

Hauptklausur Computergrafik WS 2022/2023

6.3.2023

Name	
Matrikelnummer	

Beachten Sie:

- Die Klausur umfasst 23 Seiten (12 Blätter) mit 10 Aufgaben.
- Es sind **keine Hilfsmittel** zugelassen.
- Sie haben **90 Minuten** Bearbeitungszeit.
- Schreiben Sie Ihre Matrikelnummer oben auf jedes bearbeitete Aufgabenblatt.
- Schreiben Sie Ihre Lösungen auf die Aufgabenblätter. Bei Bedarf können Sie weiteres Papier anfordern.
- Streichen Sie nicht zu bewertende Lösungen durch.
- Wir akzeptieren auch englische Antworten.

Aufgabe	1	2	3	4	5	6	7	8	9	10	Gesamt
Erreichte Punkte											
Erreichbare Punkte	15	20	10	23	16	11	21	38	15	11	180

Note



Aufgabe 1: Farbe und Perzeption (15 Punkte)

- a) In Abbildung 1 sind Bildkorrekturen dargestellt, die Pixelhelligkeiten verändern. Dabei bezeichne I_{in} die eingehende und I_{out} die ausgehende Helligkeit.

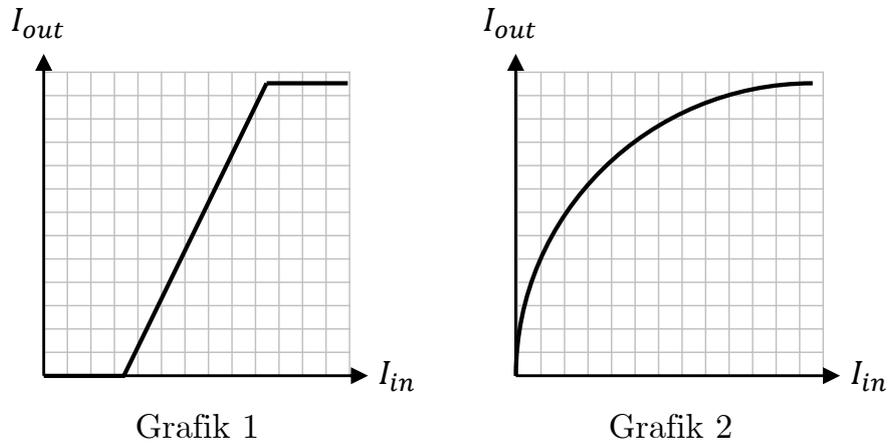


Abbildung 1: Zwei unterschiedliche Bildkorrekturen.



- i) Um welche Operation handelt es sich in den beiden Grafiken? Begründen Sie Ihre Antwort stichpunktartig! (3 Punkte)

Grafik 1:

Grafik 2:



- ii) Begründen Sie, welche dieser beiden Korrekturen verwendet wird, um das Display-Verhalten eines Monitors auszugleichen und warum dies notwendig ist! Zeichnen Sie das Display-Verhalten in die entsprechende Grafik ein! (3 Punkte)

Ersatzgrafik:

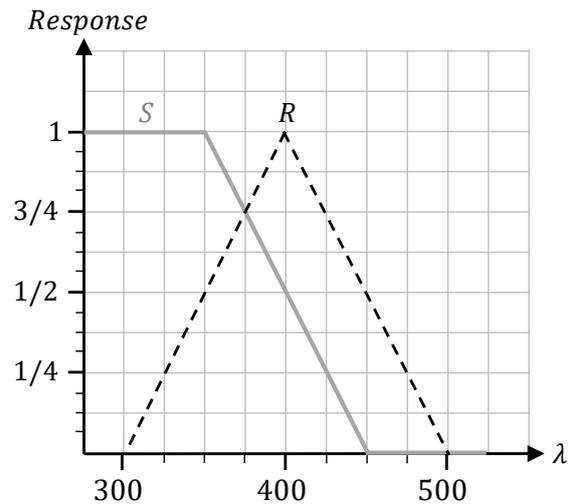
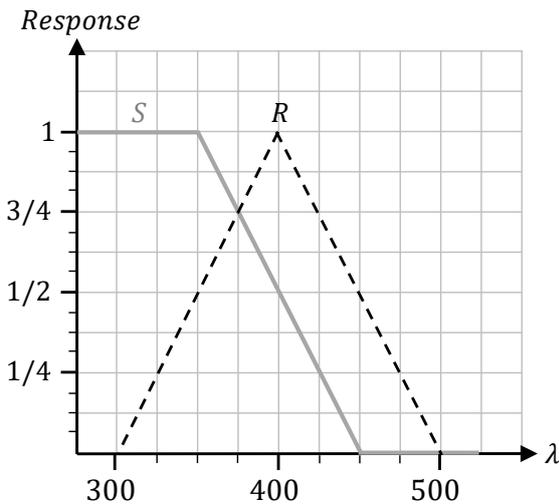


Abbildung 2: Empfindlichkeitskurve eines fiktiven Monochromaten.

b) Im Gegensatz zum Menschen sind Robben Monochromaten. In Abbildung 2 ist die idealisierte Empfindlichkeitskurve R eines solchen Monochromaten eingetragen (gestrichelte Linie), der einem Stimulus S (durchgezogene Linie) ausgesetzt wird.

i) Geben Sie an, wie die perzeptuelle Antwort r des Monochromaten bestimmt wird und skizzieren Sie diese in Abbildung 2. Zur Korrektur können Sie die zweite Grafik verwenden. *Kennzeichnen Sie eindeutig, welche Zeichnung bewertet werden soll!* **(5 Punkte)**

Hinweis: Beim Skizzieren hilft es Ihnen, wenn Sie u.a. die Wellenlängen $\lambda = 375$ und $\lambda = 425$ betrachten.

