

Graphics 2008/2009

T2

Final exam

Thu, Nov 06, 2008, 14:00–16:00

- **Do not open this exam until instructed to do so.**
- **Read the instructions on this page carefully.**
- You may write your answers in English, Dutch, or German.
Use a pen, not a pencil. Avoid usage of the color red.
- You may *not* use books, notes, or any electronic equipment
(including your cellphone, even if you just want to use it as a clock).
- Please put your student ID on the table so we can walk around and check it during the exam.
You also have to show it to the instructor when you turn your exam in before leaving the room.
- Write down your name and student number on every paper you want to turn in.
Additional paper is provided by us. You are not allowed to use your own paper.
- The exam should be doable in less than 1.5 hours. You have max. 2 hours to work on the questions.
If you finish early, you may hand in your work and leave, except for the first half hour of the exam.
- The exam consists of 7 problems printed on 5 pages (including this one).
It is your responsibility to check if you have a complete printout.
If you have the impression that anything is missing, let us know.
- The maximum number of points you can score is 18.
You need at least 17 points to get the best possible grade.

Good luck!

Problem 1: Perspective projection

Subproblem 1.1 [1.5 pt] Given an arbitrary camera position, we want to display the objects of a 3D model in a 2D image using perspective projection. List all the steps that are involved in the procedure we discussed in the lecture. (Note: Requested are the steps of the initial graphics pipeline (part I) we discussed. Just name the involved transformations. It is *not* necessary to provide detailed descriptions, equations, etc.)

Subproblem 1.2 [1.5 pt] In the lecture, we used a matrix M_o in the procedure for calculating the perspective projection of a 3D model onto a 2D image. It was defined by the following product of three matrices (note: l, r, t, b, n, f specify the left, right, top, bottom, near, and far plane of the orthographic view volume, respectively, and $m \times n$ is the size of the projected image):

$$M_o = \begin{pmatrix} \frac{m}{2} & 0 & 0 & \frac{m}{2} - \frac{1}{2} \\ 0 & \frac{n}{2} & 0 & \frac{n}{2} - \frac{1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{2}{r-l} & 0 & 0 & 0 \\ 0 & \frac{2}{t-b} & 0 & 0 \\ 0 & 0 & \frac{2}{n-f} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & -\frac{l+r}{2} \\ 0 & 1 & 0 & -\frac{b+t}{2} \\ 0 & 0 & 1 & -\frac{n+f}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Which step(s) that you listed in subproblem 1.1 are done by M_o ?
- Shortly describe what each of the three matrices does.

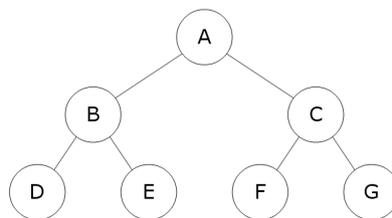
Problem 2: Hidden surface elimination

Subproblem 2.1 [0.5 pt] Which of the following statements is correct? (no explanation required)

- The calculation of the BSP and the drawing order are both independent from the camera position.
- Only the calculation of the BSP is independent from the camera position.
- Only the drawing order is independent from the camera position.
- The calculation of the BSP and the drawing order are both dependent from the camera position.

Subproblem 2.2 [1 pt] Assume a camera position e and the implicit representation of a plane in \mathbb{R}^3 . Let's further assume that the normal vector from the implicit equation of the plane is on the front side of the plane. How can we easily specify if the camera is located in front of or behind the plane?

Subproblem 2.3 [1.5 pt] Assume a scene containing seven triangles A, B, C, D, E, F, and G, and the following related BSP tree:



- Shortly explain how we can use such a BSP tree to get a correct drawing order of the triangles.
- Write down the correct drawing order of the seven triangles with respect to a camera position e which is on the positive side of the planes defined by triangles A, B, C, and on the negative side of the planes defined by triangles D, E, F, G.

Problem 3: Triangle rasterization

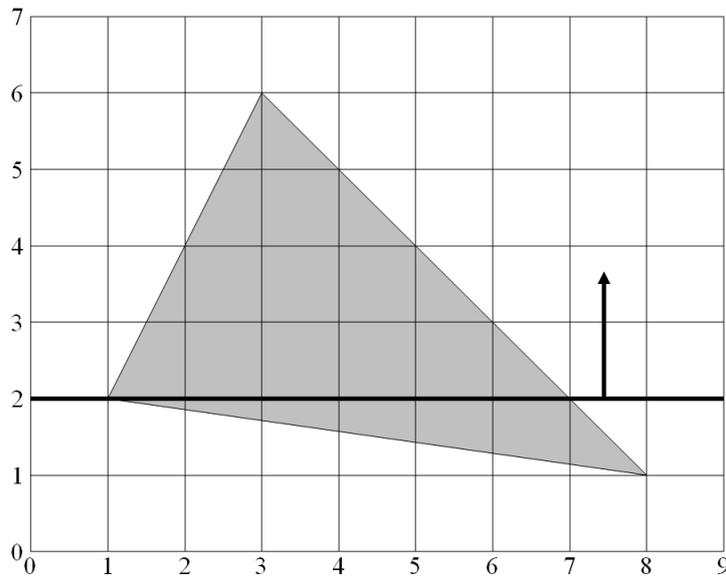
Subproblem 3.1 [1 pt] In the lecture, we used two data structures in the algorithm for triangle rasterization via scan-line conversion: The *edge table* and the *active edge table*. The entries of the edge table are called edge records. Assume the following example of an edge record:

$$2 : (9, 6, -1/2)$$

Explain what this means and describe each of the three values specified by this edge record.

