EXAM IN

COMPUTER GRAPHICS

TSBK05

Time:	6th of March, 2004, 14-18
Room:	R42, R34, R35, R36, R37, R41
Teacher:	Ingemar Ragnemalm, visits around 15 and 17
Allowed help:	None
Requirement to pass:	Grade 3: 21 points Grade 4: 31 points Grade 5: 41 points C program: G: 21 points. VG: 36 points. ECTS: E: 21 points D: 26 points C: 31 points B: 36 points A: 41 points

Answers may be given in swedish or english.

Good luck!

1. OpenGL programming

a) In typical OpenGL code, glLoadIdentity is not likely to appear at the beginning of the redrawing routine (i.e. the display() function). Why? What is used instead?

b) You need to include some trees in a 3D scene. You decide to use billboards to implement them, using a single rectangle for each tree. There are several issues where this is somewhat different from using ordinary polyhedra. There is one difference in each in the following three topics:

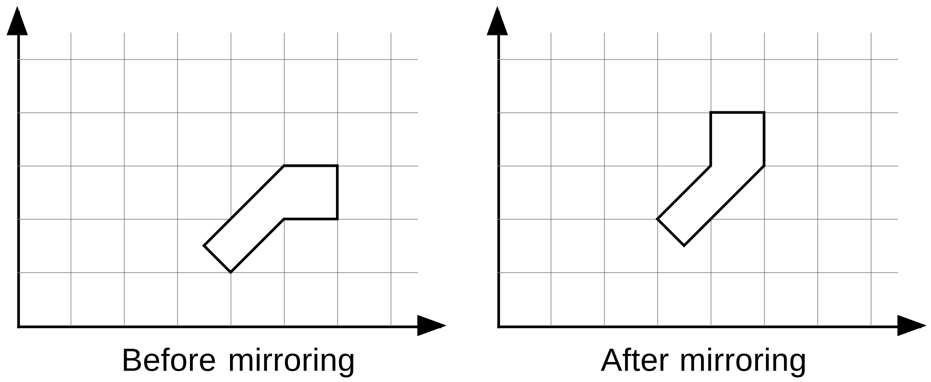
- textures
- position
- drawing order

Outline what these problems are and their solutions. You do not have to write code, just describe with words.

(4p)

2. Transformations

a) In the following figures, a 2D shape is mirrored over a line. Give a sequence of matrices that perform this transformation, with the contents of each matrix as well as the order of multiplications. You do not have to multiply the matrices together. Use numeric values or clearly defined symbols for distances and angles used.

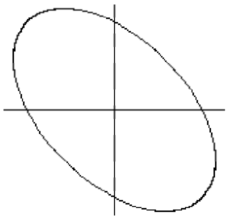


b) Write OpenGL code that performs the operation in a), transformations as well as drawing the polygon. Hint: Mirroring can be performed by glScale! If you don’t remember the exact syntax of some call, a similar syntax is acceptable, as long as it is easy to translate line by line to real OpenGL code.

(6p)

3. Curve generation

a) An ellipse is given by the formula $x^2 + y^2 + xy - r^2 = 0$. This is not an axis-aligned ellipse, but an ellipse rotated 45 degrees (see figure).



Derive the midpoint algorithm with incremental updating of the decision variable to plot a section of this curve, 4-connected, starting at (0, r) and continuing in positive x direction. Also, calculate the starting value for the decision parameter.

b) In order to draw any part of the curve in a), you must use different expressions for different parts of the curve. Plot in a figure the points where the algorithm must switch expressions, and clarify why. (You don't have to derive the incremental updating expressions for the other parts.)

c) How can you ensure C^1 continuity between two Bezier curve segments? Clarify with a figure or formulas.

Hint: $BEZ_{0,3} = (1-u)^3$
$BEZ_{1,3} = 3u(1-u)^2$
$BEZ_{2,3} = 3u^2(1 - u)$
$BEZ_{3,3} = u^3$

(8p)

4. Mapping techniques

a) Spherical coordinates can be defined by

$x = R \cos\varphi \cos\theta$
$y = R \cos\varphi \sin\theta$
$z = R \sin\varphi$

Write formulas for spherical texture mapping, mapping x, y, z to texture coordinates (s, t), normalized to the interval [0, 1].

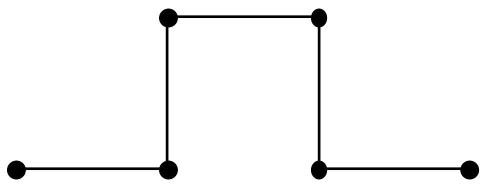
b) How much is the added memory cost for using mip-mapping? Motivate your statement numerically.

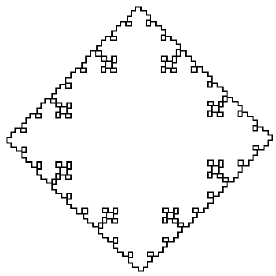
c) When rendering texture mapped polygons, linear interpolation is generally used. What values should be interpolated when rendering perspective correct texture mapping? Why?

(7p)

5. Miscellaneous

a) A fractal (below right) was generated by the generator in the figure below and a square as initiator. What is the fractal dimension of this fractal? (Hint: $ns^D = 1$)





b) Given the camera position **p**, the forward vector **n**, and the up vector $\mathbf{v_{up}}$ (which is not necessarily orthogonal to n), how do you build a matrix that can be used for the world-to-camera transformation?

(5p)

6. Light, shading and ray-tracing

a) Describe, using text and figures, how one object can cast a shadow onto another in ray-tracing.

b) Given scene geometry and parameters (light source levels, material parameters) derive a formula for the diffuse light level for a specific point in the scene. (Shadows and visible surface detection are ignored.) Define vectors and angles with a figure.

c) The form factor F_{jk} in the radiosity equation, the part of the light energy emitted from j that hits k, can be calculated as the solid angle of k seen from j. This could be calculated from geometry alone if there were not occlusions. Outline, with a figure, a practical method to calculate the form factors F_{jk}, with support for occlusions.

(7p)

7. Collision detection

a) Describe, with appropriate formulas and figures, how you calculate the intersection between a ray and a triangle.

b) Describe how camera-polygon collissions are detected.

c) In the broad phase (3D collision detection), suggest three different shapes that could be used for collision detection. Why is it not necessarily optimal to use the one that gives the simplest calculations?

(7p)

8. Visible surface detection

Suppose that you have a large terrain defined by a 2D grid. The grid is *very* large, describing a huge world. Describe how you can efficiently reduce the number of polygons that are passed to the renderer (backface culling excluded). You should be able to suggest *two* important optimizations, hinted below as a) and b). Use figures to clarify the methods that you propose.

a) Optimization from visibility. Describe the method with emphasis on how this is done efficiently.

b) Optimization due to scale. Describe the method with emphasis on how this is done with minimal visible errors.

(6p)