$r =$

ii) Tragen Sie einen anderen Stimulus $S' \neq S$ in Abbildung 2 ein, bei dem die perzeptuelle Antwort des Monochromaten die gleiche wie in Teilaufgabe i) ist! **(2 Punkte)**

iii) Der Monochromat kann einen Helligkeitsunterschied von 0.5% wahrnehmen. Wie berechnet man, wie viele solcher Schritte zu einer Dekade Dynamikumfang führen? **(2 Punkte)**



Aufgabe 2: Whitted-Style Raytracing (20 Punkte)

- a) Die folgende Szene enthält eine Kamera K , eine Punktlichtquelle L , einen Raum R mit ideal-diffusen Oberflächen und einen Glasblock G im Vakuum. Der Raum und der Block sind mit dem Phong-Beleuchtungsmodell modelliert, der Raum mit $k_d > 0, k_a = k_s = k_r = k_t = 0$, der Block mit $k_a = k_d = k_s = 0, k_r > 0, k_t > 0$ und $\eta_{Block} > 1$.

Ersatzgrafik:

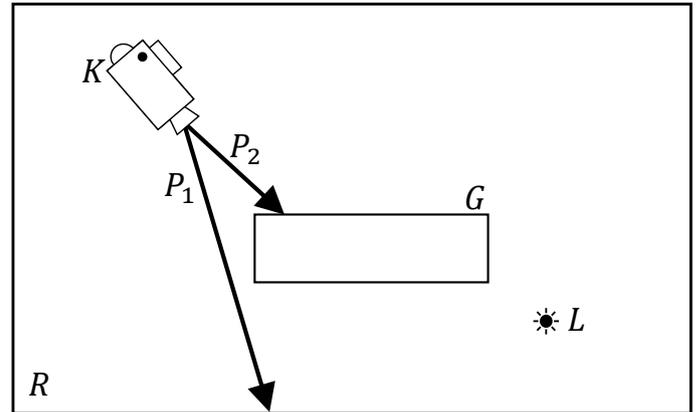
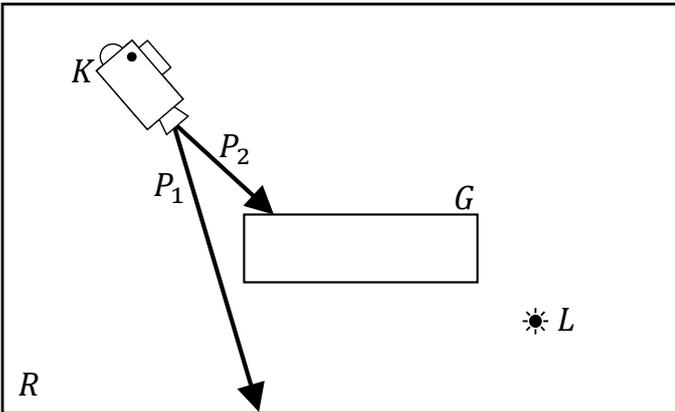


Abbildung 3: Szene mit Glasblock.

- i) Zeichnen Sie für die beiden Primärstrahlen P_1 und P_2 alle Sekundärstrahlen ein, die bei Whitted-Style Raytracing erzeugt werden und die einen **Beitrag zum Bild** liefern können! Zeichnen Sie nur Strahlen für maximal zwei weitere Rekursionen! Zur Korrektur können Sie die zweite Grafik in Abbildung 3 verwenden. *Kennzeichnen Sie eindeutig, welche Zeichnung bewertet werden soll!* (8 Punkte)



- ii) Wie wachsen die Strahlenbäume für P_1 und P_2 , wenn die maximale Rekursionstiefe verdoppelt wird? Begründen Sie Ihre Antworten! (4 Punkte)



Strahl P_1 :

Strahl P_2 :

b) Abbildung 4 zeigt links eine Fotografie einer echten Szene mit diffusen Wänden und einem spiegelnden Quader sowie einer Flächenlichtquelle hinter einem schwarzen Karton. Die Geometrie wurde exakt nachmodelliert und ein Bild mit einer Punktlichtquelle L soll mit Whitted-Style Raytracing berechnet werden. Markieren Sie zwei Stellen in der Skizze rechts in Abbildung 4, bei denen sich Fotografie und Bild unterscheiden werden, mit (1) und (2)! Begründen Sie kurz, weshalb sich diese Unterschiede ergeben! (8 Punkte)

