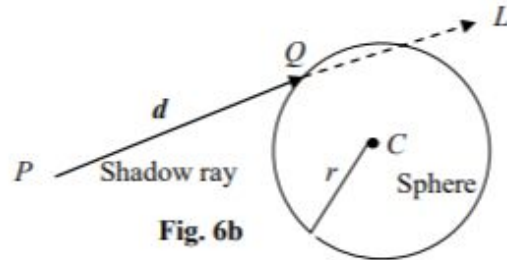

Fig. 6a   **Fig. 6b**

(a) (3 Marks) Fig. 6a shows the refraction of an incident ray *d* through a sphere. The normal vectors are indicated by dashed lines. The refractive indices η of the two mediums are also shown. You are given the function `refract(d, n, eta)` that returns the direction of the refracted ray, given the direction of the incoming ray 'd', and the surface normal direction 'n' at the point of intersection directed towards the incident ray's medium. Write the function statements with appropriate parameters required to compute the outgoing ray's direction *h*. For each refracted ray, give the expression for 'eta' as a ratio of refractive indices.

(b) (3 Marks) Refer to Fig. 6b. A shadow ray is traced from a point *P* towards a light source *L*, and intersects a sphere at *Q*. Write the conditions to be satisfied by *Q* in order that the point *P* is in shadow. Assume that the ray does not intersect any other object.

(c) (6 Marks) The points of intersection of the ray $p = P + td$ and a sphere centered at a point *C* with radius *r* is given by

$$t = -(s \bullet d) \pm \sqrt{(s \bullet d)^2 - (s \bullet s) + r^2}, \text{ where } s = P - C.$$

If $P = (2, 1, 2)$, $C = (10, 7, 2)$, $d = (0.8, 0.6, 0)$, and $r = 4$, compute the values of the ray intersection parameter *t* using the above formula.

(d) (2 Marks) With reference to the previous question Q7(c), write the coordinates of the point of intersection on the sphere that lies closest to *P*.

(e) (2 Marks). Give a geometrical interpretation of the parameters *t*, *s* used in the equation given in Q7(c).

For a), doesn't it say to give the eta as a ratio of refractive indices?
Eta is the ratio of refractive indices. It asks us to write two expressions to show ray g and ray h by using eta. But i am not sure if i am correct
I may be wrong but I interpreted it as say, eta = n1/n2 (refractive index of what its leaving divided by what it enters) this snip from here seems like it supports that, not sure though.

```
float eta=1.0/1.4; // air/glass's index of refraction
vec3 refractDir = refract(rayDir, hit_normal, eta);
```

(a)eta = n1/n2

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

g = glm::refract(d, n, eta);
h = glm::refract(g, -m, 1.0f/eta); ✔✔✔

✓ I think eta = n1/n2 for g and eta = n2/n1 for h is also valid i think so too

(b) PQ < LQ (from lecture slides, not sure if correct) ✓ ✔ this is correct, i asked Ramakrishnan. This is wrong. Slide 11 shows that it should be QP < LP. Is that just not the exact same as what's written in blue?

Not sure if this is implied but do we also have to mention how Q is a solution to both the rays equation and the spheres equation? QP < LP? Also probably a good idea to mention the point of intersection is between the light source

I put (Q - P) . (L - P) > 0, i think thats also correct?

Which lecture slides are these?

CAN SOMEONE PLEASE EXPLAIN WHY ITS PQ/LQ I READ LECTURES NOTES BUT I STILL DON"T UNDETSTAND

(c)s = P-C = (-8, -6, 0)

t = 14 or 6 agree ✓ ✔ ✔

It would just be 6 because thats the closest point, right? 14 is the outgoing point of intersection? It says compute the values of the ray intersection parameter t, it doesn't specify only one in particular, therefore you would give both, because those are the two possible values. ✓

(d) P = p + td = (2, 1, 2) + 6*(0.8,   0.6,   0) = (6.8, 4.6, 2)

(e) t represents the distance of the point of intersection along the ray. s is the vector between the origin of the ray and the centre of the circle.

^^ I Would change the wording to "s is the vector from the centre of the circle to the origin of the ray". The way it's worded implies the vector is pointing at the centre of the circle, whereas I think the vector is actually pointing at the origin of the ray.


Q8

Question 8. [15 marks for the whole question] *OpenGL Shaders*

(a) (4 Marks) Briefly describe two important operations performed in a vertex shader and in a tessellation control shader.

(a) vertex transformation and lighting calculate agree can someone add where we can find notes on this shiz? I'm so confused. ← it's in Lec 11 slide 14, and Lec 9 slide 18 :)

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)
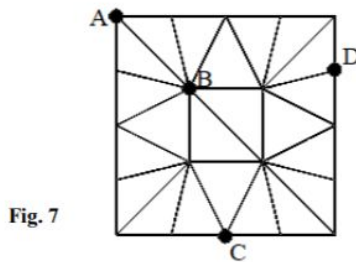
Fig. 7

(b) (6 Marks) A quad domain is tessellated as shown in the figure above (Fig. 7). Write the inner and outer tessellation levels used in generating the figure, and also give the tessellation coordinates of the points A, B, C and D.

(c) (5 Marks) A part of a tessellation evaluation shader is shown below.

```
void main()
{
    float u = gl_TessCoord.x;
    float v = gl_TessCoord.y;
    posn = (1-u)* (1-v) * gl_in[0].gl_Position
         + u * (1-v) *    gl_in[1].gl_Position
         + u * v *        gl_in[2].gl_Position
         + (1-u) * v *    gl_in[3].gl_Position;
    gl_Position = mvpMatrix * posn;
}
```

Give a brief description of each of the four statements in the above code segment, explaining the meanings of the variables gl_TessCoord, gl_in[].gl_Position, posn, and gl_Position.

Agree with all these

(b) outer:4444

Inner:33✔✔✔✔✔✔✔✔ Why are there 2 3's? Is it counting both sides? It goes from one side to the opposite. A square has two sets of opposite sides so it's 33. If it's a triangle, inner has only one digit.
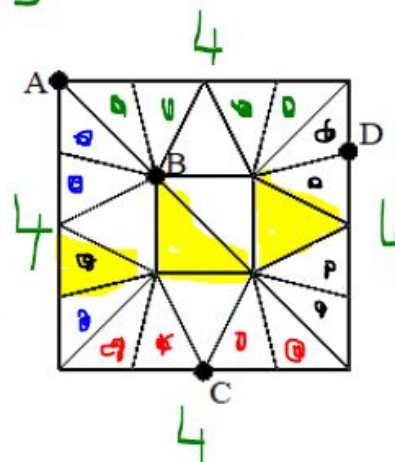
A(0, 1)

B(0.33, 0.66)

C(0.5, 0)

D(1, 0.75)✔

4444; 33

Fig. 7

Can someone explain the coordinates?

Oh nvm I'm just dumb lmao

I disagree, shouldn't it be :

outer: 3333

inner: 33

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

Nah mate: =================== >✔

(c)

gl_TessCoord are the tessellation coordinates of the tessellated mesh vertices gl_in[] contains the patch vertices

posn is computed using the quad domain mapping method which combines a set of vertices which are gl_in[0].gl_Position and so on.

gl_Position holds the homogeneous coordinates of the vertex at gl_in[]

Isn't gl_Position different from gl_In[i].gl_Position ? I think it is the position of the tesselation coordinate in clip coordinates.

Is Mukundan a tits man or an ass man? He is turtle

Please stop sexualising Mukundan he is pure and must be

protected ✓ ✓ ✓ ✓ ✓ rotected ✓ ✓ ✓ ✓ ✓

Tick if you would die for Mukundan ✓ ✓ ✔ ✔
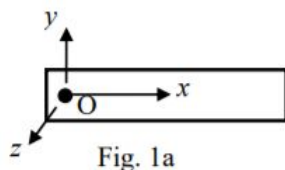
Mukundan got more cake than a birthday party

2018:

Q1:

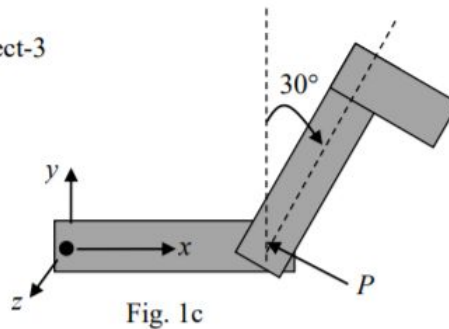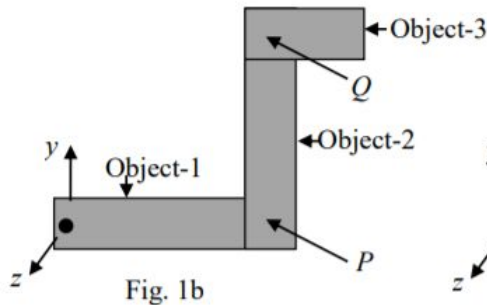**Question 1. [10 marks for the whole question]** *Transformations*

Assume that you are given a function drawObject() that draws a rectangular-shaped axis aligned object shown in Fig. 1a. The origin of the coordinate frame, denoted by O, is located at one end of the object.

Note: z-axis is perpendicular to the paper, towards the reader.

Fig. 1a

Fig. 1b

Fig. 1c

You are also given the following set of instructions for creating the model shown in Fig. 1c:

Step 1: Create the object (Object-1) as in Fig. 1b, by calling drawObject().

Step 2: Create another copy of the object, make its axis vertical (you will need to specify the angle and axis of rotation) and then translate it to the point $P = (xp, yp, zp)$. (Object-2)

Step 3: Create a third copy of the object, scale it by a factor 0.5 along the x-axis, and translate it to the point $Q = (xq, yq, zq)$. (Object-3). Steps 1-3 generate the model shown in Fig. 1b.

Step 4: Rotate the second and third objects together about the pivot point $P=(xp,yp,zp)$ as shown in Fig. 1c. The angle of rotation indicated in the figure is on the xy-plane.

Step 5: Translate the whole model to some point $S = (xs, ys, zs)$ in the scene (not shown in the figure).

Write a code segment containing OpenGL transformation functions and glPushMatrix-glPopMatrix blocks, for generating the model and transformations given in steps 1-5 above.

```
glPushMatrix()                                    //Step 5
        glTranslatef(xs, yx, zs);

        glPushMatrix();                           //Step 1
                drawObject();
        glPopMatrix();

        glPushMatrix();                           //Step 4
                glTranslatef(xp, yp, zp);
                glRotatef(-30, 0, 0, 1);
                glTranslatef(-xp, -yp, -zp);
                glPushMatrix();                   //Step 2
                        glTranslatef(xp, yp, zp);
                        glRotatef(90, 0, 0, 1);
                        drawObject();
                glPopMatrix();

                glPushMatrix();                   //Step 3
                        glTranslatef(xq, yz, zq);
                        glScalef(0.5, 1, 1);
                        drawObject();
                glPopMatrix();
        glPopMatrix();

glPopMatrix();
```
✓ ✔ ✓ ✔ ✓

Q2:

Question 2. [10 marks for the whole question] *Illumination Model*

A certain application uses the following values for the light and material properties:

Light's properties: ambient = (0.2, 0.2, 0.2), diffuse = (1, 1, 1),   specular = (1, 1, 1)
Material's properties: ambient = (1, 0.5, 0.2),   diffuse = (1, 0.5, 0.2), specular = (1, 1, 1).
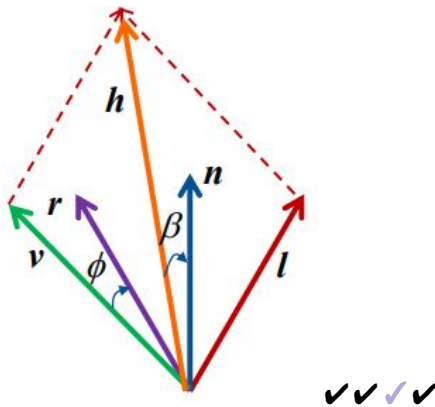
(a) [2 Marks] Write the components of the colour seen at any vertex where $n.l < 0$ (n = normal vector, l = light source vector).

a) Color = ambient(0.2,0.2,0.2) * ambient(1, 0.5, 0.2) = (0.2, 0.1, 0.04) ✓ ✓ ✔ ✔ ✓ ✔

(b) [4 Marks] Draw a figure showing a vertex of a surface, a light source vector, the normal vector, the reflection vector, the view vector and the half-way vector. Also mark the angles used in lighting calculations.

b)

✔✔✓✔

(c) [4 Marks] If the surface normal vector at a vertex is (0.707, 0.707, 0), and the light source vector is (0.8, –0.6, 0), write the colour value for diffuse reflection at the vertex.

$$L_d M_d \max(\boldsymbol{l} \cdot \boldsymbol{n}, 0)$$

Diffuse

Term

c)

max((normalVector * lightSourceVector), 0)
max((0.1414, 0.0707, 0.02828), 0) = (0.1414, 0.0707, 0.02828)

^^I agree with your answer but the working is misleading. The dot product should give a scalar value. Here's what I got.

