

**EXAM IN**  
**COMPUTER GRAPHICS**  
**TSBK07**

Time: 23th of May, 2008, 14-18

Room: T1, T2

Teacher: Ingemar Ragnemalm,  
visits around 15

Allowed help: None

Requirement to pass: Grade 3: 21 points  
Grade 4: 31 points  
Grade 5: 41 points

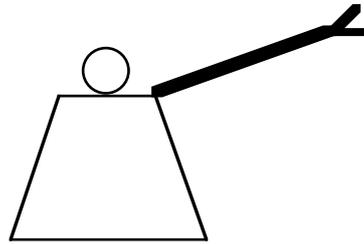
ECTS:  
C: 21 points  
B: 31 points  
A: 41 points

Answers may be given in swedish or english.

**Good luck!**

## 1. OpenGL programming

a) A robot with a head and an arm is shown in the figure below. The robot can rotate around the Y axis and move in the XZ plane. The head can rotate around Y, and the arm can rotate around the Z axis.



*Robot with an arm and a head*

The robot position is given as  $r_x$  and  $r_z$ , and its rotation is  $\alpha$ .

The head is located at the height  $h$  above the robot origin and its rotation is  $\beta$ .

The arm is fastened at  $(w, h, 0)$  and rotated around Z by the angle  $\delta$ .

The calls `drawBody()`, `drawHead()` and `drawArm()` are given for drawing the parts in their local coordinate systems.



*Each body part is drawn by a function which you do not need to write*

Write OpenGL code for drawing the robot using the calls above and standard OpenGL calls.

You should not write code for the entire program, only for the parts that deals with the drawing of the robot.

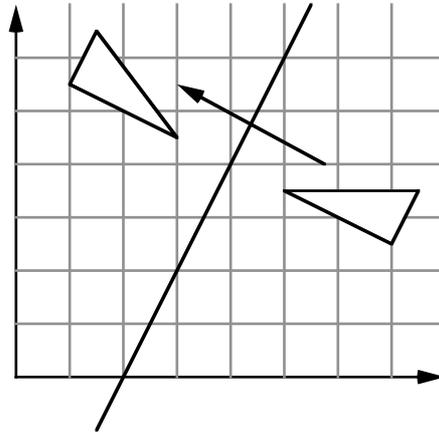
(3p)

b) Explain the effect of the modifiers “varying”, “attribute” and “uniform” in GLSL.

(2p)

## 2. Transformations

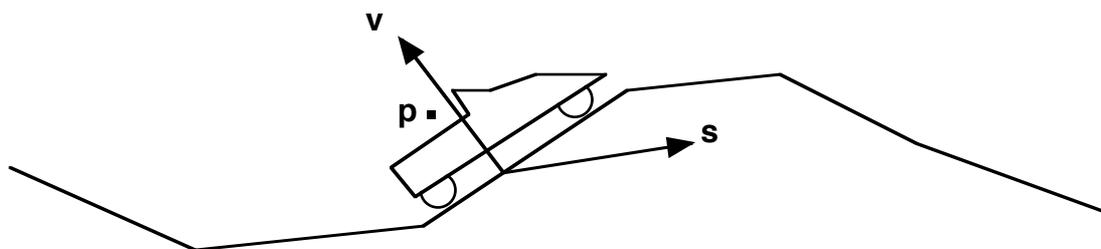
a) The following figure illustrates mirroring over a line, in 2D. Give a sequence of 3x3 matrixes, each defining one basic geometric transformation (translation, rotation around origin, mirroring, scaling, skewing), that define a transformation that produces the mirroring according to the figure. You don't have to multiply the matrices together, but the multiplication order for transformation of a point should be given.



(4p)

b) Linus and Linnea are making a car racing game as their computer graphics project, where the car drives in a terrain. The game is in first perspective, so the player is viewing the scene from the inside of the car.

Since the car drives in a terrain, it will often drive in slopes. Linus and Linnea have written functions for finding the height of the terrain as well as its slope in a specific point.