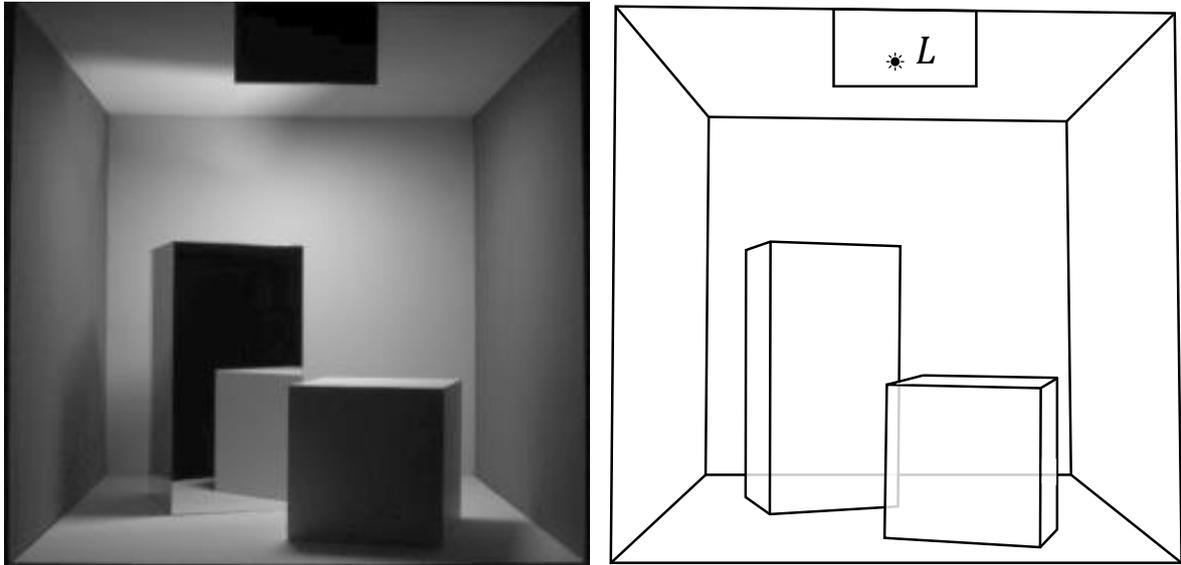


Abbildung 4: Fotografie und Modell einer Szene mit spiegelndem Quader.

(1):

(2):



Aufgabe 3: Shading und Beleuchtungsmodell (10 Punkte)



- a) Gegeben seien zwei Dreiecksnetze (Abbildung 5) mit glänzender Oberfläche. Bei welchem unterscheiden sich Gouraud und Phong Shading stärker? Begründen Sie! (4 Punkte)

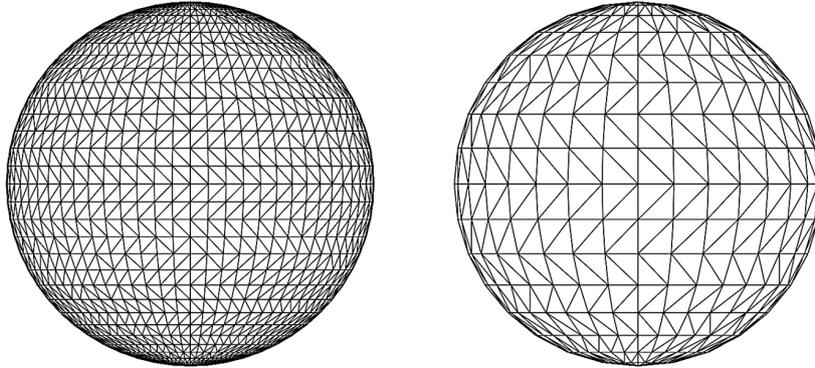


Abbildung 5: Zwei verschiedene Dreiecksnetze für eine Kugel.

Antwort:



- b) Seien \mathbf{v}_i die Eckpunkte eines Dreiecks und $\mathbf{n}_i, i \in \{0, 1, 2\}$ die korrespondierenden Normalen. Beschreiben Sie stichpunktartig, wie die Shading-Normale \mathbf{n} für einen Punkt \mathbf{p} auf dem Dreieck für Phong Shading bestimmt wird! (3 Punkte)



- c) Geben Sie an, wie die spekulare Komponente des Phong-Beleuchtungsmodells berechnet wird! Wie hängen die Unterschiede aus Teilaufgabe a) mit dem Phong-Exponenten zusammen? (3 Punkte)

Spekulare Komponente:

Unterschiede:

Aufgabe 4: Räumliche Datenstrukturen (23 Punkte)



a) Nennen Sie zwei Hüllkörper aus der Vorlesung *außer* der achsenparallelen Box (AABB)! (2 Punkte)



b) Ein Dreiecksnetz wird mit einer linearen Abbildung M transformiert. Es sei:

- B_1 die AABB des transformierten Dreiecksnetzes und
- B_2 die AABB der transformierten AABB des ursprünglichen Dreiecksnetzes.

Nennen Sie *je zwei* Klassen von Transformationen, für die immer gilt: (4 Punkte)



$B_1 = B_2$:

$B_1 \neq B_2$ (im Allgemeinen):

c) Für den in Abbildung 6 eingezeichneten Strahl soll mithilfe des angegebenen kD-Baums wie in der Vorlesung beschrieben der nächste Schnittpunkt mit der Szenengeometrie bestimmt werden. Geben Sie die Reihenfolge aller Schritte an, die dazu durchgeführt werden:

- Test des Halbraums (A - D),
- Schnitt mit Dreiecken (1 - 8)!