LdMd = (1,1,1) * (1, 0.5, 0.2) = (1, 0.5, 0.2)

Both l and n are already normalized so we can use them as they are
l.n = (0.707 * 0.8 + 0.707 * -0.6 + 0*0) = (0.5656 - 0.4242 + 0) = 0.1414
max(l.n, 0) = 0.1414

Diffuse = LdMd * max(l.n, 0)
Using the values calculated above
LdMd = 0.1414(1, 0.5, 0.2)✔ ✔✔✔✔✔✔✔

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

= (0.1414, 0.0707, 0.02828)✔ ✔✔ ✓✔

Q3:

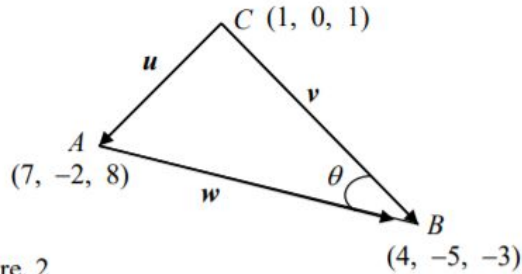**Question 3.** [10 marks for the whole question] *Mathematical Aspects*

Figure. 2

Consider the triangle given in Fig. 2 above, and answer the following questions:

(a) [3 Marks] Write the components of vectors *u* and *v* and show that the vectors are perpendicular to each other.

(b) [5 Marks] Compute the normal vector of the plane of the triangle, and convert the vector to a unit vector.

(c) [2 Marks] Write the mathematical expression for computing the angle θ using the vectors shown in Fig.2. You are not required to compute the numerical value of θ.

a) To show if vectors are perpendicular to each other, v1 dot v2 = 0
    u = A - C = (7, -2, 8) - (1, 0 , 1) = (6, -2, 7)
    v = B - C = (4, -5, -3) - (1, 0 , 1) = (3, -5, -4)
    w = B - A = (4, -5, -3) - (7, -2, 8) = (-3, -3, 11)
    I think w should = (-3,-3,-11) agreed
    u dot v = (6, -2, 7) dot (3, -5, -4)
             = 18 + 10 + -28
             = 0✔✔✓ ✓

b) NX = -2(-4 - (-11)) + -5(-11 - 7) + 3(7 - (-4)) = 109
    NY = 7(0-(-3)) + -4(-3-6) + -11(6-0) = -9
    NZ = 6(-5-3) + 0(3-(-2)) + -3(-2-(-5)) = -57
    sqrt(109^2 + -9^2 + -57^2) = 123.332…….
    // Convert Unit vector
    109/123.332882882 = 0.88378701164
    -9/123.332882882 = -0.07297323949
    -57/123.332882882 = -0.46216385012
    Unit vector = (0.88378701164, -0.07297323949, -0.46216385012)
    Disagree with this
    Yeha this is wrong unnormalized should be (43, 45, -24) ✓✔✓✔

Where do you live

Agree with (43,45,-24) for unnormalized

sqrt(43^2 + 45^2 + (-24)^2) = 66.70832032...

Convert unit vector:

43/66.70832032 =0.645

45/66.70832032 =0.675

-24/66.70832032 =-0.360

Unit vector = (0.645,0.675,-0.360)✔✔ ✔ ✔✓

c) Phi = cos^(-1) * (v dot w) //lec06 p4 - this needs to be the unit vectors ✔

also, I would imagine it should be negative v, and w, or (B-A) and (B-C)

Surely its (C-B) and (A-B) as the vector should have its origin at B and point in the other direction

It doesn't matter either way (see the images below where I calculated the dot product of (B-C) and (B-A) and alternatively (C-B) and (A-B). Both equal 50), so it's simpler to reuse the vectors that were already calculated. It definitely needs to be the unit vectors either way (aka normalized) (see lec06 slide 4).

Phi = cos^-1 (normalized(v) dot normalized(w))

Show Steps

$(4-7 \quad -5+2 \quad -3-8)(4-1 \quad -5-0 \quad -3-1)=50$

Steps

$(4-7 \quad -5+2 \quad -3-8)(4-1 \quad -5-0 \quad -3-1)$

Computing dot product of two vectors: $(x_1, ..., x_n) \cdot (y, ..., y_n) = \sum_{i=1}^{n} x_i y_i$

$= (4-7)(4-1) + (-5+2)(-5-0) + (-3-8)(-3-1)$

Simplify $(4-7)(4-1) + (-5+2)(-5-0) + (-3-8)(-3-1)$: 50   Hide Steps

$(4-7)(4-1) + (-5+2)(-5-0) + (-3-8)(-3-1)$

$(4-7)(4-1) = -9$ Show Steps

$= -9 + (2-5)(-5-0) + (-3-8)(-3-1)$

$(-5+2)(-5-0) = 15$ Show Steps

$(-3-8)(-3-1) = 44$ Show Steps

$= -9 + 15 + 44$

Add/Subtract the numbers: $-9 + 15 + 44 = 50$

$= 50$

$= 50$

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

Show Steps

$$(7-4 \quad -2+5 \quad 8+3)(1-4 \quad 0+5 \quad 1+3) = 50$$

Steps

$$(7-4 \quad -2+5 \quad 8+3)(1-4 \quad 0+5 \quad 1+3)$$

Computing dot product of two vectors: $(x_1, \ldots, x_n) \cdot (y, \ldots, y_n) = \sum_{i=1}^{n} x_i y_i$

$$= (7-4)(1-4) + (-2+5)(0+5) + (8+3)(1+3)$$

Simplify $(7-4)(1-4) + (-2+5)(0+5) + (8+3)(1+3)$:   50   Hide Steps ⊖

$$(7-4)(1-4) + (-2+5)(0+5) + (8+3)(1+3)$$

$(7-4)(1-4) = -9$ Show Steps ⊕

$$= -9 + (5-2)(0+5) + (8+3)(1+3)$$

$(-2+5)(0+5) = 15$ Show Steps ⊕

$(8+3)(1+3) = 44$ Show Steps ⊕

$$= -9 + 15 + 44$$

Add/Subtract the numbers: $-9 + 15 + 44 = 50$

$$= 50$$

$$= 50$$

$$\cos\phi = \left(\frac{v_1}{|v_1|}\right) \bullet \left(\frac{v_2}{|v_2|}\right) = \text{The dot product of the corresponding}$$ **unit** vectors

Theta = 53.2639

Q4:

Question 4. [10 marks for the whole question] *Matrices*

(a) [4 Marks] Describe the transformation represented by the following matrix product. Please give the parameters of each transformation (scale factors, angle, axis of rotation etc.) and also the order in which the transformations are applied to the point (x, y, z). You are not required to multiply the matrices.

$$\begin{bmatrix} 1 & 0 & 0 & 0.5 \\ 0 & 1 & 0 & -0.2 \\ 0 & 0 & 1 & 0.1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.707 & 0.707 & 0 & 0 \\ -0.707 & 0.707 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

(b) [4 Marks] An object is rotated about a pivot point (5, 3, 2) by an angle 30° with the axis of rotation parallel to the y-axis. Write the matrix expression (a product of matrices similar to that given in the previous question) for the transformation. You are not required to multiply the matrices. The matrices should contain only numerical values.

(c) [2 Marks] In the following equation, a vector is transformed by a matrix, but the result of the transformation is same as the original vector. Briefly explain why the vector is not changed by the transformation.

$$\begin{bmatrix} 1 & 0 & 0 & 10 \\ 0 & 1 & 0 & 20 \\ 0 & 0 & 1 & 30 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ 0 \end{bmatrix}$$

a) Translate point by: (x + 0.5), (y - 0.2), (z + 0.1)
Scale point by (x * 0.5), (y * 1), (z * 0.5)
Rotate by Z-axis by 45 degrees clockwise as cos(45) = cosc(-45) = 0.707 but sin(-45) = -0.707 ✔✔✓

(Also sin(45) = 0.707)
sin(-45) = -0.707 so yeah definitely clockwise since openGl is weird like that.

Order of transformations is: Translate, Scale, Rotate
The right column is the input point,hence the order is:
Rotate->Scale->Translate✔✔✔✔✔✓✔

b)

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

✓

1	0	0	-5		0.87	0	0.5	0		1	0	0	5
0	1	0	-3		0	1	0	0		0	1	0	3
0	0	1	-2		-0.5	0	0.87	0		0	0	1	2
0	0	0	1		0	0	0	1		0	0	0	1

✔ Believe you have the left and right matrices in the wrong order. Here you are doing the translation x+5, y+3, z+2 first rather than last. Yeah this is the wrong way around.

|1 0 0 5| |0.87 0 0.5   0| |1 0 0 -5| |x|

|0 1 0 3| |0   1 0   0| |0 1 0 -3| |y|

|0 0 1 2| |-0.5 0 0.87 0| |0 0 1 -2| |z|

|0 0 0 1| |0   0 0   1| |0 0 0 1| |1|
✔ ✔✔
Note: don't get angles mixed up, it's definitely +30
here ✓

c) Vectors are unaffected by translation because they are defined by their magnitude and direction, moving them does not affect this ✔✔

Q5:

Question 5. **[10 marks for the whole question]** *Texture Mapping*

Four quadrilaterals are arranged as shown in Fig. 3a, to form one large square.

(a) [5 Marks] Write the texture coordinates for the vertices *A..I* that give a mapping shown in Fig. 3b, where only the upper left region of the texture $(0 \le s \le 0.5, \quad 0.5 \le t \le 1)$ is used.

(b) [5 Marks] Write the texture coordinates for the vertices *A..I* that give a mapping shown in Fig. 3c. Assume that the texture wrap mode is set to GL_REPEAT along both directions *s* and *t*.

Texture:

Figure 3a

Figure 3b

Figure 3c

a) A = (0, 0.5)
    B = (0.2, 0.5)
    C = (0.5, 0.5)
    D = (0, 0.9)
    E = (0.2, 0.9)
    F = (0.5, 0.9)
    G = (0, 1)
    H = (0.2, 1)
    I = (0.5, 1)✔✔✔✔✔✓✔✔

