

## Aufgabe 1: Raytracing

### Teilaufgabe 1a

TODO

### Teilaufgabe 1b

TODO

### Teilaufgabe 1c

TODO

### Teilaufgabe 1d

1. **Ray generation:** Erzeuge Sichtstrahlen durch jeden Pixel.
2. **Ray intersection:** Schnittberechnung; also: Finde Objekt welches den Strahl schneidet und am nächsten zur Kamera ist.
3. **Shading:** Schattierung / Beleuchtungsberechnung.

## Aufgabe 2: Farben

### Teilaufgabe 2a

*Wie nennt man die Funktionen, mit denen man Tristimulus-Werte zu einem gegebenen Spektrum berechnen kann?*

Color Matching Funktionen?

### Teilaufgabe 2b

- *Es gibt eine lineare Abbildung zwischen den Farbräumen XYZ und xyY.*  
→ TODO
- *Es gibt eine lineare Abbildung zwischen den Farbräumen RGB und XYZ.*  
→ TODO
- *Die subjektiv empfundene Stärke von Sinneseindrücken ist proportional zur Intensität des physikalischen Reizes.*  
→ TODO

### **Teilaufgabe 2c**

*Welche Information beinhalten die  $x$ - und  $y$ -Komponenten einer Farbdarstellung im CIE- $xyY$ -Farbraum zusammengekommen?* TODO

### **Teilaufgabe 2d**

TODO

## **Aufgabe 3: Homogene Koordinaten**

### **Teilaufgabe 3a**

TODO

### **Teilaufgabe 3b**

TODO

### **Teilaufgabe 3c**

TODO

## **Aufgabe 4: Transformationen**

### **Teilaufgabe 4a**

TODO

### **Teilaufgabe 4b**

TODO

## **Aufgabe 5: Beschleunigungsstrukturen und Hüllkörper**

### **Teilaufgabe 5a**

TODO

### **Teilaufgabe 5b**

TODO

### **Teilaufgabe 5c**

TODO

## **Aufgabe 6: Texturen**

### **Teilaufgabe 6a**

TODO

### **Teilaufgabe 6b**

TODO

## **Aufgabe 7: Beleuchtung**

### **Teilaufgabe 7a**

TODO

### **Teilaufgabe 7b**

TODO

### **Teilaufgabe 7c**

TODO

### **Teilaufgabe 7d**

TODO

### **Teilaufgabe 7e**

TODO

### **Teilaufgabe 7f**

TODO

## **Aufgabe 8: Partikeleffekte und OpenGL-Blending**

### **Teilaufgabe 8a**

TODO

### **Teilaufgabe 8b**

TODO

### **Teilaufgabe 8c**

TODO

### **Teilaufgabe 8d**

TODO

## **Aufgabe 9: OpenGL**

TODO

## Aufgabe 10: Reflexionen in OpenGL

```

1 // ...
2 uniform float h; // Y-Koordinate der Wasserebene im World Space
3 in vec3 P; // World-Space-Position des aktuellen Fragments
4 // ...
5
6 void main(void)
7 {
8     // Implementieren Sie hier das Clipping:
9     if (P.y < h) {
10         discard;
11     }
12
13     // Es folgt das spezifische Shading des jeweiligen Objekts
14     // ...
15 }
```

## Aufgabe 11: GLSL-Hatching

### Teilaufgabe 11a

$$\left(1 - \frac{x - x_1}{x_2 - x_1}\right) y_1 + \frac{x - x_1}{x_2 - x_1} y_2$$

oder

$$y_1 + \frac{y_2 - y_1}{x_2 - x_1} (x - x_1)$$

### Teilaufgabe 11b

```

1 uniform sampler2D dunkel; // dunkle Textur
2 uniform sampler2D mittel; // mittlere Textur
3 uniform sampler2D hell; // helle Textur
4 in vec2 tex_coord; // Texturkoordinate
5
6 vec4 get_hatched_color(float h) // Helligkeit h liegt in [0, 1].
7 {
8     vec4 t_d = texture(dunkel, tex_coord);
```

```
9     vec4 t_m = texture(mittel, tex_coord);
10    vec4 t_h = texture(hell, tex_coord);
11
12    float weight_d = clamp(1. - 4. * (h - .25), 0., 1.);
13    float weight_m = clamp(1 - 4. * abs(h - .5), 0., 1.);
14    float weight_h = clamp(4. * (h - .5), 0., 1.);
15
16    return weight_d * t_d + weight_m * t_m + weight_h * t_h;
17 }
```

---

## Aufgabe 12: Bézierkurven

### Teilaufgabe 12a

TODO

### Teilaufgabe 12b

TODO

### Teilaufgabe 12c

1. Ja
2. Ja
3. Nein, da die Kurve nicht innerhalb der konvexen Hülle der Kontrollpunkte ist.