Beispiel für Angabe einer Lösung: A, 1, 2, B, 3, ... (5 Punkte)

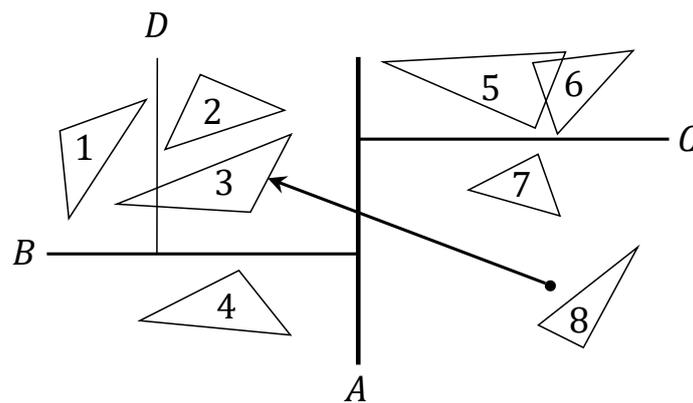


Abbildung 6: Strahlschnitt mit kD-Baum.

d) In dieser Aufgabe geht es um die Konstruktion von Hüllkörperhierarchien für Raytracing mit der Surface Area Heuristic (SAH).

i) Welche zwei Entscheidungen werden anhand der Kosten getroffen, die bei der Konstruktion mit der SAH berechnet werden? (**3 Punkte**)

ii) Welche Eigenschaften der Kindknoten fließen in die Berechnung der SAH ein? (**3 Punkte**)

iii) Wieso führen überlappende Kindknoten zu höheren Traversierungskosten? (**2 Punkte**)

iv) In Abbildung 7 sind zwei überlappende Kindknoten dargestellt. Zeichnen Sie *ein* weiteres Dreieck ein, sodass die eingezeichneten AABBs zu minimalen SAH Kosten führen würden!

Zur Korrektur können Sie die zweite Zeichnung verwenden. *Kennzeichnen Sie eindeutig, welche Zeichnung bewertet werden soll!* (2 Punkte)

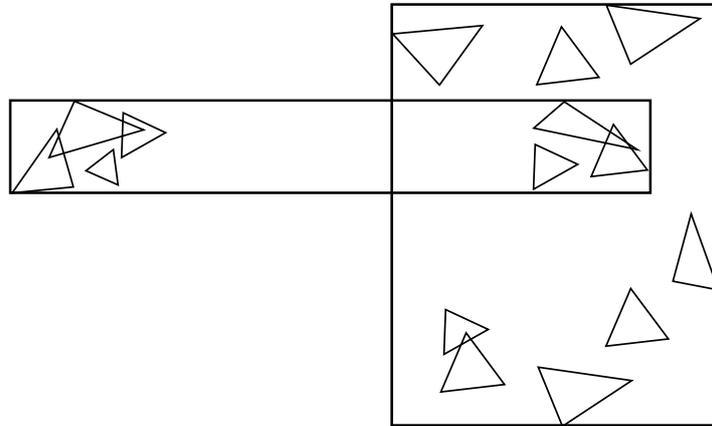
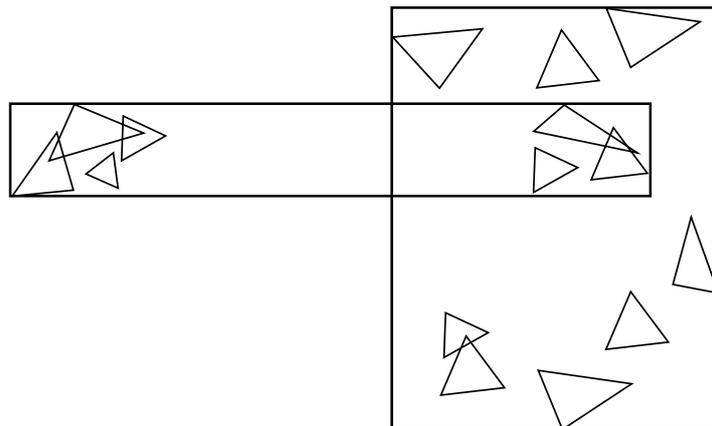


Abbildung 7: Zwei sich überlappende Kindknoten.

Ersatzgrafik:



v) Wie kann das in Aufgabenteil iv) auftretende Problem vermieden werden? (2 Punkte)

Aufgabe 5: Transformationen (16 Punkte)

In dieser Aufgabe betrachten Sie eine Kamerasteuerung. Gegeben sei hierfür die aktuelle homogene Kameramatrix \mathbf{V} . Die positive \mathbf{z} -Achse des Kamerakoordinatensystems zeigt entgegen der Blickrichtung.

- a) Geben Sie an, wie die Kameraposition \mathbf{x} und Blickrichtung \mathbf{w} aus der Kameramatrix \mathbf{V} in Weltkoordinaten bestimmt werden! (4 Punkte)

$$\mathbf{x} =$$

$$\mathbf{w} =$$

- b) Die Kamera soll in Kamerakoordinaten um d Längeneinheiten entlang der Blickrichtung verschoben werden. Geben Sie an, wie die neue Kameramatrix \mathbf{V}' aus \mathbf{V} berechnet wird! (5 Punkte)

$$\mathbf{V}' =$$

- c) Geben Sie an, wie eine neue Kameramatrix \mathbf{V}'' bestimmt wird, bei der die Blickrichtung um die \mathbf{z} -Achse in Weltkoordinaten mit dem Winkel ϕ rotiert wird. Geben Sie hierzu \mathbf{V}'' als Matrixprodukt von \mathbf{V} mit einer Translation \mathbf{T} und der Rotation an! Für die Rotation können Sie $\mathbf{R}_z(\phi)$ verwenden. (7 Punkte)

$$\mathbf{T} =$$

$$\mathbf{V}'' =$$



Aufgabe 6: Texturen (11 Punkte)

In dieser Aufgabe betrachten wir die Darstellung einer Wasseroberfläche A (Abbildung 8). Durch Brechung wird die parallele Fläche B darunter sichtbar. Mithilfe des gebrochenen Strahls wird die Texturkoordinaten ermittelt, um die Textur auf B auszulesen. Alle Betrachtungen in dieser Aufgabe finden in 2D mit 1D-Texturen statt. Die Texturkoordinaten u entspricht der x -Koordinate in Weltkoordinaten, d.h. $u = x$.