Figure 3a

b) A = (0 , 0)
B = (0.8 , 0)
C = (0, 2)
D = (0, 1.8)
E = (0.8,
F = (0.5, 1.8)
G = (0, 2)
H = (0.8, 2)
I = (2, 2)
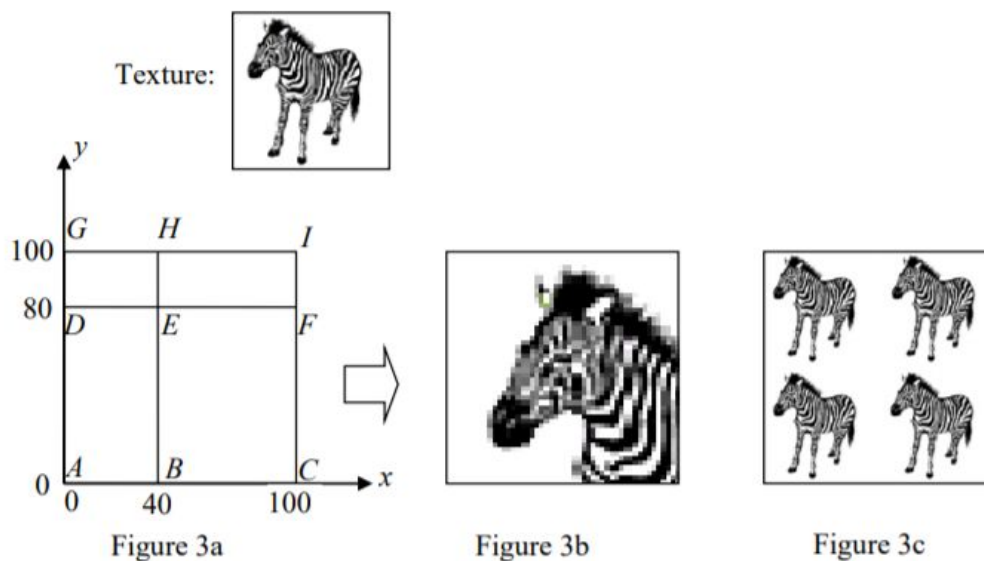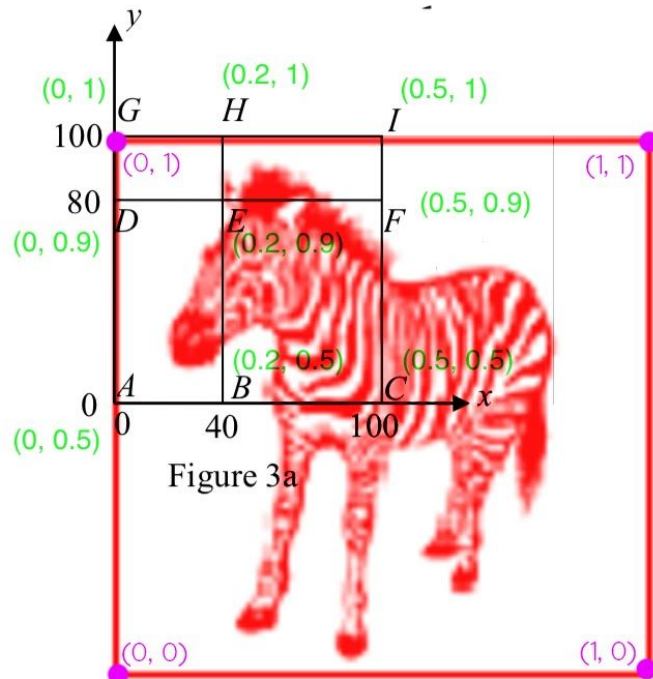This question is gay like your mum F My mum can beat up your mum

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

I think this is it
A = (0 , 0)
B = (0.8 , 0)
C = (2, 0)
D = (0, 1.6)
E = (0.8, 1.6)
F = (2, 1.6)
G = (0, 2)
H = (0.8, 2)
I = (2, 2) ✔✔✔✔✔✔✓✔✔✔

Q6:

Question 6. **[10 marks for the whole question]** *Projections*

A view frustum is specified using the following statement:

```
gluPerspective(60.0, 1, 20.0, 100.0);
```

(a) [4 Marks] Give a brief (one sentence) description for each parameter of the above function.

(b) [6 Marks] Use the equation $h = 2N \tan(\text{fov}/2)$ to compute the height h of the near plane. Then compute the values of the parameters of the function glFrustum(L, R, B, T, N, F) that represents the same view frustum generated by the gluPerspective function given above.

a) gluPerspective(60.0, 1, 20.0, 100.0);
   60 is the fov of the camera/eye
   1 is the aspect ratio
   20 is the distance from the camera/eye to the near plane
   100 is the distance from the camera/eye to the far plane

   Should we specify that it is the vertical field of view?

b) H = 2N * tan(fov/2)
       = 2*(20) * tan(60/2)
       = 23.0

   W = a * h
      = 1 * 23.0

   L = -w/2

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

= -23 / 2

= -11.5

R = w/2

= 23/ 2

= 11.5

B = -h / 2

=- 23/2

= -11.5

T = h /2

= 23/ 2

= 11.5

N = 20

F = 100

Thus, glFrustum(-11.5, 11.5, 11.5, -11.5, 20, 100)✔✔✔✔✓✔✔✔

Q7:

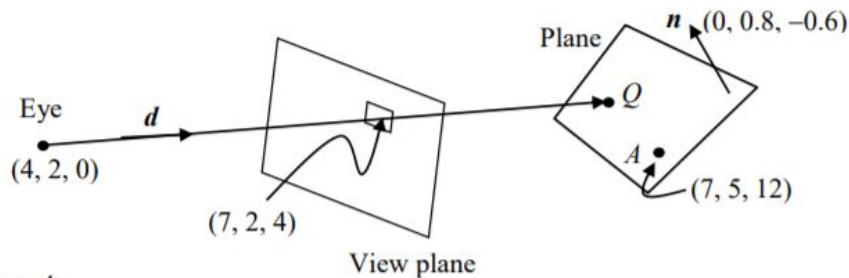**Question 7.** **[10 marks for the whole question]** *Ray Tracing*

Figure. 4

A primary ray is traced from the eye position (4, 2, 0) through the point (7, 2, 4), as shown in Fig. 4. The ray hits a plane that contains the point $A = (7, 5, 12)$. The plane's unit normal direction is (0, 0.8, −0.6).

(a) [2 Marks] What are the values of p_0 and d in the equation of the ray : $p = p_0 + td$?

(b) [5 Marks] Compute the value of the ray parameter t at the point Q where the ray meets the plane. What does this value of t represent?

(c) [3 Marks] Compute the coordinates of the point of intersection Q.

a) p = p0 + td
   P0 = (4, 2, 0)

   D = (7, 2, 4) - (4, 2, 0)
   = (3, 0, 4)

   Then normalise by dividing each value but its sqrt thing
   Thus d = (⅗, 0, ⅘)
   ✓✔

b) T = (Q-p0) * n) / (d * n)
   = ((7, 5, 12) - (4, 2, 0) * (0, 0.8, -0.6)) / (⅗, 0, ⅘) * (0, 0.8, -0.6)
   = 10 ✓✔✔

   T represents the distance from the eye position to the intersection point Q.✔✔✔

   ^^ I don't think t is the distance from the eye position to Q. t is a scalar that modifies the length of the direction vector d. In this case the distance from eye position to intersection point Q is actually 10, not 2. What do you mean? **KISS**

Does anyone know why A is used for Q?   What do you mean? A is any point on the plane that's not Q. You can use the equation on pg 29 of Lec8 to find t.

   c) P = P0 + td = (4,2,0) + 10*(0.6,0,0.8) = (10,2,8)✔ ✓ ✔ ✔

Q8:

Question 8. [10 marks for the whole question] *Bezier Curves.*

(a) [7 Marks] Given three control points $P_0 = (2, 0, 2)$, $P_1 = (1, -4, 2)$, $P_2 = (0, -2, 1)$, write the parametric equations for $x(t)$, $y(t)$, $z(t)$ for points on the Bezier curve.

(b) [3 Marks] Show that all points on the Bezier curve in the previous quesion satisfy the equation $x^2 = y + 2z$.

a) x(t) = 2(1 - t) - how do u get this I dont understand how to get to these answers, Expand and simplify

y(t) = 2t(3t - 4)

z(t) = 2 - t^2

$$Q = (1-u)^2\, P_0 + 2(1-u)u\, P_1 + u^2\, P_2$$

This eqn gives u the vector (x(t), y(t), z(t))

Can you show more working please, still unsureb how to solve this one.

The same question as you ,please give more detail

Idk if this is right at all but i tried using the formula above

Qx = (1-t)^2 P0x + 2t(1-t)P1x + t^2 P2x

       = 2(1-t)^2 + 2t(1-t) = 2-2t

Qy = (1-t)^2 P0y + 2t(1-t)P1y + t^2 P2y

        = 2t(1-t)(-4) + t^2(-2) = 6 t^2 - 8 t

Qz = (1-t)^2 P0z + 2t(1-t)P1z + t^2 P2z

        = 2(1-t)^2 + 2t(1-t)2 + t^2 = 2 - t^2

x(t) = 2-2t

y(t) = 6t^2 - 8t

z(t) = 2-t^2

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

✓✔✔

b) Sub in values of x, y ,z into the equation $x^2 = y + 2z$

4t^2 -8t + 4 = 4t^2 - 8t + 4) where did this come from? I'm confused, you rearrange x^2=y+2z into x^2-y-2z=0

0 = 0

Thus it satisfies the equation ✓ ✔✔✔

Q9:

Question 9. **[10 marks for the whole question]** *OpenGL-4 Shaders.*

(a) [2 Marks] Briefly describe two important computations that are commonly performed inside a vertex shader.

(b) [8 Marks] Give a brief description for each of the following statements taken from the code of a vertex shader.

      (i) `layout (location = 0) in vec4 position;`

      (ii) `uniform mat4 mvpMatrix;`

      (iii) `gl_Position = mvpMatrix * position;`

      (iv) `out vec4 theColour = ambient + diffuse + specular;`

a) Outputs vertices from world coordinates to clip coordinates
Performs lighting calculations on current vertex ✔

- Tessellation Control Shader: modify patch vertices, if needed. Specify tessellation levels

- Primitive Generator: the primitive generator can output only triangles. These triangles form a tessellation of either a square or a triangle domain. The vertices of every triangle in the tessellation will have normalised coordinates

- Tessellation Evaluation shader: The evaluation shader converts the primitive vertices(u, v) to 3D points using use-defined functions, and outputs them in clip coordinate space.

b)

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

    i)   Creates a variable position at location 0 bounded to the VBO
    ii)   Declares variable mvpMatrix of type mat4 and links to OpenGl and apps
    iii)   Define variable gl_Position to be equal to the model view matrix * world space position
    iv)   Declare out variable of type vec4 with name theColor to be the sum of ambient, diffuse and specular

Q10:

Question 10. [10 marks for the whole question] *Tessellation Shaders.*

(a) [5 Marks] Outline two uses of a tessellation control shader. Briefly describe one application where a tessellation control shader could be effectively used.

Figure. 5
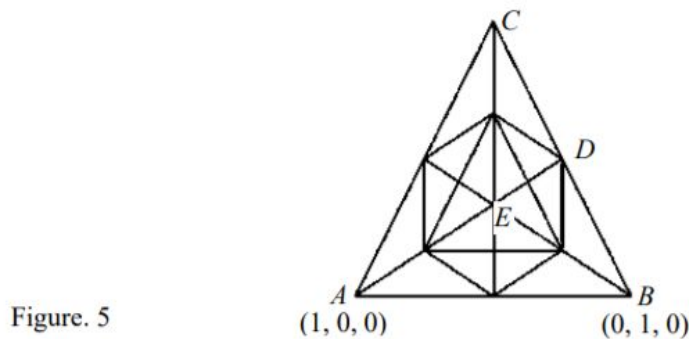
(b) [5 Marks] A triangle domain is tessellated as shown in Fig. 5. The tessellation coordinates of two points A, B on the domain are also given.

(i). Write the tessellation coordinates of the points C, D and E.

(ii). Write the inner and outer tessellation levels used in generating the figure.

a) The tessellation control shader is commonly used to set the inner and outer tessellation levels. It can also create new or remove existing patch vertices. All patch vertices are available inside the shader in an array. (Topic 10 slide 14)
Effectively used in terrain rendering✔

- They can be used to adjust the tessellation levels based on the distance of the current patch from the camera.

- Can modify patch vertices, if needed. They can also Specify tessellation levels

- The tessellation control shader is commonly used to set the inner and outer tessellation levels

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓if you think you would fail an open book exam ✓ (Seng201 flashbacks)

b)
  i) C = (0, 0, 1)
     D = (0, 0.5, 0.5)
     E = (0.33, 0.33, 0.33)✔
  ii) 2220, 40✔
  How do we enter an ans for a question like this? Is this form alright?
  How do you figure out the inner level? Its how many edges you need to traverse to get from a corner to an edge.

2017

Question 1 - Transformations

Question 1. [10 marks for the whole question] *Transformations*

Assume that you are given three functions: (i) `drawCylinder()` that draws a cylinder centred at the origin with axis along the y-axis, height 1 unit and radius 1 unit (ii) `drawCone()` that draws a cone with the centre of the base at the origin and axis along z-axis, radius 1 unit and height 1 unit, and (iii) `drawFin()` that draws a fin shaped object near the origin (Fig. 1).

