Exam Computer Graphics (INBCG-08) - Edition Spring 2013

This examination consist of **five** exercises with in total 20 subquestion. Each subquestion weighs equally for the final grade. In all cases: **EXPLAIN YOUR ANSWER. Use sketches and formulas where needed to clarify your answer. Read first all questions completely.** Aim at compactness and clarity and simplicity (no unnecessary embellishments). Use additional functions and procedures or pseudocode if desired. Explain all notations (used in drawings and formulas) clearly before being used in your explanation. The use of the book, copies of slides, notes and other material is not allowed during the exam.

1. Let is consider the following basic techniques for computer graphics. Explain, for each technique, the issues indicated in the respective question.

a) Painter's, or depth-sort, algorithm for hidden surface removal: This algorithm is simple, but had some problems with correctly depth-sorting any configuration of polygons in 3D. Give two *different* examples of 3D polygonal scenes, each with a different camera position, where the painter algorithm would produce the incorrect rendering of the scene. For each example, illustrate the rendering problems by an actual sketch of the 3D scene and viewpoint so that the hidden surface removal problems are evident.

b) Phong illumination model: This illumination model describes the illumination of a 3D point using different components. Explain *two* differences between the diffuse and specular components of this model. Illustrate your explanation by a 3D sketch involving the camera parameters, 3D shaded point, and light source.

c) In your practical assignment, you have implemented a raytracing renderer. Based on your experience in this assignment, describe *one realistic shading effect* that raytracing can produce, and which cannot be achieved with the local point-lighting model implemented in, for example, OpenGL.

d) In computer graphics, most implementation use so-called *homogeneous coordinates.* Give a definition of homogeneous coordinates (in 2D or 3D, as you wish), and explain *one example* of basic transformation of coordinates which can be implemented as matrix multiplication *only* if we use homogeneous coordinates.

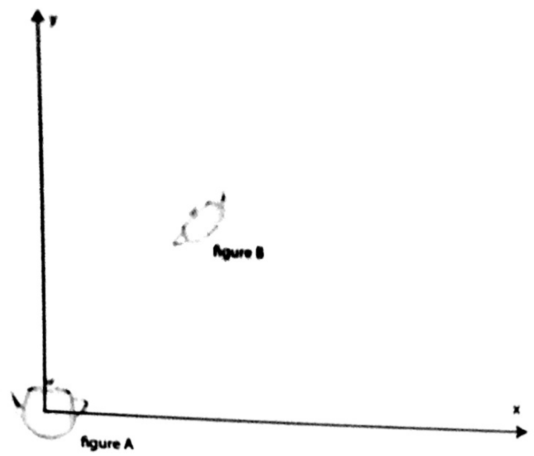
2. Look at the figure below.

a) Describe, using the matrix notation of transformations, the types, individual values, and order of applying transformation matrices that need to be applied to figure A in order to obtain the figure B.

b) What is the minimal number of transformations needed to obtain figure B from figure A?

c) If and where needed, you can make annotations on the figure to indicate values of coordinates, angles, and sizes. Also, explain if the needed transformations should be 2D or 3D ones.

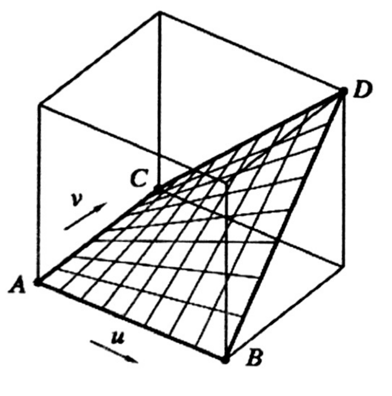
d) Given your sequence of transformations, draw an image (similar to the figure below) indicating what would be the effect if you applied the sequence you propose in *inverse order* on figure A.