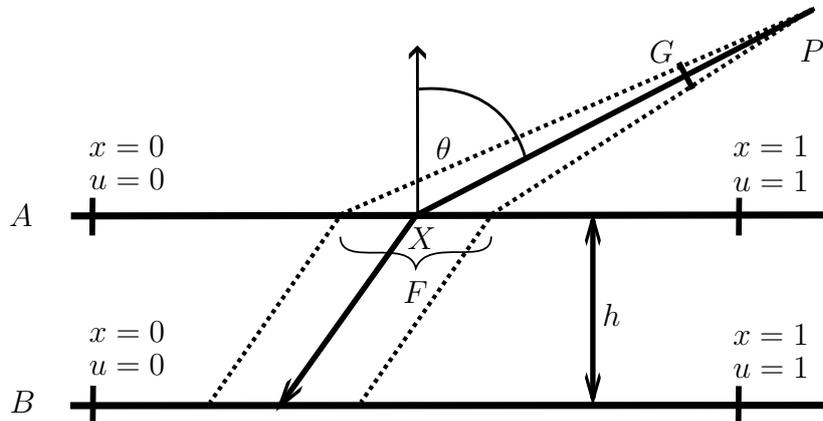
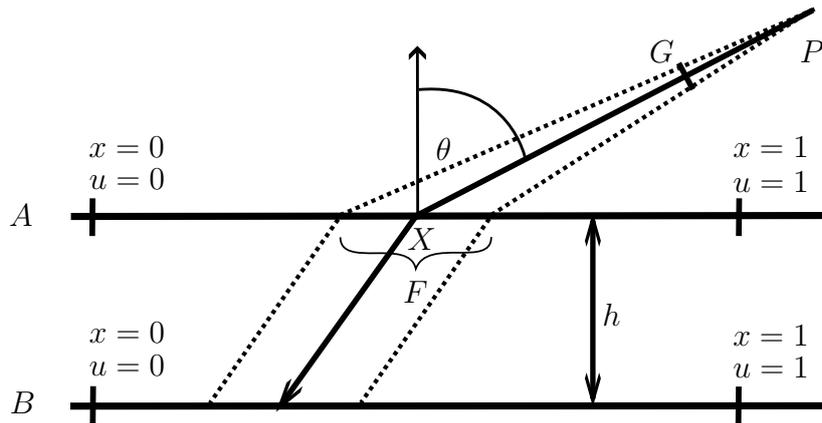


Abbildung 8: Wasseroberfläche A mit darunterliegender Fläche B .

- a) Der eingezeichnete Sichtstrahl schneidet A an der Stelle $x = 0.5$. Gegeben sei weiterhin die Richtung des gebrochenen Strahls $(-1, -1)$. Die Höhe des Wassers ist $h = 0.1$. Berechnen Sie die Texturkoordinaten des Schnittpunktes bei B ! (4 Punkte)





Duplikat von Abbildung 8.

- b) Berechnen Sie die Größe F des Pixel-Footprints auf A in Weltkoordinaten in Abhängigkeit von der Pixelgröße G , dem Abstand $d = \overline{PX}$ und θ ! (4 Punkte)

$$F =$$

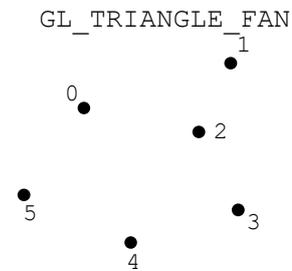
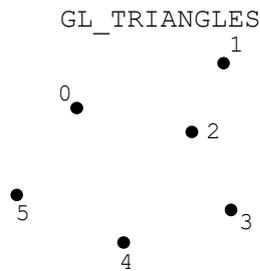
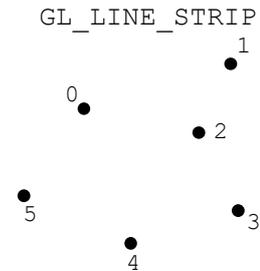
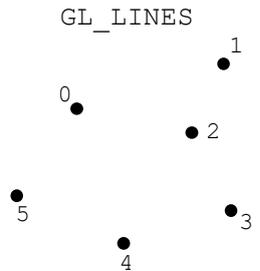
- c) Die Größe des Footprints in der Textur sei 16 Texel. Auf welche Mip-Map Stufe der Textur wird zugegriffen, wenn die Textur 512 Pixel groß ist? Die feinste Auflösungsstufe ist die Ebene 0 der Mip-Map Pyramide. (3 Punkte)

Aufgabe 7: OpenGL und Blending (21 Punkte)

