1. Ear cutting

A polygon is defined by vertices [0, 0], [1, 2], [0, 4], [4, 4], [3, 2], and [4, 0]. What are the ears that are cut by the ear-cutting algorithm?

2. Clipping line segments

For clipping a line segment connecting $[x_1, y_1]$ and $[x_2, y_2]$, how can an intersection point with a borderline $x=x_{max}$ be calculated?

3. Viewport transformation

How can normalized device coordinates x_{ndc} and y_{ndc} be mapped onto pixel coordinates x_{pix} and y_{pix} if the bottom-left corner of the viewport is $[v_x, v_y]$, and the width and height of the viewport are defined by v_{width}, and v_{height}, respectively?

4. Rasterization

What are the coordinates of the pixels that are painted by the rasterization of a line segment connecting points [0, 0] and [3, 2]?

5. Texture mapping

The vertices of a triangle are defined as [-1, -1], [1, -1], [-1, 1]. The corresponding texture coordinates are [0, 0], [1, 0], and [0, 1], respectively. What are the texture coordinates that correspond to the point [0, 0]?

6. Graphics hardware

What are the functionalities of the fixed pipeline between the vertex shader and the fragment shader?

7. Ray tracing

Using ray tracing, how can an intersection point be calculated between a ray defined by $\mathbf{r}(t) = \mathbf{o} + \mathbf{d} \cdot t$ and a plane defined by $(\mathbf{r} - \mathbf{p}) \cdot \mathbf{n} = 0$?

8. Ray tracing

Using ray tracing, how can it be determined that an intersection point between a ray and a plane of a triangle is inside the given triangle?

9. Ray tracing

Using ray tracing, how can it be determined whether a light source contributes to the illumination of a surface point?

10. Ray tracing

Using recursive ray tracing, how can an infinite recursion be avoided?