Subproblem 3.2 [2 pt] Look at the image below. Assume that the scan-line is horizontal and moves vertically from the bottom of the image to the top as illustrated by the black line and the associated arrow.

- (a) Give the values for the edge table that are stored for scan-line number 1.
- (b) Give the values for the active edge table that are stored for scan-line number 3.



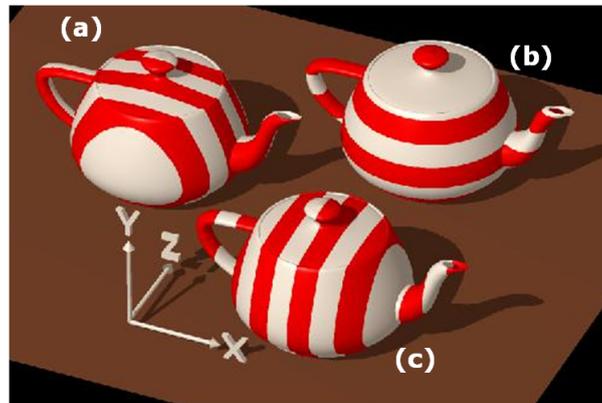
Problem 4: Ray tracing

Subproblem 4.1 [1 pt] In the lecture, we discussed how we can create new objects based on a given one by *instancing*. Suppose we have an object O , a matrix M which does a basic transformation, and a ray r . Now we want to create a new object O' by applying the matrix M to the object O . Assume that intersections with the object O are much easier to calculate than with object O' (e.g. because it is compiled of circles whereas O' is compiled of ellipses). Explain how we can easily compute the intersection p of r and O' .

Subproblem 4.2 [1 pt] Assume we have a 2D image containing many triangles. Explain how you can use the Octree approach to separate the space in a way that each of the resulting cells contains only two triangles at the most. (Note: for simplification, you can assume that the triangles in the image are distributed in a way that there is no need to split them when new cells are created.)

Problem 5: Texture mapping

Subproblem 5.1 [1 pt] In the following image, different procedures have been used to create the stripe textures for each of the three teapots:



(a) Which texture was created with the following procedure?

```
stripe(point ( $x_p, y_p, z_p$ )) {  
  if ( $\sin x_p > 0$ )  
    return color0;  
  else  
    return color1;  
}
```

(b) How do you have to change it to create the texture on the other two teapots?

Subproblem 5.2 [1.5 pt]

- Give a short description of bump mapping.
- Give a short description of displacement mapping.
- In which situation(s) should one prefer bump mapping over displacement mapping?

Problem 6: Radiosity

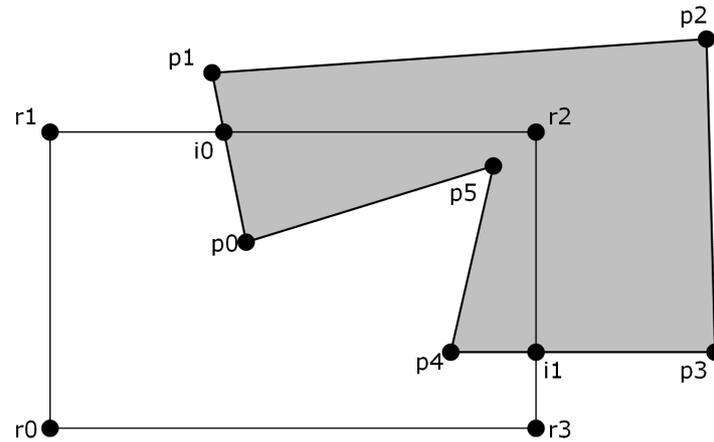
Subproblem 6.1 [0.5 pt] Assume you have to create an animated movie with a walk-through of the interior of some buildings. Give a short explanation why you might use radiosity to compute diffuse global illumination instead of, for example, ray tracing and ambient lighting.

Subproblem 6.2 [1.5 pt] Explain the meaning of the following formula, which is used to calculate the radiosity B_i of a patch A_i :

$$B_i = E_i + \rho_i \sum_j B_j F_{ij}$$

Problem 7: Clipping

Subproblem 7.1 [1 pt] We want to clip the gray polygon against the clipping area represented by the rectangle in the image below. Construct the graph used by the Weiler-Atherton algorithm.



Subproblem 7.2 [1.5 pt] Give a general explanation about how the graph from the Weiler-Atherton algorithm is used to determine the clipped polygons. Then use the graph you have drawn in the previous subproblem to create the resulting polygons.