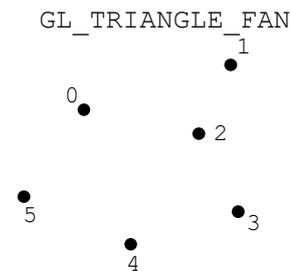
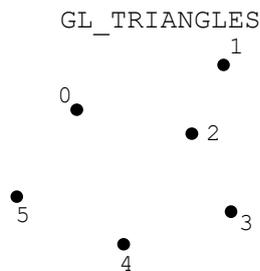
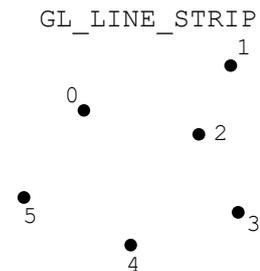
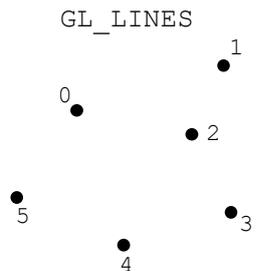


a) Nehmen Sie an, dass die folgenden Vertices in der gegebenen Reihenfolge von 0 bis 5 an die Pipeline übergeben werden.

i) Zeichnen Sie die resultierende Topologie zur jeweils gegebenen Konfiguration durch Verbinden der Punkte ein! Zur Korrektur können sie die Grafik darunter verwenden. *Kennzeichnen Sie eindeutig, welche Zeichnung bewertet werden soll!* (4 Punkte)



Ersatzgrafik:



ii) Muss die Primitive Assembly-Stufe vor oder nach dem Fragment-Shader ausgeführt werden? Begründen Sie Ihre Antwort kurz! (**3 Punkte**)

iii) An welcher Stelle der OpenGL-Pipeline werden die Pro-Fragment-Operationen ausgeführt? (**1 Punkt**)

iv) Der Stencil-Test ist eine der Pro-Fragment-Operationen. Was wird im Stencil-Puffer gespeichert? Wie kann dessen Inhalt beim Rendering verändert werden? (**4 Punkte**)

- b) Eine Wohnzimmer-Szene, bestehend aus opaken Dreiecken mit semi-transparenten Fenstern, soll mit OpenGL dargestellt werden. Für die opaken Dreiecke wurde die Pipeline bereits konfiguriert:

```
glEnable( GL_DEPTH_TEST );
glDepthMask( GL_TRUE );
glDisable( GL_BLEND );
```

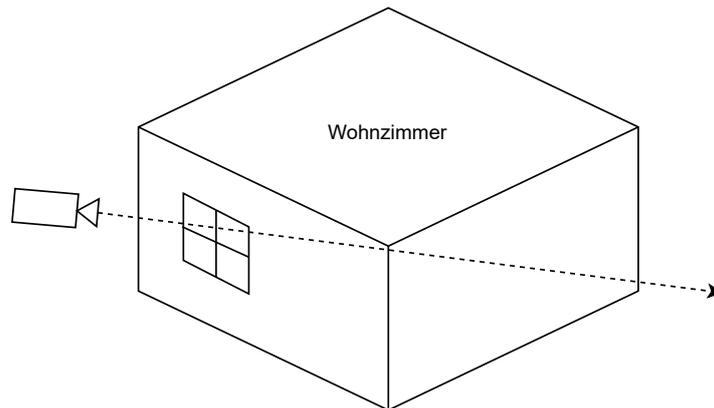


Abbildung 9: Wohnzimmer-Szene. Ein möglicher Sichtstrahl der Kamera ist gestrichelt eingezeichnet.

Das Zimmer wird mit einer Kamera von außerhalb rasterisiert (Abbildung 9). Dabei ist die Kamera auf eines der semi-transparenten Fenster gerichtet. Die opake Geometrie des Raumes wurde bereits rasterisiert und nun sollen die Fenster hinzugefügt werden. Konfigurieren Sie die OpenGL-Pipeline so, dass der Raum korrekt durch das semi-transparente Fenster erscheint! (4 Punkte)



```
glEnable (           ) ;
glBlendFunc (           ) ;
glBlendEquation (           ) ;
```

- c) Auf der gegenüberliegenden Seite des Raumes wird nun ein weiteres semi-transparentes Fenster platziert (Abbildung 10).

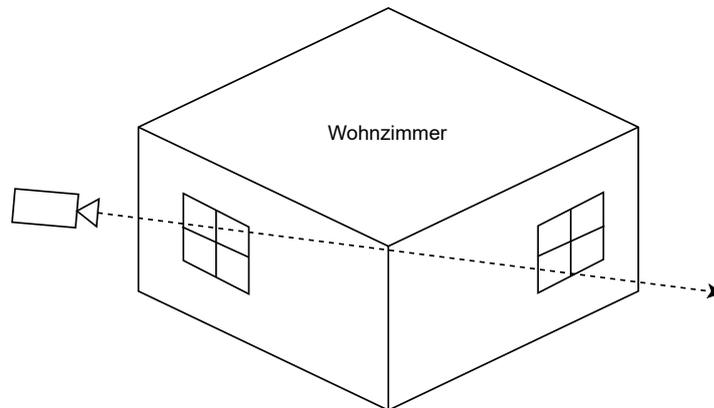


Abbildung 10: Wohnzimmer-Szene mit zusätzlichem Fenster. Ein möglicher Sichtstrahl der Kamera ist gestrichelt eingezeichnet.