Fig. 1.
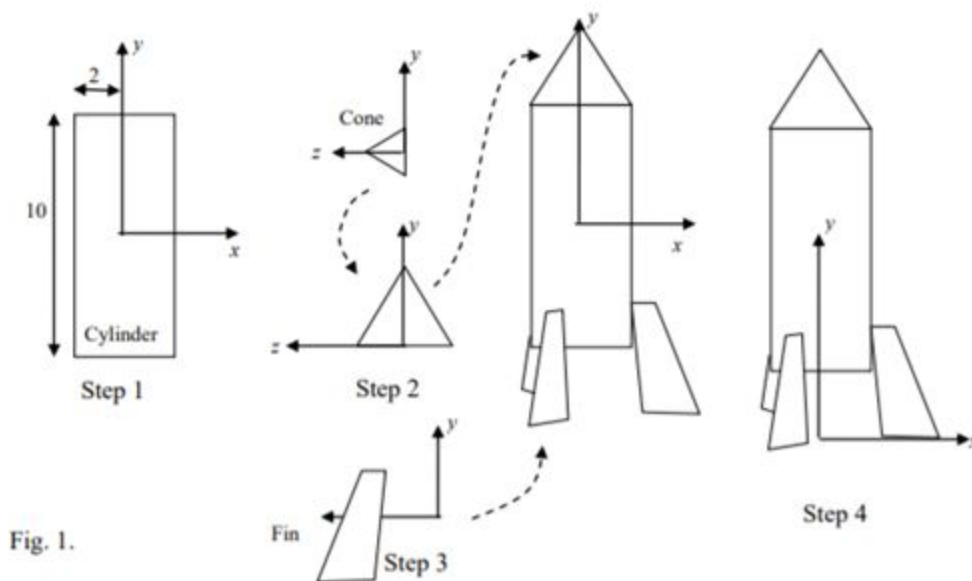
You are given the following instructions for creating the model of a rocket:

Step 1: Create a cylinder using `drawCylinder()` and scale it along both x and z axes by a factor 2, and along the y axis by a factor 10.

Step 2: Create a cone using `drawCone()` and rotate it to make its axis vertical as shown in the figure above. You need to find the angle and axis of rotation. Scale the rotated cone by a factor of 2 along all three axes, and then translate it to position (0, 5, 0).

Step 3: Create three fins calling the function `drawFin()` thrice. Rotate the second fin by 120 degs about the y-axis, and the third fin by 240 degs about y-axis. Translate the three fins to (0, −5, 0).

Step 4: Steps 1-3 above create the rocket model. Translate the whole model by (0, 8, 0) .

Using the given functions, OpenGL transformation functions and glPushMatrix-glPopMatrix blocks, write a code segment for defining only the transformations as outlined above, so that a model as depicted in the Fig. 1 (Step 4) is generated. You need not write any other OpenGL code or functions as part of the answer.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

(From 2019 Study doc)

```
glPushMatrix();
        glTranslatef(0,8,0);        it should be before glPushMatrix(), shouldn't it ? agree
        glPushMatrix();
                glScalef(2,10,2);
                drawCylinder();
        glPopMatrix();

        glPushMatrix();
                glTranslatef(0,5,0);
                glScalef(2,2,2);
                glRotatef(-90,1,0,0);   should be glRotatef(-90, 0, 0, 1) ← nope, rotation axes is X
                drawCone();
        glPopMatrix();

        glPushMatrix();
                glTranslatef(0,-5,0);
                drawFin();
                glRotatef(120,0,1,0);
                drawFin();
                glRotatef(120,0,1,0);
                drawFin();
        glPopMatrix();
glPopMatrix();
```

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

Question 2 - Vectors and Matrices

Question 2. [15 marks for the whole question] *Vectors and Matrices*

(a) [4 Marks] Consider the following code segment:

```
glMatrixMode(GL_MODELVIEW);
glLoadIdentity();
glTranslated(5.0, 3.0, 1.0);
glScaled(0.5, -1.0, 0.5);
glRotated(30, 0, 0, 1);
glBegin(GL_POINTS);
     glVertex3d(6., 4., 3.);          //Point P
glEnd();
```

Write the transformation of the point P as a matrix expression, where each transformation is represented by a 4x4 matrix containing only numerical values. You are not required to multiply the matrices. Rotation matrices are given on Page 2.

(b) Consider the triangle formed by three points A, B, C as shown below (Fig. 2):
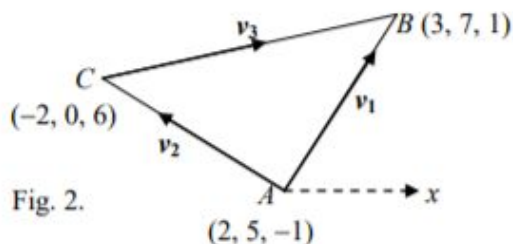
Fig. 2.

(i) [2 Marks] Compute the vectors v_1 and v_2 shown in the figure above.

(ii) [2 Marks] Compute the unit vector along the direction of v_1

(iii) [2 Marks] Show that the vector v_1 is perpendicular to v_2 .

(iv) [3 Marks] Using the vector cross-product formula (see Page 2), compute the normal vector of the triangle. You need not convert this vector to a unit vector.

(v) [2 Marks] Compute the length of the projection of the vector v_3 on the x-axis.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.
put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

(a)
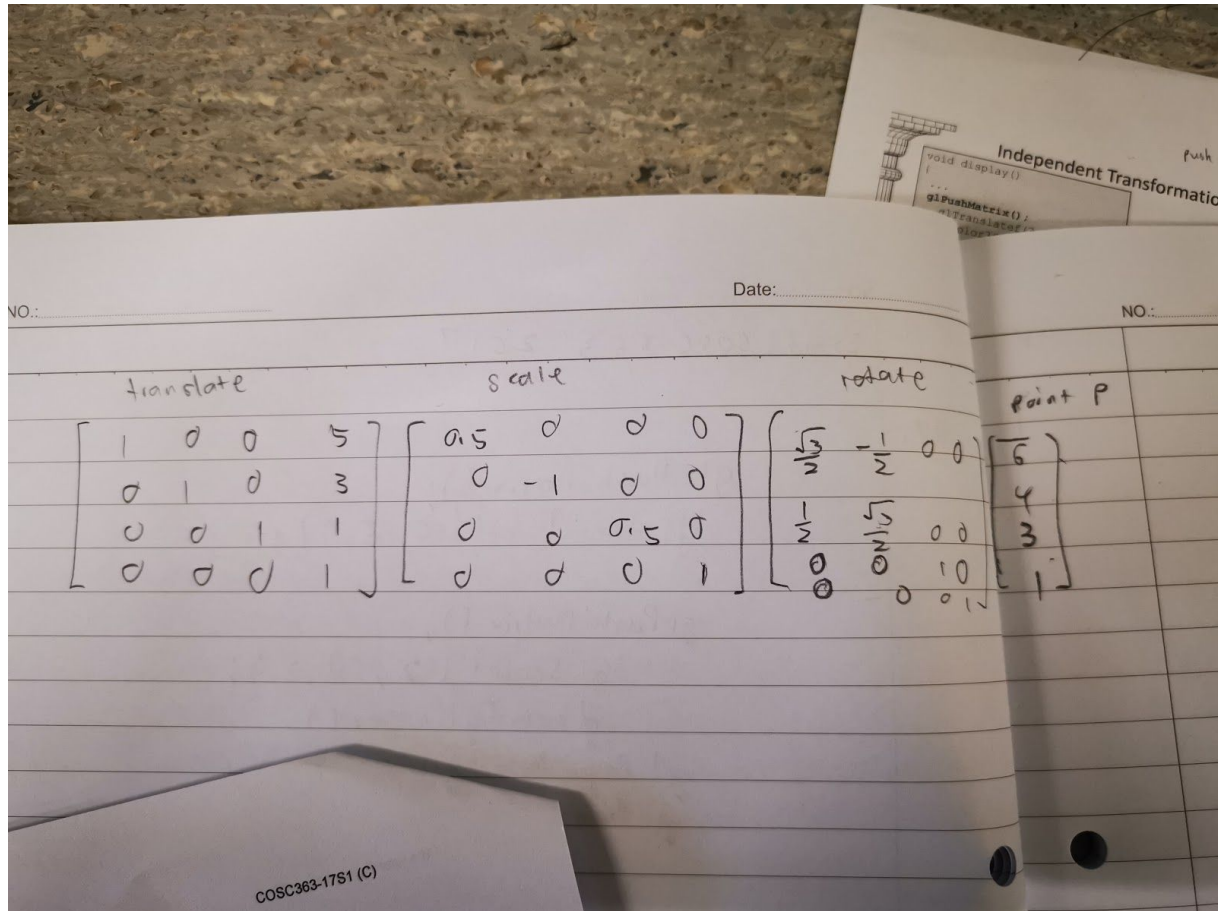
Is it not like this? Is this correct? agree agree ✔✔
Does order Matter? Yes ✔ ✓

(b)

(i) [2 Marks] Compute the vectors v1 and v2 shown in the figure above.

V1 = B - A = (1, 2, 2)   agree✔
V2 = C - A = (-4, -5, 7) agree ✓ ✔

(ii) [2 Marks] Compute the unit vector along the direction of v1

V1 = (1, 2, 2)
Magnitude of V1 = sqrt(1^2 + 2^2 + 2^2) = 3
Unit Vector of V1 = (⅓, ⅔, ⅔) ✓ ✔

(iii) [2 Marks] Show that the vector v1 is perpendicular to v2 .

1*-4 + 2 * -5 + 2*7 = -4 +-10 +14 = 0 ✓ ✔

IT IS PERPENDICULAR I'm pretty sure this is wrong. You have to normalise the vectors (i.e. they have to be unit vectors) before you take their dot product. Then, if the dot product is 0, they are perpendicular.

Nah you dont need to normalize them if you are just checking if the dot product is zero. Normalizing just changes the size of the output, but anything times 0 is still 0...

You dont need to do that ^

(iv) [3 Marks] Using the vector cross-product formula (see Page 2), compute the normal vector of the triangle. You need not convert this vector to a unit vector.

I got (0, -23, 3) ? I got (12, -5, -1)?

v1 x v2 = ((2*7) - (-5*2), (2*-4) - (1*7), (1*-5) - (-4*2)) = (24, -15, 3)
+1 +1 +1 ✔✔✓✔

(v) [2 Marks] Compute the length of the projection of the vector v3 on the x-axis.

V3 = B - C = (5, 7, -5)
The length of the projection of V3 on the x-axis is 5
How to compute this one?n = v.n?
I think it's the magnitude? Which would be 9.95? Square root of (5^2 + 7^2 + 5^2) ?
Should be square root of(5^2 + 7^2 + 5^2)

This is on the formula sheet. Length of projection of a vector v on a unit vector n = v •n
V = (5, 7, -5)
N = (1, 0, 0) because we are projecting onto the x-axis
V.n = 5 agree ✔✔✓
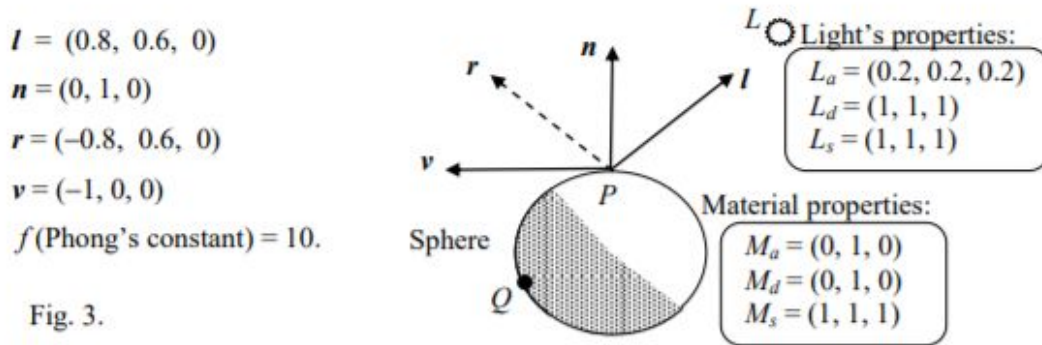
Question 3 - Illumination

Question 3. [10 marks for the whole question] *Illumination*

The following figure (Fig. 3) gives the components of unit vectors used for lighting calculations at a point P on a sphere, and also the ambient, diffuse, specular properties of the light and the sphere material (l = light source vector, n = normal vector, v = view vector, r = reflection vector).

$l = (0.8, 0.6, 0)$

$n = (0, 1, 0)$

$r = (-0.8, 0.6, 0)$

$v = (-1, 0, 0)$

f (Phong's constant) = 10.

Fig. 3.

