EXAM IN

COMPUTER GRAPHICS

TSBK07

(TEN1)

Time:	20st of August, 2020, 8-12 (13)
Room:	Distance
Teacher:	Ingemar Ragnemalm, available by E-mail during exam.
Allowed help:	N/A. Copying text from other sources is not allowed.
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.

Upload answers to https://twokinds.se/liu/fileuploadaugust/

In case of problems, you can also mail answers to ingemar.ragnemalm@liu.se

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

1. OpenGL programming

a) You have a variable in your host OpenGL program, *GLfloat time*, which holds the time. This value is needed in your vertex shader to control an animation. Describe how the host OpenGL program sends the variable to a shader program. Clarify how variables are identified. A code-like example is preferred.

(2p)

b) In our first OpenGL examples, models were drawn with glDrawArrays(). Motivate why glDrawElements() can be a better choice.

(2p)

2. Transformations

a) You want to place a camera at (1, 0, 0) looking at origin. What transformations (translation, rotation) are needed to do this? The order of matrices, with contents, should be included.

(3p)

b) You are making a project based on the terrain rendering lab, and want to have the camera being aligned to the ground as well as models. Using a vector along the ground plus the camera position, you can get a "lookat" point and create the "lookat" matrix from the vector library. Explain how you can use this kind of information for *placing models*. Your answer should relate to the coordinate systems involved.

(4p)

3. Light, shading and ray-tracing

a) A scene is all black except for two surfaces, numbered 1 and 2. Number 1 is a large white lamp. Number 2 is a smaller light grey object.

Surface 1 is a light source with intensity = 1. It also reflects light as a 100% white surface (reflecting all incoming light). Surface 2 is light grey, reflecting 75% of all incoming light.

Since surface 2 is so small, only one third of the total light emitted from surface 1 hits surface 2. Surface 1, however, is larger, so 2/3 of the light emitted from surface 2 hits surface 1.

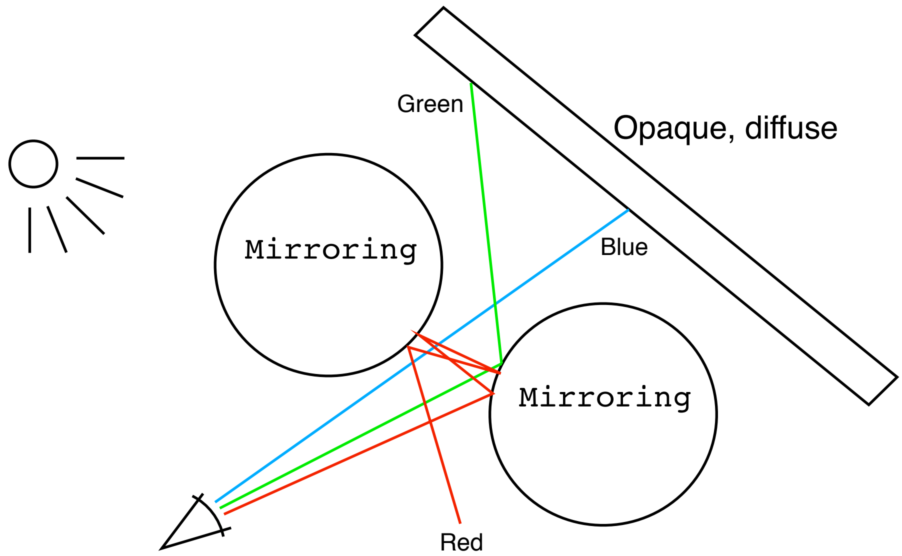
The radiosity equation is:

$$B_k = E_k + R_k \sum_{j=1}^{n} B_j F_{jk}$$

Express the light exchange between the two surfaces above using the radiosity equation. How much light does surface 2 emit?

(4p)

b) Two mirroring spheres, a diffuse non-transparent surface and the camera are placed as in the figure. An image is rendered using ray-tracing. Outline how the ray-tracer works for this case. Three rays are shown as red, green and blue. Are there any special considerations for any of them?



(Note: The rays are named in case the colors are lost in printing.)

(3p)

4. Surface detail

a) MIP mapping does not always give a good result; there is a case when even the best settings does not help. What is this case?

(2p)

b) What mapping techniques are most suitable for describing material variations in a surface? Motivate your answer.

(2p)

c) You want to make a variant of the mappings that we have seen in the course, a two-sided planar mapping, making a linear/planar mapping different for each side of the model. Describe how such a mapping would work.

(2p)

5. Curve generation

a) A curve, shown in the figure below, is defined by two circle segments. In the intersection between the segments (marked with black spot), what continuity critera are fulfilled, granted that the angular velocity along each segment are the same?

You may use either a mathematical solution or geometrical reasoning.