3. Three-dimensional objects can be modeled using a variety of techniques, such as polygonal meshes, implicit surfaces, and volumetric voxel representations.

a) Give, for such type of technique, one advantage and one disadvantage. For advantages, think of what the respective technique can model *easily*, or *computationally efficient*. For disadvantages, think of types of 3D shapes the respective technique *cannot model* or computational inefficiencies.

b) Consider the 3D quadrilateral surface in the figure below:



a) Give a parametric description of this surface, as a function F(u,v) = (x(u,v), y(u,v), z(u,v)).

c) Suppose that the four points are located on the vertices of a unit cube, as follows: *A*=(0, 0, 0), *B*=(1, 0, 0), *C*=(0, 1, 0), and *D*=(1, 1, 1). What are the 3D coordinates and parametric of the quadrilateral's center point? Show how you derive these values using the parametric surface description indicated at point (a).

d) Derive the expression of the normal vector in parametric coordinates, **n**(u,v) = (n_x(u,v), n_y(u,v), n_z(u,v)). Hint: A simple way to do this, is to first derive parametric expressions for the *tangent* vectors on the surface in the two parametric directions u and v. Next, compute the normal from these directions using a cross-product.

4. Different techniques are used to create correct depth impressions of 3D rendered scenes: painter's algorithm, backface culling, and depth buffering.

a) Explain why the painter's algorithm is computationally inefficient (as compared to depth buffering) for large 3D scenes consisting of many polygonal primitives. Considering a scene composed by N such primitives, give and explain the complexity orders of the painter's algorithm and depth buffering.

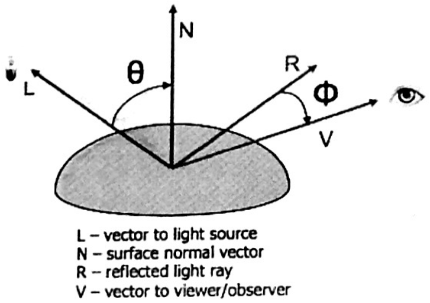
b) Consider a low-cost device where memory is expensive. Here, we shall use a depth-buffer with a *fixed point implementation* consisting of a byte (8 bit) integer per pixel. Sketch a simple 3D polygon configuration and camera parameters for which this implementation will give incorrect depth orderings of the polygons (similar to the drawing in exercise 1a).

c) Consider the previous question: Describe, in detail, what are the incorrect effects that can occur given the low resolution of the depth buffer. Hint: Think of the incorrect effects that can occur when using the painter's algorithm – and consider which of these, and also which other effects, will occur in the case of depth buffering.

d) We want to increase the quality of the hidden surface removal computed using our low-accuracy depth buffer described above. If no additional bits per pixel can be stored, describe other mechanisms that we can use to achieve such an increase of accuracy. Hint: Think of the way in which we can most accurately represent a *range* of depth values on a *fixed-point* 8-bit representation.

5. Consider the Phong lighting model given by the expression below:

$$I_{Phong} = L_a k_a + \frac{1}{a + bd + cd^2}\left(L_d k_d (L \cdot N) + L_s k_s (R \cdot V)^e\right)$$



Here, L_a, L_s, and L_d are the intensities of the ambient, specular, and diffuse lighting components; and k_a, k_d, and k_s are the color components of the above-mentioned lighting components. The equation is applied three times (to compute the R, G, and B components of the final lighting I_{Phong}).

a) Consider a blue shape which is illuminated by red light. What is the color of the diffuse component of the lighting? What is the color of the specular highlight of the light? Support your answers by the Phong illumination formula and the right parameter values used for it.

b) Consider an application in which we have a *given* 3D shape (which we cannot change). However, in this application, we would like to control the *shape* of the specular highlight obtained by a Phong illumination model. For example, we would like to obtain highlights of specific shapes, such as round, square, triangle, or star-shaped. Describe *two* different techniques which we could use to achieve this effect. Hint: You are allowed to use more than a single point-

light, or lights which have a directional distribution; think also of using other techniques which create `patterns' on a given 3D surface.

c) Explain (briefly) what is the difference between the Phong illumination model and Phong shading.

d) Consider three widely-used shading techniques: flat, Gouraud, and Phong. Explain which is the main advantage and disadvantage of each technique. Next, consider a scene with N polygons which is to be drawn on a screen of P pixels. Give, for each technique, the *worst-case complexity* as a function of N and P. Hint: To derive these complexities, think of the rendering algorithm that has to rasterize each polygon, perform the required interpolations, and finally render these polygons on the raster screen.