Light's properties:

$L_a = (0.2, 0.2, 0.2)$
$L_d = (1, 1, 1)$
$L_s = (1, 1, 1)$

Material properties:

$M_a = (0, 1, 0)$
$M_d = (0, 1, 0)$
$M_s = (1, 1, 1)$

(a) [2 Marks] Assume that L is the only light source in the scene. A point Q lies in the shadow region of the sphere. Write the condition satisfied by the light source vector and the normal vector at this point. Write the colour components of net reflection from this point.

(b) [3 Marks] Write the mathematical expression for the colour of diffuse reflection at P, and compute the numerical values of its components.

(c) [3 Marks] Write the mathematical expression for the colour of specular reflection (including the shininess term) at P along the view direction v, and compute the numerical value of the colour components.

(d) [2 Mark] Write the expression for the half-way vector at P, and compute its components. You need not normalize this vector.

(a) [2 Marks] Assume that L is the only light source in the scene. .A point Q lies in the shadow region of the sphere. Write the condition satisfied by the light source vector and the normal vector at this point. Write the colour components of net reflection from this point.

l·n = (0.8,0.6,0)·(0,1,0) = 0.6 > 0
This should be of point Q rather than P
Seconded. Shouldn't the condition be l·n<0 ? since the dot product is less than 0 when the angle is greater than 90.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

L·n >= 90 // If the angle between l and n is greater than or equal to 90 it is in shadow. However, the dot product doesn't calculate this angle. The answer in blue below is correct.
La*Ma = (0, 0.2, 0) so colour in shadow is dark green

The point Q is in shadow. Therefore l.n <= 0 where n is the normal vector at point Q.
The colour components are LaMa = (0, 0.2, 0) i.e. only ambient light. Agree a+gree ✔✔ ✓

(b) [3 Marks] Write the mathematical expression for the colour of diffuse reflection at P, and compute the numerical values of its components.

Id = Ld Md max(l·n, 0) = (0, 0.6, 0) ? YES agree ✔ agree✔✔ ✓
(page 5 of lecture 7)

(c) [3 Marks] Write the mathematical expression for the colour of specular reflection (including the shininess term) at P along the view direction v, and compute the numerical value of the colour components.

Ls*Ms*(max(r·v, 0))^f = (1, 1, 1) * (1, 1, 1) * (0.8)^10 = (0.107, 0.107, 0.107) is this correct? (AGREE)✔ ✔ ✓
When do we use r·v or h·n? I am also wondering.
^ At this stage we have not calculated the halfway vector h. Use r.v instead.
h.n is an approximation of r.v, therefore r.v should be used when possible.

(d) [2 Mark] Write the expression for the half-way vector at P, and compute its components. You need not normalize this vector.

h = l + v = (0.8, 0.6, 0) + (-1, 0, 0) = (-0.2, 0.6, 0) This answer is brought to you by team Proffat //agree +1 ✔ ✓

-

Question 4 - Texture Mapping

Question 4. [10 marks for the whole question] *Texture Mapping*
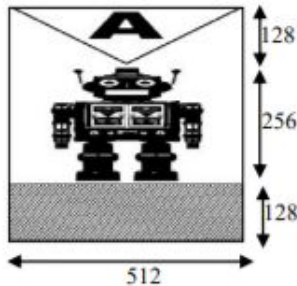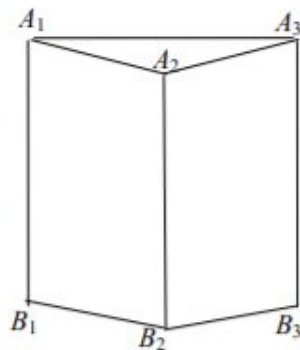

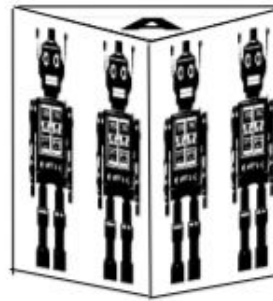Fig. 4 (a)   Fig. 4(b)   Fig. 4(c)

A single texture image of size 512x512 pixels is shown in Fig. 4(a). The top part of the image has a triangle with the letter 'A' inside it, the middle section has a robot image, and the bottom part has a striped pattern. A polygonal model is shown in Fig. 4(b). It consists of a quad strip $\{A_1, B_1, A_2, B_2, A_3, B_3\}$, and a triangle $\{A_1, A_2, A_3\}$. The texture is required to be mapped onto the model as shown in Fig. 4(c) with each of the two sides of the quadstrip containing two robot images and the triangle on top containing the 'A' image. The shaded/striped part of the texture is not used.

(a). [7 Marks] Write the texture coordinates for the six vertices of the quadstip in the following format in your answer booklet:

$$A_1 = (\quad , \quad)$$
$$B_1 = (\quad , \quad) \text{ .. etc.}$$

(b) [3 Marks] Write the texture coordinates for the vertices of the triangle $\{A_1, A_2, A_3\}$.

(a). [7 Marks] Write the texture coordinates for the six vertices of the quadstip in the following format in your answer booklet:
A1 = (,)
B1 = (,) .. etc.

A1 = (0, 0.75)   can y-value be 1 here? No
A2 = (2, 0.75)
A3 = (4, 0.75)

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

B1 = (0, 0.25)
B2 = (2, 0.25)
B3 = (4, 0.25)
+1 +1 +1✔✔✓✔

(b) [3 Marks] Write the texture coordinates for the vertices of the triangle {A1, A2, A3}.

A1 = (0, 1)
A2 = (0.5, 0.75)
A3 = (1, 1)
+1 +1 +1✔✔✓✔

Question 5. View Transformation

Question 5. [10 marks for the whole question] *View Transformation*

A view transformation matrix transforms points from world coordinates to eye coordinates. Consider the following view transformation matrix:

$$\begin{bmatrix} 0.8 & 0 & 0.6 & -11 \\ 0 & 1 & 0 & -7 \\ -0.6 & 0 & 0.8 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(a) [3 Marks] Compute the eye coordinates of the point P (10, 7, 5) using the above matrix. From your answer, what can you conclude about the point P?

(b) [4 Marks] Consider a vector v in homogeneous coordinates (6, 0, −8, 0). Transform the vector using the above view transformation matrix. From your answer, what can you conclude about the vector v?

(c) [3 Marks] A point Q has world coordinates (−5, 3, 10). Using its eye coordinates, determine if this point is behind or in front of the camera .

(a) [3 Marks] Compute the eye coordinates of the point P (10, 7, 5) using the above matrix. From your answer, what can you conclude about the point P ?

(0, 0, 0, 1) agree✔✔✓
Position of the camera agree (in eye coordinates)

HOW THE HECK DO YOU DO THIS - PLS HALP

--multiplying the matrix by the vector(10, 7, 5, 1), like MP

Rotate around y-axis by about 37 degree than translate by(-11, -7, 2)?

      -- don't think you can simply break the matrix into one translate matrix and one rotate matrix agree

Topic 7 slide 21 is useful for this

Lec06 p12

(b) [4 Marks] Consider a vector v in homogeneous coordinates (6, 0, -8, 0). Transform the vector using the above view transformation matrix. From your answer, what can you conclude about the vector v?

View transformation matrix * v = (0, 0, -10, 0) agree ✔the matrix*v is the same as MP?

This is the negative z-axis i.e. the direction the camera is facing (in eye coordinates).✔✔

(See "view axis" diagram in Topic 7 slide 21)✓

(c) [3 Marks] A point Q has world coordinates (-5, 3, 10). Using its eye coordinates, determin , 13, 1) how did you do this? Topic 6, slide 12

The point Q is behind the camera. We know this because the camera faces in the negative z-direction while the point Q has a positive z-value. +1 +1✔ ✓

Question 6. Projections

Question 6. [10 marks for the whole question] *Projections*

A view frustum is specified using the following statement:

```
glFrustum(-40.0,  40., -30., 30.,  50.0, 100.0);
```

(a) [6 Marks] Draw a sketch of the view frustum, clearly showing what each of the six parameters of the above function represent. The figure should also show the camera position, camera coordinate axes directions, and the view axis direction.

(b) [4 Marks] Using the parameters of the function, compute the aspect ratio of the near plane, and the field of view along the y-direction.

(a) [6 Marks] Draw a sketch of the view frustum, clearly showing what each of the six parameters of the above function represent. The figure should also show the camera position, camera coordinate axes directions, and the view axis direction.

Would this just be a 3D square or two drawings; 1st being like -40| -30_ |40 --30 (a box with those things). Then a second picture with the line close to the projection being 50 and then another far away being 100. At least my kinda thoughts no idea really tho.
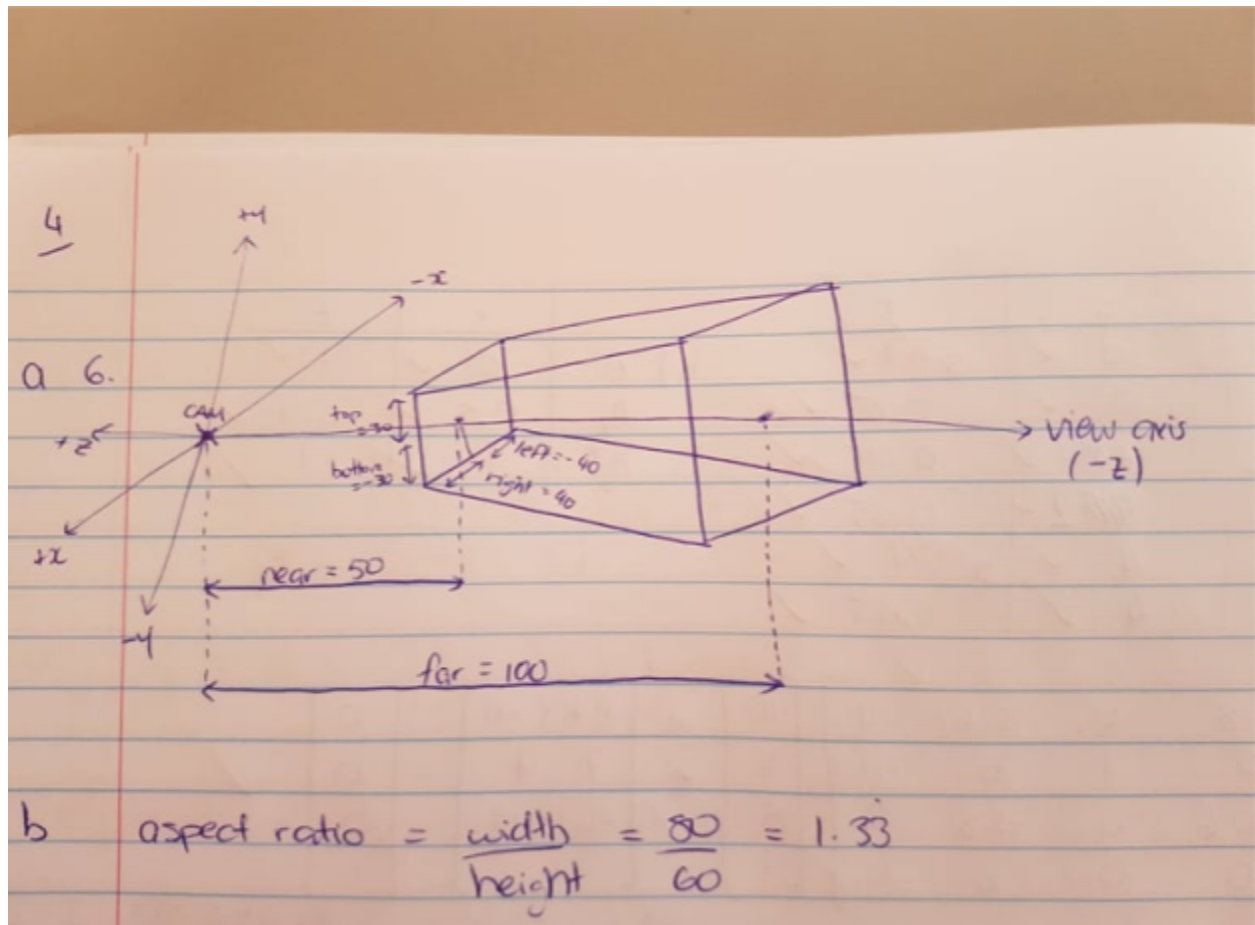
Someone please confirm above image.✔ ✓

(b) [4 Marks] Using the parameters of the function, compute the aspect ratio of the near plane, and the field of view along the y-direction.