(3p)

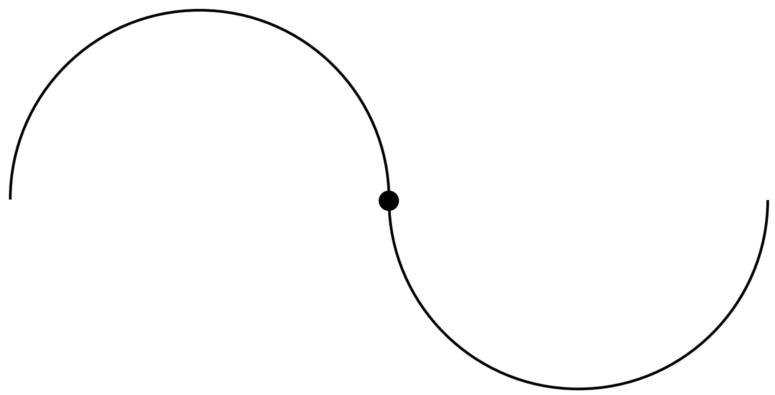


Figure for 5a

b) Granted that the bottleneck of the operation is memory access (both to RAM and registers), compare the DDA algorithm to Bresenham's algorithm.

(2p)

6. Miscellaneous

a) Compare supersampling to multisampling. In what areas of the picture are aliasing is suppressed equally for both methods and in which one is one better? Which one, and why?

(3p)

b) When I describe the Diamond-square algorithm, I claim that a bigger filter (i.e. cubic spline) than a linear interpolation (average between neighbors) will produce a better result. Draft an example using figures with a height curve that proves this statement, where we can see why the simple linear interpolation will be inferior and causes undesirable artifacts. Hint: Five points with a peak in the middle will suffice.

(3p)

7. Collision detection and animation

a) There are two fundamentally different ways to perform *picking*. Describe the two. Suggest situations when one would be preferable over the other and vice versa.

(2p)

b) Outline the principles for an efficient collision detection system based on broad and narrow phase tests. There are really three phases, which ones? List, by name and a brief explanatory comment with a figure, a typical solution for each phase.

(3p)

8. Visible surface detection and large worlds

a) A student describes frustum culling like this:

*The frustum is set up as six planes. In order to optimize it, the top and bottom planes are removed. The remaining planes are tested against spherical bounding shapes of objects in the scene. The spheres are transformed to view coordinates where the tests against the frustum are performed, using, for each plane and sphere, a point **a** in the frustum plane, the plane's normalized normal vector **n** pointing outwards, the sphere position **c** and its radius r. Then, the object is drawn if*

$$\mathbf{a} \bullet \mathbf{n} > (\mathbf{c} + \mathbf{n} \cdot r)$$

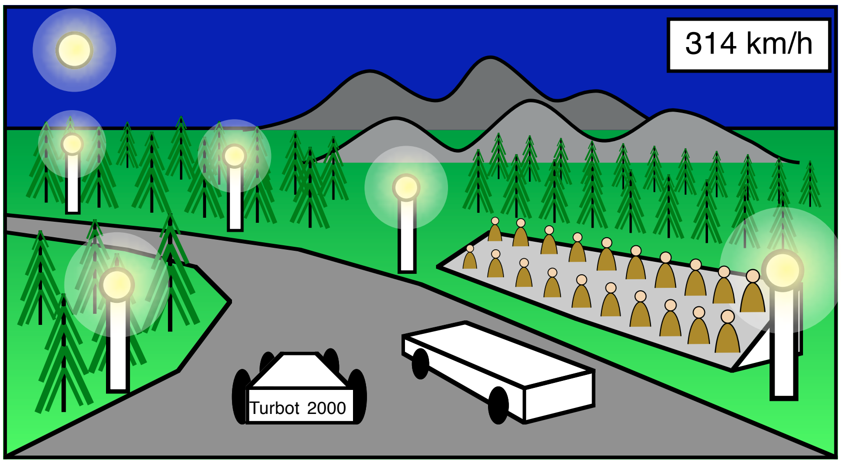
is true for any plane.

The examiner is not satisfied with this answer. Why? Correct the errors and/or bad reasoning.

(4p)

b) Linus and Linnea are creating a particularly pleasing project, a car racing game, illustrated by the figure below. The scene includes the cars, the racing track, trees, sky, grass, audience, distant mountains etc. All surfaces are texture mapped and lit. Over a dozen cars may participate in the race and there are hundreds of people in the audience. It takes place at night with a large number of light sources (>100) along the track. Every car also has two lights in front and back.

The game was too slow even on high-end hardware, so they tried optimizing it with frustum culling, using a sphere around each object, which improved the performance but not quite enough.



Suggest *three* remedies that you would say are the most likely to improve rendering speed without significantly ruining rendering quality for this application. Motivate your answer. Full score requires a brief description of how the remedy works, and to what part of the scene.

(6p)