*A car in a terrain*

Your task is to help them with the camera placement. The camera position, inside the car, is given as the point  $\mathbf{p}$ . The slope of the ground is given by its normal vector  $\mathbf{v}$ . The vector  $\mathbf{s}$  is the forward vector of the car from the previous frame; it may not be entirely correct due to changes of slope as the car moves.

So, the camera must be placed in  $\mathbf{p}$ , aligned with the ground, and its forward direction should be as close as possible to the old forward vector  $\mathbf{s}$ .

Derive a camera placement matrix that fulfills these premises.

(3p)

### 3. Curve generation

a) The Bézier curve is defined by the following formulas:

$$\text{BEZ}_{0,3} = (1-u)^3$$

$$\text{BEZ}_{1,3} = 3u(1-u)^2$$

$$\text{BEZ}_{2,3} = 3u^2(1-u)$$

$$\text{BEZ}_{3,3} = u^3$$

Under what conditions will two Bézier curve segments fulfill  $C^1$  or  $G^1$  continuity?  
Show mathematically that this is the case.

(4p)

b) Derive Horner's rule for evaluating polynomial functions. What is the computational advantage?

(2p)

### 4. Surface detail

a) Describe how a view plane oriented billboard can be implemented.

(2p)

b) Cylindrical coordinates  $(\theta, v)$  can be defined by

$$x = R \cos\theta$$

$$y = R \sin\theta$$

$$z = v$$

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

(3p)

### 5. Light, shading and ray-tracing

a) Describe how anti-aliasing is performed in ray-tracing.

(2p)

b) Motivate why radiosity generally only supports Lambertian surfaces. What parts of the radiosity light model has problems modelling specular surfaces?

(2p)

c) The popular three-component illumination model can be described by the following formula:

$$I = k_a I_a + k_d I_l (\mathbf{N} \cdot \mathbf{L}) + k_s I_l (\mathbf{V} \cdot \mathbf{R})^n$$

Explain what each symbol represents, using text and figure(s) as appropriate.

(3p)

## 6. Miscellaneous

a) The Julia set is defined by the formula

$$z_{k+1} = z_k^2 + L$$

Describe how the algorithm works to produce an image.

(3p)

b) Describe how Bresenham's line drawing algorithm works for drawing a line from (10, 10) to (20, 14). Hint:

$$p_k = 2\Delta y \cdot x_k - 2\Delta x \cdot y_k + 2\Delta y + \Delta x \cdot (2b-1)$$
$$p_0 = 2\Delta y - \Delta x$$

Incremental update is required for full score.

(3p)

c) Using a figure, illustrate the difference between the odd-even rule and the non-zero winding number rule. The figure should have at least one area where the two rules give different result.

(2p)

## 7. Collision detection

a) Describe, using a figure, how collision detection can be performed between two polyhedra (i.e. narrow phase).

(2p)

b) Describe, using figures, how camera-polygon collisions are resolved.

(2p)

c) How do you test if a line segment intersects a triangle, granted that you know where the corresponding infinite line intersects the plane defined by the triangle?

(2p)

## 8. Visible surface detection and large worlds

Linus&Linnea have another problem with their car-racing game from part 2. As long as their terrain is small, the game runs nicely, but when they enlarge the terrain and place many objects (other cars, trees and rocks) in it, it no longer runs fast enough.

The game was implemented with OpenGL, rendering all polygons on a fairly fast GPU. They have found that the problem is that the polygon count is too high for the GPU. They need to render the scene in not more than half the time. They do not want to sacrifice more visual quality than necessary. Removing texture mapping for any surface is not an option, neither is lower screen resolution. The loss of visual quality should be almost unnoticeable.

- a) Suggest two relevant methods that would improve speed with no visual change whatsoever (assuming that they have not already implemented it). (2p)
  
- b) Suggest two relevant methods that would compromise rendering speed and rendering quality for this application (again assuming that they have not already implemented that method. Hint: Different parts of the scene can be handled in different ways. Describe what parts should be rendered one way or the other. (2p)
  
- c) With the camera placed inside the car, Linus and Linnea add a semitransparent polygon as the front window of the car. Unfortunately, they now only see their skybox, while terrain, trees etc are no longer visible. If the window polygon is removed, all is well again. What did they do wrong and what should they do about it? (2p)