Aspect: 80/60 = 4/3 or 1.33 Note: width = 40 * 2; height = 30 * 2….
FOV: theta = 2 * arctan2(h / 2N) = 2 * arctan2(60/100) = 61.93 deg

Question 7. Surfaces of revolution

Question 7. [8 marks for the whole question] *Surfaces of revolution*

Consider the following code segment which generates a surface of revolution by transforming points (vx[i], vy[i], vz[i]), i = 0..N-1, of a base polygonal curve.

```
for(int j = 0; j < 36; j++)    //36 slices
{
    for(int i = 0; i < N; i++)    //Vertex transformations
    {
        wx[i] = cos(0.1745)*vx[i] + sin(0.1745)*vz[i];
        wy[i] = vy[i];
        wz[i] = -sin(0.1745)*vx[i] + cos(0.1745)*vz[i];
    }

    glBegin(GL_QUAD_STRIP);           //Create quads
    for(int i = 0; i < N; i++)
    {
        glTexCoord2f(s1,  t1);
        glVertex3f(vx[i], vy[i], vz[i]);
        glTexCoord2f(s2,  t2);
        glVertex3f(wx[i], wy[i], wz[i]);
    }
    glEnd();

    ...    //Update vertices  (code removed)
}
```

(a) [3 Marks] Briefly describe the transformation applied to the vertices, giving the angle and axis of rotation used in this transformation.

(b) [3 Marks] Give the expressions for the texture coordinates s1, t1, s2, t2 in terms of the parameters i, j, such that values of i are mapped along the vertical axis of the texture and j along the horizontal axis.

(c) [2 Marks] How are the vertex coordinates updated after construction of a quad strip? Why is the update operation necessary?

(a) [3 Marks] Briefly describe the transformation applied to the vertices, giving the angle and axis of rotation used in this transformation.

Rotation by 10 degrees about y axis why is it the inverse of sin(0.1745) and not the inverse of cos(0.1745)? How do you know which one gives you the correct angle?

No inverses necessary. 10 degrees = pi/18 radians = 0.1745 radians

(b) [3 Marks] Give the expressions for the texture coordinates s1, t1, s2, t2 in terms of the parameters i, j, such that values of i are mapped along the vertical axis of the texture and j along the horizontal axis.

s1=j/36 ✓ , s2=(j+1)/36 ✓ , t1=i/N, t2=(i+1)/N

i think t2 is also i/N (same)

If t1 and t2 are the same then the texture between them will stretch the same colour from one side to the other. So they must be different, I think?

t1=i/(N-1) t2 = i/(N-1) possibly wrong I think so, for each slice we draw layer by layer from bottom to top, so vy[i] = wy[i] and have same y-coord in texture file (t1.t2 is height in the texture)

(c) [2 Marks] How are the vertex coordinates updated after construction of a quad strip? Why is the update operation necessary?

The last vertices must be connected to the first vertices (not sure on this one)

v[i] = w[i], so that it continues revolving around the axis

That makes sense, agreed. ✓

Question 8. Ray Tracing

Question 8. [12 marks for the whole question] *Ray Tracing*

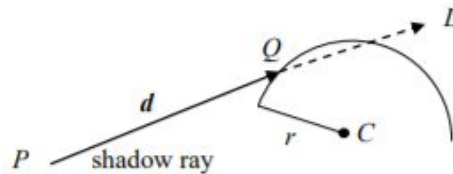A shadow ray is traced from a point P towards the light source L as shown in the following figure (Fig. 5).

Fig. 5.

We define two vectors $s = P - C$, and $l = L - P$.

The points of intersection of the ray $p = P + td$ and a sphere centered at a point C with radius r is given by

$$t = -(s \bullet d) \pm \sqrt{(s \bullet d)^2 - (s \bullet s) + r^2}$$

(a) [2 Marks] Write the conditions for the existence of a valid point of intersection of the ray with the sphere.

(b) [2 Marks] If a valid point of intersection exists and is given by the ray parameter t, what other condition should t satisfy in order that the point P is in shadow?

(c) [2 Marks] What does the term $(s \bullet s)$ in the above equation represent?

(d) [6 Marks] If $P = (0, 2, -10)$, $C = (0, 12, -15)$, $d = (0, 0.6, -0.8)$, $r = 5$, compute the values of the ray intersection parameter t using the formula given above. What can you conclude from the two values of t?

(a) [2 Marks] Write the conditions for the existence of a valid point of intersection of the ray with the sphere.

The equation above must have 2 real roots

Is having one real root ok? If there is only 1 real root, then it means the ray is tangent with the sphere. I remember there was a quiz question, where 1 real root was not considered a valid point of intersection. It's an ambiguous case though.
I believe that 2 roots means if the ray continues it would pass through the object again. ✓
(quiz7,q6)

I think having no root is the problem :) so delta >= 0 to ensure intersection

(b) [2 Marks] If a valid point of intersection exists and is given by the ray parameter t, what other condition should t satisfy in order that the point P is in shadow?

The value of t cannot be larger than the magnitude/length of the light vector (l) and the value of t cannot be negative

(c) [2 Marks] What does the term (s•s) in the above equation represent?

The magnitude squared of the vector s - Mukundan confirmed
Also note that s is the vector p0-C, i.e. from start of ray to center of circle

(d) [6 Marks] If P = (0, 2, 10), C = (0, 12, 15), d = (0, 0.6, 0.8), r = 5, compute the values of the ray intersection parameter t using the formula given above. What can you conclude from the two values of t ?

I got 10 and 10 for my 2 values of t. This means that the ray is tangent to the sphere at point Q
I got -10 and -10, but agreed. The ray is tangent to the sphere. ✓
-(-10) = 10 for both t

Question 9. Bezier Curves.

Question 9. [7 marks for the whole question] *Bezier Curves.*

A two-dimensional cubic Bezier curve through four control points P_0, P_1, P_2, P_3 is given by the equation $P(t) = (1-t)^3 P_0 + 3(1-t)^2 t P_1 + 3(1-t)t^2 P_2 + t^3 P_3$, $0 \le t \le 1$.

(a) [5 Marks] If $P_0 = (0, 0)$, $P_1 = (0, 1)$, $P_2 = (1, 1)$, $P_3 = (1, 0)$, write the parametric equations of the Bezier curve in the form $x(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$; $y(t) = b_0 + b_1 t + b_2 t^2 + b_3 t^3$, where some of the coefficients may be zeros.

(b) [2 Marks] Show that for the curve in the previous question (Q.9a), the point on the curve at $t = 0.5$ is $(0.5, 0.75)$.

(a) [5 Marks] If P0=(0, 0), P1 = (0, 1), P2 = (1, 1), P3 = (1, 0), write the parametric equations of the Bezier curve in the form x(t) = a0 + a1t + a2t 2 + a3t 3 ; y(t) = b0 + b1t + b2t 2 + b3t 3 , where some of the coefficients may be zeros.

X = (1-t)3*0 + 3(1-t)2*t*0+3(1-t)t2*1+t3*1
Y = (1-t)3*0 + 3(1-t)2*t*1+3(1-t)t2*1+t3*0

Those equations are correct, but they haven't been put in the form asked by the question. You still need to rearrange the equations to get: My bad thank you <3.

X = 0 + 0*t + 3t2 - 2t3   <agree Can you put the steps to rearrange it plss ✔
How to calculate ^ ?? agree
You just need to plug each x and y value for P0-P4 into an equation for x(t) and y(t)

Y = 0 + 3t - 3t2 + 0*t3 ✔
Agree. Agree agree

(b) [2 Marks] Show that for the curve in the previous question (Q.9a),

the point on the curve at t = 0.5 is (0.5, 0.75).
X = (1-0.5)3*0 + 3(1-0.5)2*0.5*0+3(1-0.5)*0.52*1+0.5*1= 0.5
Y = (1*0.5)*0 + 3(1-0.5)2*0.5*1+3(1-0.5)*0.52*1+0.53*0 = 0.75
Therefore correct.

Question 10. OpenGL 4 Shaders.

Question 10. [8 marks for the whole question] *OpenGL 4 Shaders.*

(a) [2 Marks] What are the two important computations or operations commonly performed in a vertex shader?

(b) [1 Mark] Give an example of an input (a variable with qualifier **in**) to a vertex shader.

(c) [1 Mark] What does the built-in output variable of a vertex shader, `gl_Position` represent?

(d) [2 Marks] Give an example each of a uniform variable that is commonly used in a vertex shader and a fragment shader.

(e) [2 Marks] Give a brief outline of two operations that can be performed in a tessellation control shader.

(a) [2 Marks] What are the two important computations or operations commonly performed in a **vertex shader?**

Lighting, transformations
Agree, as out var for the fragment shader

VBO and VAO?
VAO is an object pointing to information about vertices, normals etc., while VBOs hold the actual data itself, including vertices and colours. Vertex Shaders use these, but they are not really operations so much as parts necessary to perform operations. Refer to the 2018 exam for a better answer.

(b) [1 Mark] Give an example of an input (a variable with qualifier in) to a vertex shader.

Position or colour (i.e. 'layout (location=0) in vec4 position')

(c) [1 Mark] What does the built-in output variable of a vertex shader, gl_Position represent?

Position of current vertex in clip coordinates

(d) [2 Marks] Give an example each of a uniform variable that is commonly used in a vertex shader and a fragment shader.

Vertex shader: mvpMatrix, Fragment Shader: txSampler, gl_FragColor? gl_fragColor is an out variable, set by fragment shader. Since it is changed during shading, it cannot be uniform. uniform variable are mvpMatrix for vertex shader and txSampler for fragment shader. See slide 51 and 35 of lec10

(e) [2 Marks] Give a brief outline of two operations that can be performed in a tessellation control shader.

(Refer to 2018 Q10a) <- lazy lol
Set the inner and outer tessellation
Add/remove patch vertices

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

2016

Question 1

Question 1. [10 marks for the whole question] *Transformations*

Assume that you are given three functions: (i) `drawCube ()` that draws a cube of side 1 unit, centred at the origin, (ii) `drawSail ()` that draws a single sail/blade of a windmill at the origin, and (iii) `drawCone ()` that draws the base of a windmill (a cone with a predefined radius and height) at the origin, with the cone's axis along the *z*-axis. Figures (1a), (1b), (1c) show the models generated by the functions.

Fig. 1a

Fig. 1b

Fig. 1c

Fig. 1d
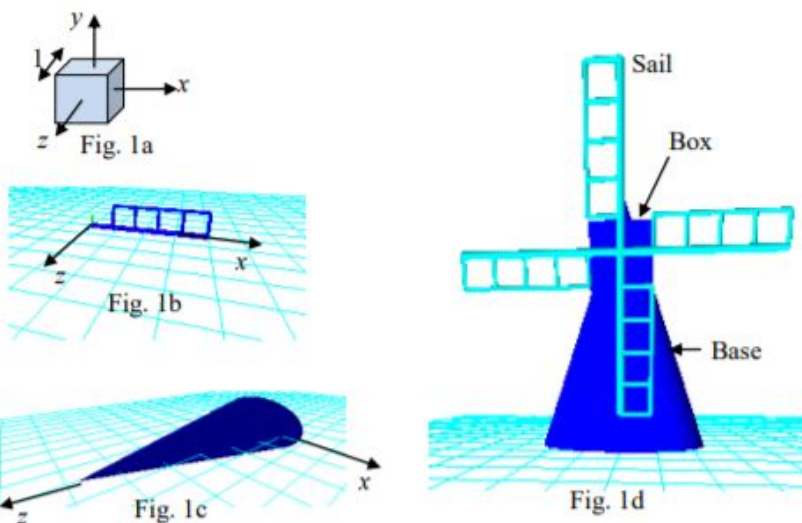
You are also given the following set of instructions for creating the model of a windmill:

Step 1: Create the base of the windmill using `drawCone ()` and rotate it to make the axis vertical. You need to find the angle and axis of rotation.

Step 2: Create a cube using `drawCube ()` at the origin and scale it by 8 times along all three axes.

Step 3: Move the scaled cube along the *y*-axis to position (0, 25, 0).

Step 4: Using a for-loop and the function `drawSail ()`, create four blades at the origin and rotate them 90 degrees apart.

Step 5: Translate the set of four blades to position (0, 25, 5).

