HÁSKÓLINN Í REYKJAVÍK
REYKJAVIK UNIVERSITY

Computer Graphics
T- 511 – TGRA

Final Exam

Teacher: Kári Halldórsson

Date: 25. November, 2016

Time: 14:00 - 17:00

Helping materials: non-programmable calculator

Name: _____

Kt.: _____

Answers can be given in English and/or Icelandic.

1. **(10%) Shaders and lighting**
   Describe the difference between per-vertex lighting and per-fragment/per-pixel lighting.
   In each case bear the following questions in mind:
   What are the advantages and drawbacks of the method?
   What calculations happen where, and what values are set to the final result?
   How is the data processed between different parts of the calculations?

2. **(10%) Depth testing**
   Describe the process of depth testing in OpenGL.
   What is the purpose of it?
   What values are used and how, and where/when are they calculated?
   How is the data processed between different parts of the calculations?

3. **(20%) OpenGL pipeline**
   Apart from the shader programs themselves, which a programmer can write (*no need to describe what happens inside the shaders, only what data is input into them and what data they affect/output*) **briefly describe each part of the OpenGL graphics pipeline**. What is the state of data at each point in the pipeline, from first input to final output, what data does each module use and what is each module's purpose?

   When describing each module, and the pipeline in general, each can be viewed as a fairly detailed description of ***1)*** *it's purpose,* ***2)*** *the data "before" and* ***3)*** *the data "after"*, but you don't need to describe the algorithm or calculation of each one. As for the shaders, describe how/when they are run and in general their "input" and "output", how this input is prepared and how the output is used afterwards.

   Try to view this as several 4%-5% parts (general structure + modules) rather than a single 20% block. Next page is blank and you can also use the back of this page and the next if you need, to keep the answer clear and organized.

   Feel free to use diagrams to aid your answer.

4. (25%) Matrices and transformations

a) (10%)
A camera is set up to be positioned in (-1,-2,4)
looking at the point (3,1,-2).
It has an up vector (0,1,0).
Set up the values in a matrix that represents this position and orientation of a camera.
Which matrix in your shader should be set to these values?

b) (5%)
The camera should have a field of view of 30°, an aspect ratio of 16:9, a near plane at 1 and a far plane at 11. Find the exact values for a matrix that calculates this camera.
Which matrix in your shader should be set to these values?

c) (5%)
Vertex data should be drawn into a coordinate frame that has been translated by (3, 2, -6) and then rotated by 30° about the x-axis. Represent this coordinate frame in a matrix. Which matrix would this commonly be?

d) (5%)
A vertex is run through the vertex shader.
It has the position values (1, 2, 3).
Given the matrix values calculated in parts a, b & c, what values will the vertex shader set to gl_Position?
Will this vertex be within the viewing volume and thus (other tests notwithstanding) be rendered as part of the final image? Explain.

5. **(10%) Lighting Calculations**

A single light is in the light model in an OpenGL program. It has the ambient values (0.0, 0.0, 0.0), diffuse values (0.5, 0.3, 0.7), specular values (0.3, 0.8, 0.7) and position (3.0, -2.0, 2.0). There is also a global ambient factor of (0.3, 0.2, 0.4) in the light model. A camera is positioned in (-3.0, 3.0, 4.0) and looks towards P.

P has the color values: ambient (0.4, 0.7, 0.3), diffuse (0.4, 0.7, 0.2) and specular (0.6, 0.6, 0.6). It has a shininess value of 21. It has the position (0.0, 1.0, 1.0) and a normal (1.0, -1.0, 3.0).

What will be the green color value for P on the screen?

6. (5%) Blending

Before running the OpenGL pipeline the following function were called:

```
glEnable(GL_BLEND);
glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA);
```

Now the pipeline is rendering a fragment at pixel coordinates (4, 5) in the frame buffer. These are current values directly after the fragment shader is run:
gl_FragColor = (0.2, 0.4, 0.7, 0.3)
frame_buffer[4,5] = (0.6, 0.9, 0.3, 1.0)

What color will the pixel (4, 5) in the frame buffer be set to before the current algorithm continues to the next fragment?

Calculate the color if the previous lines had instead been:

```
glEnable(GL_BLEND);
glBlendFunc(GL_DST_COLOR, GL_ONE);
```

7. **(10%) Bezier motion**
   Scalars in bezier curves can be found by factoring Bernstein polynomials:
   $BL = ((1- t) + t)^L$ for a bezier curve with $L + 1$ control points.

   The camera is moved along a bezier curve with 4 control points.
   P1 = (12, 6, 8), **P2** = (10, 2, 12), **P3** = (5, 1, 15), **P4** = (12, 5, 16)

   The motion should start 10 seconds after the program starts and it should end 40 seconds later, 50 seconds after the program starts.
   What is the camera's position 40 seconds after the program started?

8. **(10%) Rasterization**
   Three vertices of a triangle have been sent through the OpenGL pipeline. They have the following pixel positions as well as values for the varying variable v_d:
   P1: position = (7,3) – v_d = 4
   P2: position = (5,7) – v_d = 12
   P3: position = (13,9) – v_d = 7
   What will the fragment shader value of v_d be set to at pixel (8,5)?

Bonus 3%
In which movie does the following dialog occur?
Who is character A?
Who is character B?

A: Hey <B>, have you ever been mistaken for a man?
B: No. Have you?