EXAM IN

COMPUTER GRAPHICS

TSBK07

(TEN1)

Time:	2nd of November, 2018, 8-12
Room:	G34
Teacher:	Ingemar Ragnemalm, available by phone only.
Allowed help:	None
Requirement to pass:	Grade 3: 21 points Grade 4: 31 points Grade 5: 41 points ECTS: C: 21 points B: 31 points A: 41 points

Answers may be given in swedish or english.

- Wish us luck!
- I wish you skill!
[Martin Landau, ”Mission Impossible”]

1. OpenGL programming

a) Below follows a few lines of GLSL code that your examiner dreamed up one stormy night. Not only is the code incomplete and rather meaningless, but there are some details that will prevent if from working, or even compiling.

```
#include ”glsl.h”
int main(int argc, char **argv)
{
      vec4 out;
      uniform texture2D myTexture;
      uniform f;
      float gl_s = f/gl_TexCoord[0];
      float gl_t = f/gl_TexCoord[1];
      vec4 color = myTexture[gl_s, gl_t];
      if (color.r = 0)
            printf(”Red is zero\n”);
      out = color.axsw;
}
```

What errors or otherwise ”bad” code can you find? A few words explaining the problem for each is enough (like ”divide by zero”). Each error should only be given once. Hint: There is more than one error. For full score, a majority of the errors must be found and correctly described.

(3p)

b) A common situation in shader programming is to interpolate values between vertices. Give an example in simple GLSL (or GLSL-like code) how this is programmed.

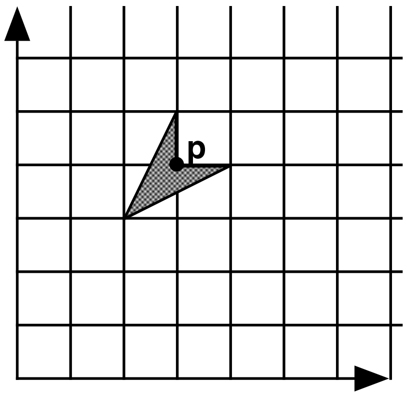
(2p)

c) What is the purpose of a VAO (vertex array object)?

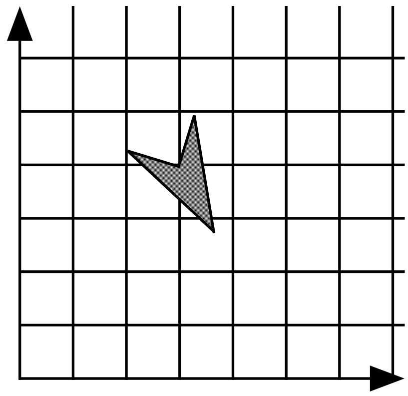
(2p)

2. Transformations

a) In the figure below, a 2D shape is shown together with a point **p**. Produce a sequence of 3x3 matrixes that define a transformation that rotates the shape (or anything else) around **p** by an angle ϕ. The contents of each matrix should be given. You don't have to multiply the matrices together.



Original shape and position, and the point p that the shape is rotated around



After transformation

(3p)

b) Which one(s) of the following statements are true (given names and symbols used in the course)?

(1) Translation: $T(\mathbf{p})^{-1} = T(\mathbf{p})^T$
(2) Rotation: $R(\phi)^{-1} = R(\phi)^T$
(3) Mirroring: $M = R(\pi)$
(4) Composite transformations: $T(\mathbf{p})*S(a) = S(a)*T(\mathbf{p})$
(5) Composite transformations: $R(\phi)*S(a) = S(a)*R(\phi)$
(6) Cross product: $\mathbf{a}\times\mathbf{b} = \mathbf{b}\times\mathbf{a}$

(2p)

3. Light, shading and ray-tracing

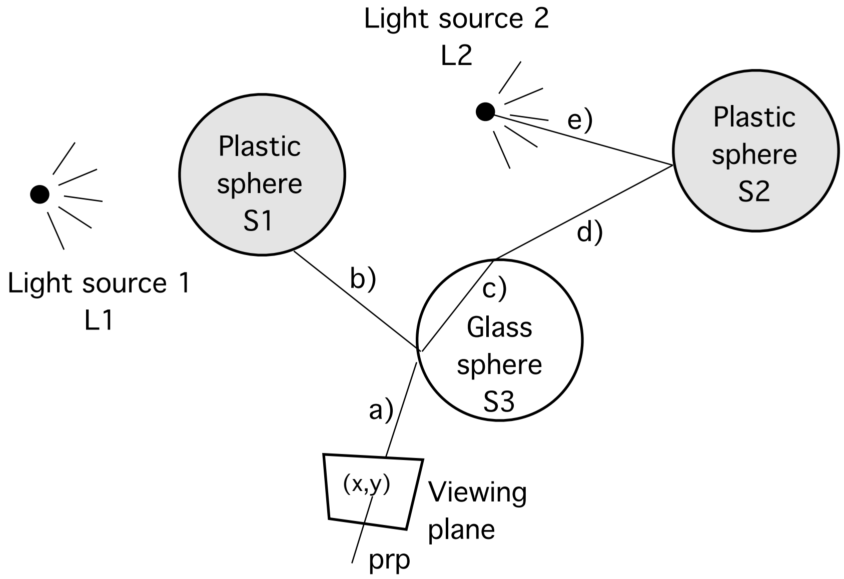
a) Jittering can produce many interesting effects in ray-tracing. Give descriptive names (or short descriptions) for three of them and clarify in a figure how they can be produced.

(3p)

b) A student describes the ray-tracing shown below like this: "A ray starts at light source L2, bounces off sphere 2, is then refracted through S3 following Snell's law, and finally hits the viewing plane, contributing to the light level of the pixel (x, y). Rays from source 1, however, will be blocked by S1 and will therefore not contribute from that direction."

However, the student didn't score many points. What is wrong in this description? How is the ray shown really computed, and why doesn't L1 contribute to (x,y) in a real ray-tracer?

(4p)



4. Surface detail

a) Describe, with formulas, how to map a texture to an object, vertex by vertex, using a cylinder as intermediate surface. You may use the $\tan^{-1}$ (by one variable). Texture coordinates should be normalized properly.

(4p)

b) After mapping point-to-point in a), one problem remains. You can not simply map every vertex to texture space and assume a correct result. Why, what will happen? Suggest a remedy.

(2p)

5. Curve generation

a) Under which circumstances will a Bézier curve fulfill C^1 contunity? How about G^1 continuity?

(2p)

b) Demonstrate how Horner's rule can be used to accelerate the evaluation of polynomials.

(2p)

6. Miscellaneous

a) Anti-aliasing can be performed by supersampling. Under what conditions (what kind of signal) will supersampling produce a good result, and why?

(3p)

b) Given a point in a plane, describe how you can test whether that point is in a triangle in the same plane.

(3p)

7. Collision detection and animation

a) Suggest a method to reduce the number of broad phase tests for scenes with a large number of objects.

(2p)

b) Using a figure, give examples of two depth cues that are useful in 2D (pseudo-3D) graphics.

(2p)

c) Describe how a simplified collision resolving can be performed between two identical spheres, given their positions, radii and speed vectors. The elasticity should be controlled by a variable.

(3p)

8. Visible surface detection and large worlds

a) A common VSD method has problems with transparency. Describe the problem and a remedy.

(2p)

b) How does geomipmapping work? Outline the principles for the method with a figure. How do we select and switch between resolutions?

(3p)

c) Describe the difference between view plane oriented billboards and viewpoint oriented billbaords. One of them is particularly easy to implement. Which one, and how?

(3p)