Step 6: Steps 1-5 above create the windmill model. Translate the whole windmill to position (100, 0, 50)

Using the given functions, OpenGL transformation functions and glPushMatrix-glPopMatrix blocks, write a code segment for defining only the transformations as outlined above. The transformed model of the windmill at position (100, 0, 50) is shown in Fig. 1d.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

```
glPushMatrix();
        glTranslatef(100, 0, 50);
        glPushMatrix();
                glRotatef(-90, 1, 0, 0);
                drawCone();
        glPopMatrix();

        glPushMatrix();
                glTransaltef(0, 25, 0);
                glScalef(8, 8, 8);
                drawCube();
        glPopMatrix();

        glPushMatrix();
                glTranslatef(0, 25, 5);
                for (int i = 0; i < 4; i++) {
                        glPushMatrix();
                                glRotatef(i * 90, 0, 0, 1);
                                drawSail();
                        glPopMatrix();
                }
        glPopMatrix();
glPopMatrix();
```

✔ ✔

Question 2

Question 2. [12 marks for the whole question] *Illumination Model*

Sam wrote an OpenGL program to render a three-dimensional scene, and included the following lines of code in the `initialize()` function.

```
glColorMaterial(GL_FRONT, GL_AMBIENT_AND_DIFFUSE);
glEnable(GL_COLOR_MATERIAL);
```

(a) [3 Marks] Briefly describe the usefulness of the above statements in a program, and how it affects lighting calculations in a scene.

(b) [3 Marks] Sam noticed that his program did not produce specular highlights on polygonal elements even when lighting was enabled and the light's specular colour was set to white. Give a possible cause for this problem, and suggest a method to generate specular reflections on all polygonal objects. You are not required to write any OpenGL functions in your answer.

(c) [3 Marks] Sam also noticed a problem with the implementation of a spotlight that was directed towards the centre of a floor plane defined using a large quad. The spot light was not visible even though the light's parameters were set correctly. Explain the cause of this problem by describing the way lighting computations are performed in the OpenGL fixed function pipeline, and suggest a method to rectify this problem.

(d) [3 Marks] The figure given below shows the normal vector **n**, light source vector **l**, and the view vector **v** at a vertex *V*. All vectors are unit vectors on the *xy*-plane. Redraw this diagram in your answer booklet, and show the reflection vector **r** and the half-way vector **h**. Also write the values of the angle between **n** and **h**, and the angle between **r** and **v**.

(a) [3 Marks] Briefly describe the usefulness of the above statements in a program, and how it affects lighting calculations in a scene.

> This forces OpenGL to use the colour value defined using glColor3f() for the current material's ambient and diffuse properties.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

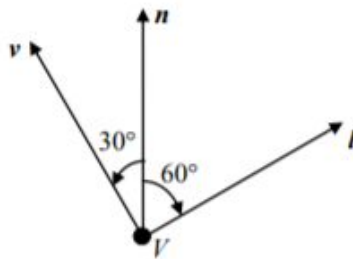put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

(b) [3 Marks] Sam noticed that his program did not produce specular highlights on polygonal elements even when lighting was enabled and the light's specular colour was set to white. Give a possible cause for this problem, and suggest a method to generate specular reflections on all polygonal objects. You are not required to write any OpenGL functions in your answer.

> He propabaly hasn't set the specular material colour for his
> object to white. He can do it by specifying
> glMaterialfv(GL_FRONT, GL_SPECULAR, white); before drawing the object.
> He probably did not specify a shininess term for the object as well.

(c) [3 Marks] Sam also noticed a problem with the implementation of a spotlight that was directed towards the centre of a floor plane defined using a large quad. The spot light was not visible even though the light's parameters were set correctly. Explain the cause of this problem by describing the way lighting computations are performed in the OpenGL fixed function pipeline, and suggest a method to rectify this problem.

> OpenGL calculates lighting only on the vertices of every polygon, thus resulting
> the spotlight lighting up the whole floor pane. To fix this issue, Sam should
> create the floor plane using many small quads.

I think the cutoff angle would mean that the floor plane was not lit by the spotlight right? So it would just be the ambient colour

✓

(d) [3 Marks] The figure given below shows the normal vector n, light source vector l, and the view vector v at a vertex V. All vectors are unit vectors on the xy-plane. Redraw this diagram in your answer booklet, and show the reflection vector r and the half-way vector h. Also write the values of the angle between n and h, and the angle between r and v.

> Angle between n and h is 15 degrees, angle between r and v is 30 degrees.

Question 3

Question 3. **[8 marks for the whole question]** *Texture Mapping.*

The curved surface of a cylinder is required to be texture mapped using a single image as shown in the figure below. One half of the cylinder uses the top portion of the image, while the other half of the cylinder is textured using the bottom half of the image.

Texture
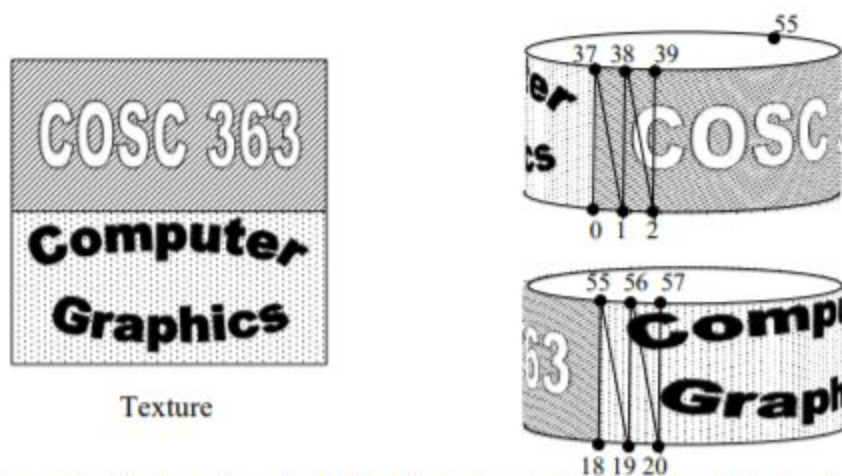
The cylindrical surface is divided into two halves and modelled using two triangle strips. The first few triangles of each triangle strip are shown in the figure.

	Verts on lower edge	Verts on upper edge
Triangle strip-1	0, 1, ...18	37, 38, ... 55
Triangle strip-2	18, 19, ...36	55, 56, ...73

The vertex coordinates are stored in arrays as given below.

```
float vx[74], vy[74], vz[74];
```

Assume that the coordinates of all vertices have been computed. Use glBegin()-glEnd() blocks and two separate for-loops to define the two triangle strips. Assign texture coordinates to vertices to generate a texture mapping as shown in the figure above. Use glVertex3f() function for vertex definition, and glTexCoord2f() for specifying texture coordinates. No other OpenGL functions should be used.

```
glBegin(GL_TRIANGLE_STRIP);
for (int i = 0; i < 19; i++) {
        glTexCoord2f( (float)i/(float)(18), 0.5);
        glVertex3f(vx[i], vy[i], vz[i]);
```

```
        glTexCoord2f( (float)i/(float)(18), 1);
        glVertex3f(vx[i+37], vy[i+37], vz[i+37]);
}
for (int j = 0; j < 18; j++) {                    Should it be < 19 ?
        glTexCoord2f( (float)j/(float)(18), 0);
        glVertex3f(vx[j+18], vy[j+18], vz[j+18]);
        glTexCoord2f( (float)j/(float) (18), 0.5);
        glVertex3f(vx[j+55], vy[j+55], vz[j+55]);
}
glEnd();
        ✓
```

Question 4

Question 4. [10 marks for the whole question] *Mathematical Aspects*

(a) [2 Marks] Compute the magnitude (length) of the vector (2, 3, −6) and normalize the vector.

(b) [4 Marks] A spotlight S is defined with a direction vector $d = (-1, -2, 2)$ as shown in the figure below. An animation sequence requires a single rotation of this spotlight from its initial direction d towards the direction (0, −1, 0). Compute the angle of rotation θ, and the vector r about which this rotation must be performed. It is not necessary to normalize the vector r.

(c) **[2 Marks]** Write the transformation defined in the code segment below as a product of two 4x4 matrices. You are not required to multiply the matrices.

```
glPushMatrix();
glTranslatef(-2, 3, 8);
glScalef(2, 0.5, 4);
glutSolidTeapot(1);
glPopMatrix();
```

(d) **[2 Marks]** What transformation does the following matrix represent?

$$\begin{bmatrix} 0.866 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \\ -0.5 & 0 & 0.866 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(a) [2 Marks] Compute the magnitude (length) of the vector (2, 3, 6) and normalize the vector.

Magnitude = 7
Normalize is (2/7, 3/7, -6/7) ✓

(b) [4 Marks] A spotlight S is defined with a direction vector d = (1, 2, 2) as shown in the figure below. An animation sequence requires a single rotation of this spotlight from its initial direction d towards the direction (0, 1, 0). Compute the angle of rotation , and the vector r about which this rotation must be performed. It is not necessary to normalize the vector r.

r = (2, 0, 1) ✔
Need to normalize d first and then dot product
angle = 48.19 degrees

✓
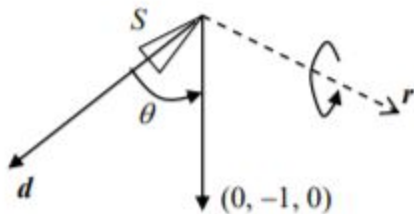
If you want, put a ✓ Pick a colour or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

(c) [2 Marks] Write the transformation defined in the code segment below as a product of two 4x4 matrices. You are not required to multiply the matrices.

```
glPushMatrix();
        glTranslatef(-2, 3, 8);
        glScalef(2, 0.5, 4);
        glutSolidTeapot(1);
glPopMatrix();
        float matrix1[16] = {1, 0, 0, -2,
                                        0, 1, 0, 3,
                                        0, 0, 1, 8,
                                        0 0, 0, 1};
        float matrix2[16] = {2, 0, 0, 0,
                                        0, 0.5, 0, 0,
                                        0, 0, 4, 0,
                                        0, 0, 0, 1};
        glPushMatrix();
        glMultMatrixf(matrix1);
        glMultMatrixf(matrix2);
        glutSolidTeapot(1);
        glPopMatrix();
```

✓

Dont think you need to answer this question as if its code, just write the two matrices.

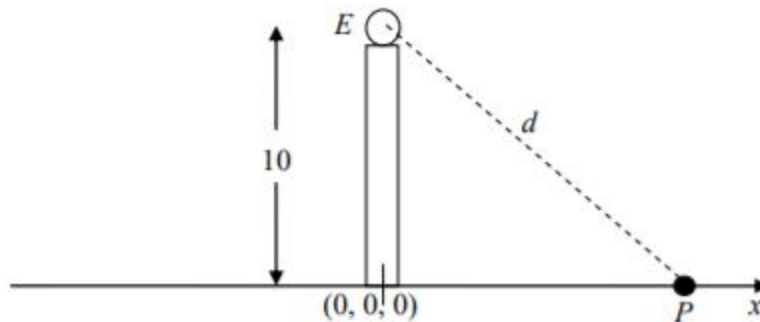(d) [2 Marks] What transformation does the following matrix represent?

Rotation about the y-axis at 30 degrees

✓

Question 5

Question 5. [10 marks for the whole question] *Viewing*

In an OpenGL program, a camera E is required to be positioned at the point $(0, 10, 0)$ as shown in the figure above. The camera looks at a point P on the x-axis, and the distance EP is d units.

(a) [2 Marks] For the camera configuration given above, write the values of all nine parameters of the following function:

    gluLookAt(ex, ey, ez, lx, ly, lz, upx, upy, upz);

(b) [3 Marks] Draw a figure in the answer booklet showing the points E, P, the x-axis, and the three axes u, v, w of the eye coordinate frame. Indicate an axis perpendicular to the plane of the paper with a '+' sign (eg. $+u$) if it is directed towards the reader, or a '−' sign if it is oriented away from the reader.

(c) [2 Marks] Write the eye-coordinates of the points E and P.

(d) [3 Marks] Briefly describe in words how the u-axis of the eye coordinate frame is computed.

