1. What abstract light models have we studied? What real light sources can be modeled with each type?
2. Regarding light–matter interactions, what type of material models did we study?
3. What is the BRDF? What parameters are used to define it? (Include a figure.)
4. Explain how the diffuse illumination defined by Lambert's cosine law is calculated.
5. Characterize the appearance of a diffuse surface.
6. In abstract light source calculations, what is the angle between the surface normal and the view direction used for?
7. Describe specular reflection and the Phong model. Write down the BRDF and explain the coefficients appearing in the formula.
8. Describe specular reflection and the Blinn–Phong model. Write down the BRDF and explain the coefficients appearing in the formula.
9. Characterize the appearance of a surface with specular properties.
10. How is the Phong model of specular reflection calculated? Which built-in shader functions are used in its implementation, and what do they do?
11. For different kinds of abstract light sources, how is the „toLight" vector calculated? What does this vector represent?
12. When using ambient, diffuse, and specular models, what surface properties are used in the calculations?
13. A spot light is located at $[4, 8, 0]$ and points in the direction $[-1, 0, 0]$. The inner light circle has an opening angle of $60°$, and the outer light circle has an opening angle of $90°$. Among the following points, which lie inside the inner light circle, which lie between the inner and outer light circles, and which lie outside the illuminated region? $[0, 0, 0]$, $[0, 4, 0]$, $[8, 3, 8]$, $[4, 10, 2]$, $[-4, 4, 0]$, $[-4, 5, 0]$.
14. Before computing the lighting models in the fragment shader, how do we obtain the positions and normals in world coordinates? Why do we need to obtain them in this way?
15. What does sampling (filtering) mean?
16. When sampling textures, what problem occurs during magnification? What solutions do we use for it?
17. When sampling textures, what problem occurs during minifying? What solutions do we use for it?
18. What is a procedural texture? How is it sampled?
19. What problems can be solved using textures whose data is not interpreted as colors?
20. How is parallelization carried out on the GPU in the graphics pipeline?
21. How is the GPU structured? What are the basic principles of the SIMD architecture?
22. How is branching executed on a SIMD architecture?
23. How can idle execution (stalls) on the GPU be avoided, for example when waiting for a texture fetch?