- i) Erklären Sie, wie es passieren kann, dass das hintere Fenster nicht zum Bild beiträgt! (**3 Punkte**)

- ii) Was müssen Sie tun, damit die Fenster korrekt dargestellt werden? (**2 Punkte**)

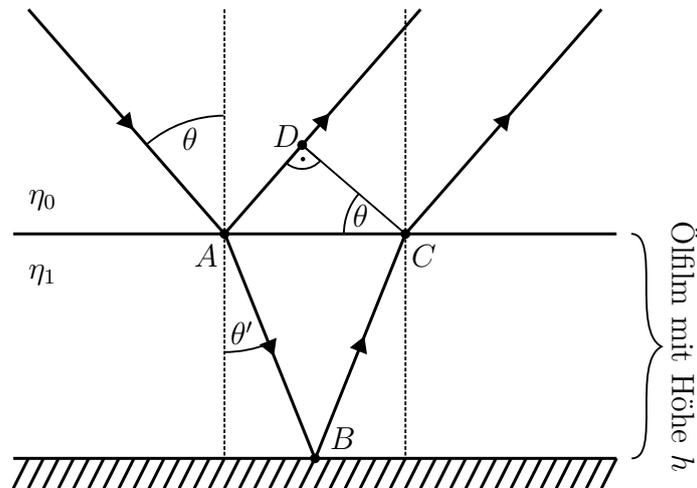
Aufgabe 8: Interferenzeffekte mit GLSL-Shader (38 Punkte)

Abbildung 11: Lauflängenunterschied zwischen reflektiertem und transmittiertem Strahl.

Im Folgenden werden Sie Interferenzeffekte durch eine dünne Schicht (z.B. ein Ölfilm auf Wasser) auf einer Oberfläche modellieren. Die Interferenz entsteht dabei durch den Lauflängenunterschied von an der dünnen Schicht reflektiertem und transmittiertem Licht (Abbildung 11).

- a) Bestimmen Sie zunächst den Lauflängenunterschied $\overline{AB} + \overline{BC} - \overline{AD}$ zwischen dem reflektierten und transmittierten Strahl in der GLSL-Funktion `pathDifference`! **(15 Punkte)**



Hinweis: Verwenden Sie das Snellsche Brechungsgesetz zur Berechnung der Transmissionsrichtung: $\eta_0 \sin \theta_0 = \eta_1 \sin \theta_1$. Außerdem können Sie trigonometrische Funktionen aus GLSL wie `sin`, `cos`, `tan` und `asin`, sowie griechische Buchstaben verwenden. Das Problem wird einfacher, wenn Sie Teilstrecken berechnen.

```
void pathDifference(
    float theta, float eta_0, float eta_1, float h) {
```

```
}
```

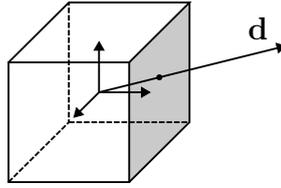


Abbildung 12: Cube Environment Map.

b) Die eingehende Beleuchtung soll mithilfe einer Cube Environment Map bestimmt werden (Abbildung 12).

i) Wie wird für eine Richtung \mathbf{d} *effizient* bestimmt, auf welche Seitenfläche zugegriffen wird? **(3 Punkte)**

ii) Sei nun für eine gegebene Richtung \mathbf{d} die Seitenfläche in positiver \mathbf{z} -Richtung gewählt. Ergänzen Sie die GLSL-Funktion um die Bestimmung der Texturkoordinate! **(5 Punkte)**

Hinweis: Beliebige Cube Map-Konventionen sind zulässig!

```
vec2 texCoordPositiveZ(vec3 d) {
    // Richtung d -> Texturkoordinate für Cube Map
```

```
}
```

c) Bestimmen Sie nun im folgenden Fragment Shader die reflektierte Farbe! **(15 Punkte)**

Verwenden Sie hierzu die GLSL-Funktionen:

- pathDifference aus Teilaufgabe a).
- envmap, die die eingehende Beleuchtung für eine Richtung L zurückgibt.
- computeIridescence, die für einen gegebenen Lauflängenunterschied die Interferenz bestimmt. Diese ist ein Faktor für jeden Farbkanal, der die Auswirkung der Interferenz beschreibt.

```
uniform vec3 cP;           // Position der Kamera in Weltkoordinaten
uniform float  $\eta_0$ ,  $\eta_1$ , h; // Oberflächenparameter

in vec3 wP; // Oberflächenposition in Weltkoordinaten
in vec3 wN; // Interpolierte Oberflächennormale in Weltkoordinaten

out vec4 fC; // Geben Sie hier den finalen Farbwert aus

vec3 envmap(vec3 direction) { ... }
vec3 computeIridescence(float pathDifference) { ... }

void main(){
    // Berechnen Sie hier den Lauflängenunterschied,
    // das eingehende Licht und die Interferenzeffekte!
```

```
}
```

Aufgabe 9: Prozedurale Modellierung (15 Punkte)

Ein gängiges Beispiel für die Verwendung von Noise-Funktionen $n(\mathbf{x})$ im Rahmen der prozeduralen Modellierung ist das Erzeugen von Höhenkarten für Landschaften (Height-maps). In dieser Aufgabe geht es darum, eine einfache Höhenkarte zu generieren und zu texturieren.

a) Wie wird aus einer stückweise konstanten Zufallstextur $z(\mathbf{x})$ eine Noise-Funktion $n(\mathbf{x})$? Nennen Sie zwei wichtige Eigenschaften, die $n(\mathbf{x})$ erfüllen muss! **(3 Punkte)**

b) Geben Sie die Formel für die Turbulenzfunktion aus der Vorlesung an! **(3 Punkte)**

c) Wie hängen die Oktaven mit der Größe der erzeugten Strukturen zusammen? **(2 Punkte)**