(a)) [2 Marks] For the camera configuration given above, write the values of all nine parameters of the following function:
gluLookAt(ex, ey, ez, lx, ly, lz, upx, upy, upz);

    gluLookAt (0, 10, 0, px, 0, 0, 0, 1, 0)
I put sqrt(d^2 - 100) instead of px, since the coordinates of P are not specifically given

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

(b) [3 Marks] Draw a figure in the answer booklet showing the points E, P, the xaxis, and the three axes u, v, w of the eye coordinate frame. Indicate an axis perpendicular to the plane of the paper with a '+' sign (eg. +u) if it is directed towards the reader, or a " sign if it is oriented away from the reader.
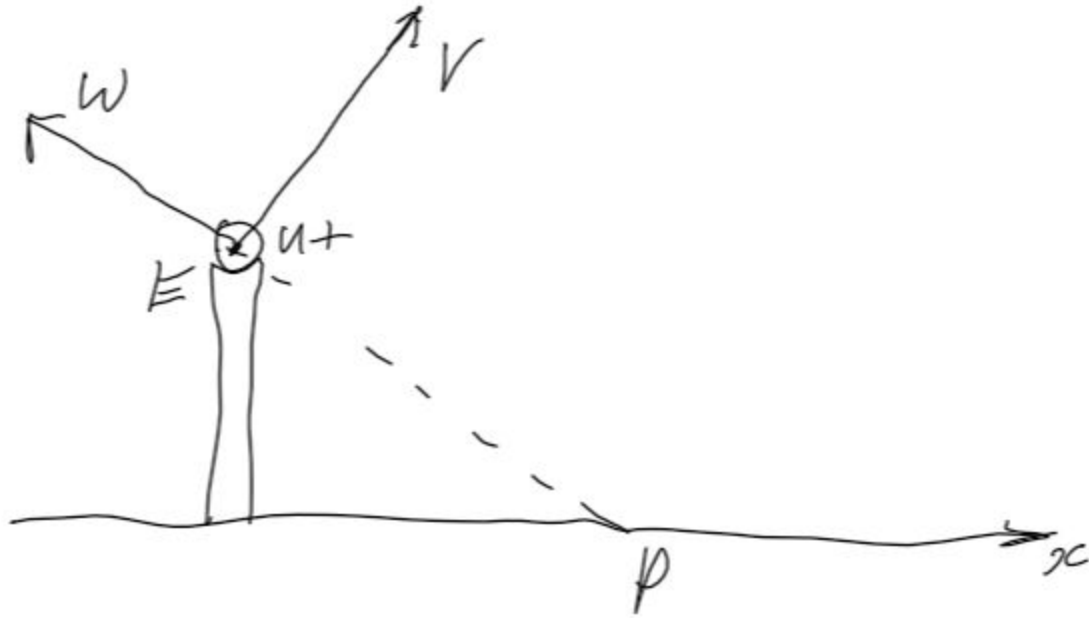
(unconfirmed)

(c) [2 Marks] Write the eye-coordinates of the points E and P

E = (0, 0, 0), P = (0, 0, -d) ✓

(d) [3 Marks] Briefly describe in words how the u-axis of the eye coordinate frame is computed

The w-axis is calculated as opposite direction from the view vector. Then, the u-axis is in the direction of the cross product of the up vector and w-axis.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

Question 6

Question 6. [8 marks for the whole question] *Ray tracing.*
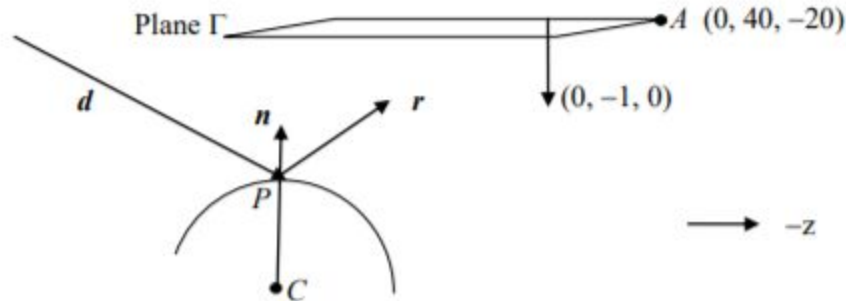
A primary ray in the direction d hits a reflective sphere at P and bounces off in the direction r. The ray r then hits a ceiling plane Γ as shown in the figure above.

(a). [1 Mark] If the centre of the sphere is at $C = (2, 6, -10)$, and the point of intersection is given by $P = (2, 10, -10)$, compute the **unit** normal vector n at P.

(b). [3 Marks] The direction of the reflected ray is given by $r = -2\,(d \cdot n)\,n + d$. If $d = (0, -0.6, -0.8)$, find the components of the vector r.

(c). [3 Marks] The plane Γ passes through the point A, and has a normal vector $n_1 = (0, -1, 0)$. Compute the ray parameter t at the point of intersection of the ray r and the plane, using the equation

$$t = \frac{(A - P) \cdot n_1}{r \cdot n_1}$$

(d) [1 Mark] What does the above value of t represent?

(a) [1 Mark] If the centre of the sphere is at C = (2, 6, 10), and the point of intersection is given by P = (2, 10, 10), compute the unit normal vector n at P.

(0, 1, 0)✓

(b) [3 Marks] The direction of the reflected ray is given by r = 2 (d •n) n + d. If d = (0, 0.6, 0.8), find the components of the vector r.

(0, 0.6, -0.8)✓

(c) [3 Marks] The plane passes through the point A, and has a normal vector n1 = (0, 1, 0). Compute the ray parameter t at the point of intersection of the ray r and the plane, using the equation

50 ✓

(d) [1 Mark] What does the above value of t represent?

t represents the distance from the point P and the point that hits the plane at
P + t * r

Question 7

Question 7. [15 marks for the whole question] *Vertex and Fragment Shaders*

The source codes for a vertex shader and a fragment shader are given below.

Vertex shader:
```
layout (location = 0) in vec4 position;
layout (location = 1) in vec4 color;
uniform mat4 mvpMatrix;
out vec4 theColor;
void main()   {
     gl_Position = mvpMatrix * position;
     theColor = color;

}
```

Fragment shader
```
in vec4 theColor;
void main() {
     if (theColor.r < 0.5) discard;
     gl_FragColor = theColor;
}
```

Give a brief description of each of the 6 lines of code (excluding "void main()") in the vertex shader, and each of the 3 lines of code in fragment shader. For declaration statements, explain the role of storage qualifiers in determining how values are received from the application or passed to other shaders. For statements within the main() function, explain what computations are done, or why a variable is assigned to another.

In vertex shader line 1, the in means that the vec4 position is recieved from somewhere, in this case the position vector is recieved from the application at location 0.

In vertex shader line 2, the in means the vec4 color is recieved from somewhere, in this case the color vector is recieved from the application at location 1.

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

In vertex shader line 3, the uniform means that this attribute is taken from the application, mvpMatrix is a 4 by 4 matrix provided by the application.

In vertex shader line 4, the out means the this is an output of the vertex shader to pass to the next
shader, theColor is a vector of size 4.

In vertex shader line 6, the computation of converting position which is in world coordinates by multiplying
with the model view project matrix gets us the gl_Position which will be in clip coordinates.

In vertex shader line 7, the color is simply passed into theColor for the next shader.

In fragment shader line 1, theColor which is a vector of size 4 is received from the Vertex shader as it has the in storage qualifier.

In fragment shader line 3, this condition is used to discard theColor if theColor has a red value that is less than 0.5

In fragment shader line 4, gl_FragColor will be set to as theColor.

Question 8

Question 8. [10 marks for the whole question] *Bezier Curves.*

A quadratic Bezier curve is generated by combining three control points using second degree Bernstein polynomials ($(1-t)^2$, $2t(1-t)$, t^2).

(a) [4 Marks] Given three control points $P_0 = (3, 0, 0)$, $P_1 = (1, 0, 1)$, $P_2 = (-1, 4, 2)$, write the parametric equations for $x(t)$, $y(t)$, $z(t)$ for points on the Bezier curve.

(b) [4 Marks] Show that every point on the above Bezier curve satisfies the equations $y = z^2$ and $x + 2z = 3$.

(c) [2 Marks] A general n^{th} degree Bezier curve is formed using $n+1$ control points $P_0, ..P_n$. Outline two geometrical properties satisfied by the Bezier curve.

(a) [4 Marks] Given three control points P0 = (3, 0, 0), P1 = (1, 0, 1), P2 = (1, 4, 2), write the parametric equations for x(t) , y(t), z(t) for points on the Bezier curve.

x(t) = 3-4t

y(t) = 4t^2
z(t) = 2t ✓

(b) [4 Marks] Show that every point on the above Bezier curve satisfies the equations y = z^2 and x + 2z = 3.

y = 4t^2 = (2t)^2 = z^2

x + 2z = (3 - 4t) + 2 * (2t) = 3 - 4t + 4t = 3 ✓

(c) [2 Marks] A general n th degree Bezier curve is formed using n+1 control points P0, ..Pn. Outline two geometrical properties satisfied by the Bezier curve.
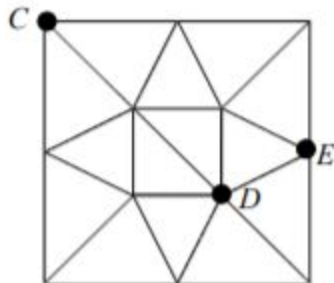
1. The bezier curve always lies within the convex hull of the control points
2. The bezier curve passes through the first and the last control points.

Question 9

Question 9. [10 marks for the whole question] *Tessellation Shaders*

(a) [5 Marks] Consider a surface design application where a 4x4 control polygonal grid is used to generate a bi-cubic Bezier surface. Briefly describe how the tessellation stage of the OpenGL-4 pipeline could be effectively used for this application, clearly outlining the main computations performed inside the control and evaluation shaders. You are not required to write any equations or code as part of your answer.

(b) [5 Marks] A quad domain is tessellated as shown in the figure below. Write the inner and outer tessellation levels used in generating the figure, and give the tessellation coordinates of the points C, D and E.

(a) [5 Marks] Consider a surface design application where a 4x4 control polygonal grid is used to generate a bi-cubic Bezier surface. Briefly describe how the tessellation stage of the OpenGL-4 pipeline could be effectively used for this application, clearly outlining the main computations performed inside the control and evaluation shaders. You are not required to write any equations or code as part of your answer.

Unanswered

(b) [5 Marks] A quad domain is tessellated as shown in the figure below. Write the inner and outer tessellation levels used in generating the figure, and give the tessellation coordinates of the points C, D and E.

    2222; 33
    C: (0, 1)
    D: (2/3, 1/3)
    E: (1, 1/2)

Question 10

Question 10. [7 marks for the whole question] *Geometry Shader*

A geometry shader provides the following useful features:

(i) The shader can process a primitive as a whole.

(ii) The shader can generate new primitives.

(iii) The shader can discard a primitive.

Give short descriptions of three applications, each using one of the above features.

(i) The shader can process a primitive as a whole.

To perform lighting calculations ont he repositioned mesh triangles from the tessellation evaluation shader, we required surface normal vectors at the vertices, since the geometry shader receives all vertices of each triangle of the mesh, we can perform the lighting calculation inside the geometry shader. Useful for rendering terrains and Bezier surfaces.

(ii) The shader can generate new primitives.

When using adjacent triangles, we may detect crease edges that are detected, we can then generate new primitives such as a triangle strip and assign a colour value to it to represent this crease edge.

(iii) The shader can discard a primitive.

This is useful for culling backfaces of a three dimensional model, so that we don't need to rasterize these backface that won't be viewed anyway, culling them in the geometry shader means that they won't need to be removed after rasterization in the fragment stage.

2015 - Mostly Unanswered

Question 1

```
glPushmatrix()
    glTranslate(4,0,0)
    glRotate(-90, 1, 0, 0)
    glScale(0.5, 0.5, 2)
    drawCylinder()
glPopMatrix()

glPushmatrix()
    glTranslate(6, 2, 0)
    glsScalef(0.5, 0.5, 0.2)
    drawCylinder()
glPopMatrix()
```

Question 2

a)

If you want, put a ✓ **Pick a colour** or ✔ next to answers you agree with.

put a ✓ if you think you would fail an open book exam ✓ (Seng201 flashbacks)

A1: 0, 0
A2: 0, 1/2
B1: 5/3, 0
B2: 5/3, 1/2
C1: 10/3, 0
C2: 10/3, 1/2
D1: 5, 0
D2: 5, 1/2

P: 1/4, 1
Q: 3/4, 1
B2: 1/4, 1/2
C2: 3/4, 1/2

b)
???

Question 3

a)
b)

Question 4

Question 5

Question 6

Question 7

Question 8

Question 9

Question 10