Matrikelnummer: _____

- d) Die Landschaft soll nun abhängig von der Höhe und der Steigung mit Schnee eingefärbt werden. Schnee liegt ab einer festen Höhe H_{SNOW} , wenn zusätzlich die Steigung weniger als 30° beträgt. Gegeben sind die Position und Normale an einem Punkt auf der Höhenkarte. Vervollständigen Sie die untenstehende GLSL-Funktion, die *true* zurückgeben soll, wenn Schnee liegt und *false*, wenn kein Schnee liegt. Die y -Achse zeigt hierbei nach oben. (7 Punkte)



```
uniform float H_SNOW;
```

```
bool snow_condition(vec3 position, vec3 normal){
```

```
}
```



Aufgabe 10: Bézier-Kurven (11 Punkte)

a) Gegeben sei die Bézier-Kurve $F(u) = \sum_{i=0}^4 \mathbf{b}_i B_i^4(u)$ vom Grad 4 mit $u \in [0, 1]$ und

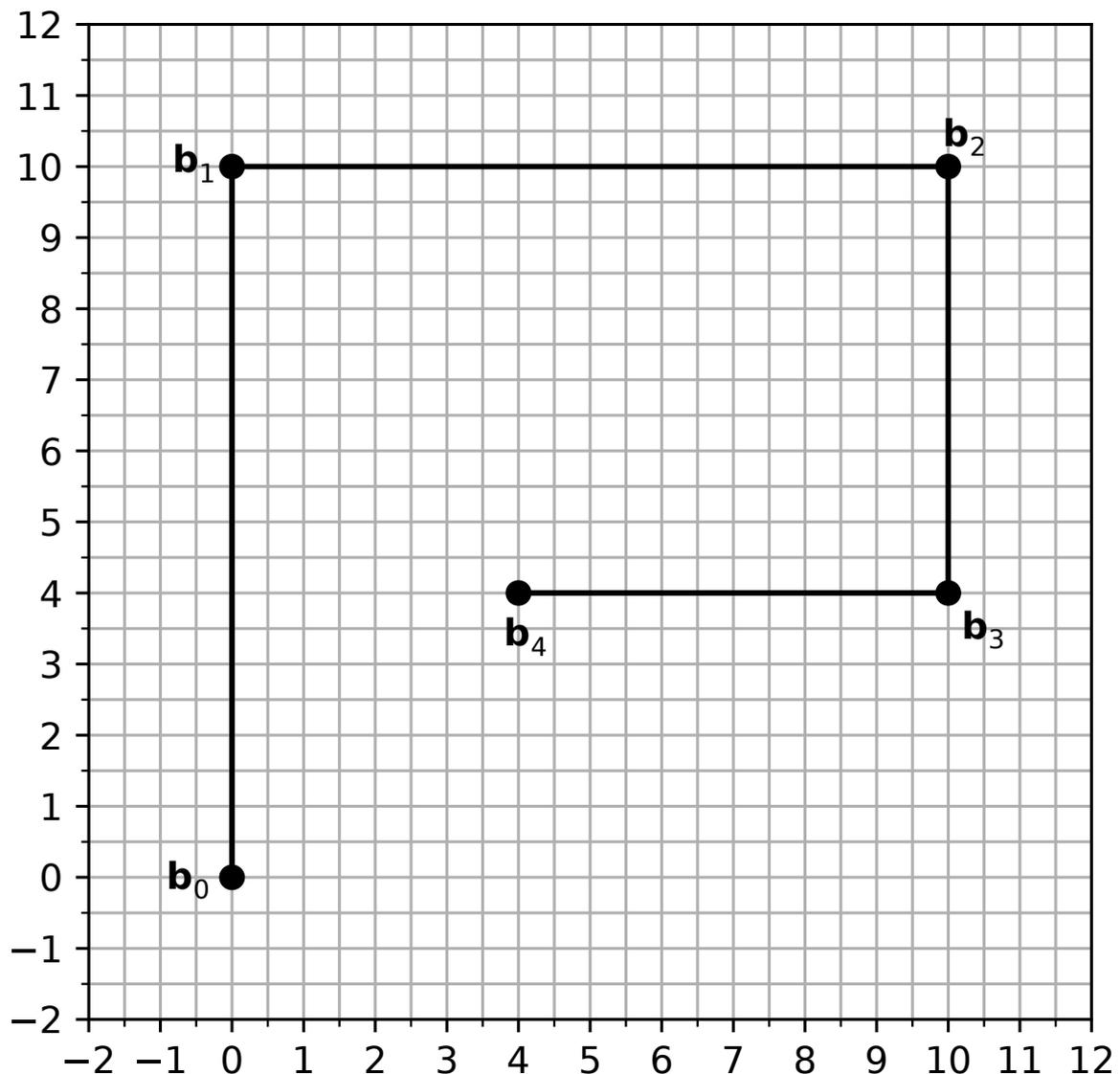
$$\mathbf{b}_0 = (0, 0), \quad \mathbf{b}_1 = (0, 10), \quad \mathbf{b}_2 = (10, 10), \quad \mathbf{b}_3 = (10, 4), \quad \mathbf{b}_4 = (4, 4).$$

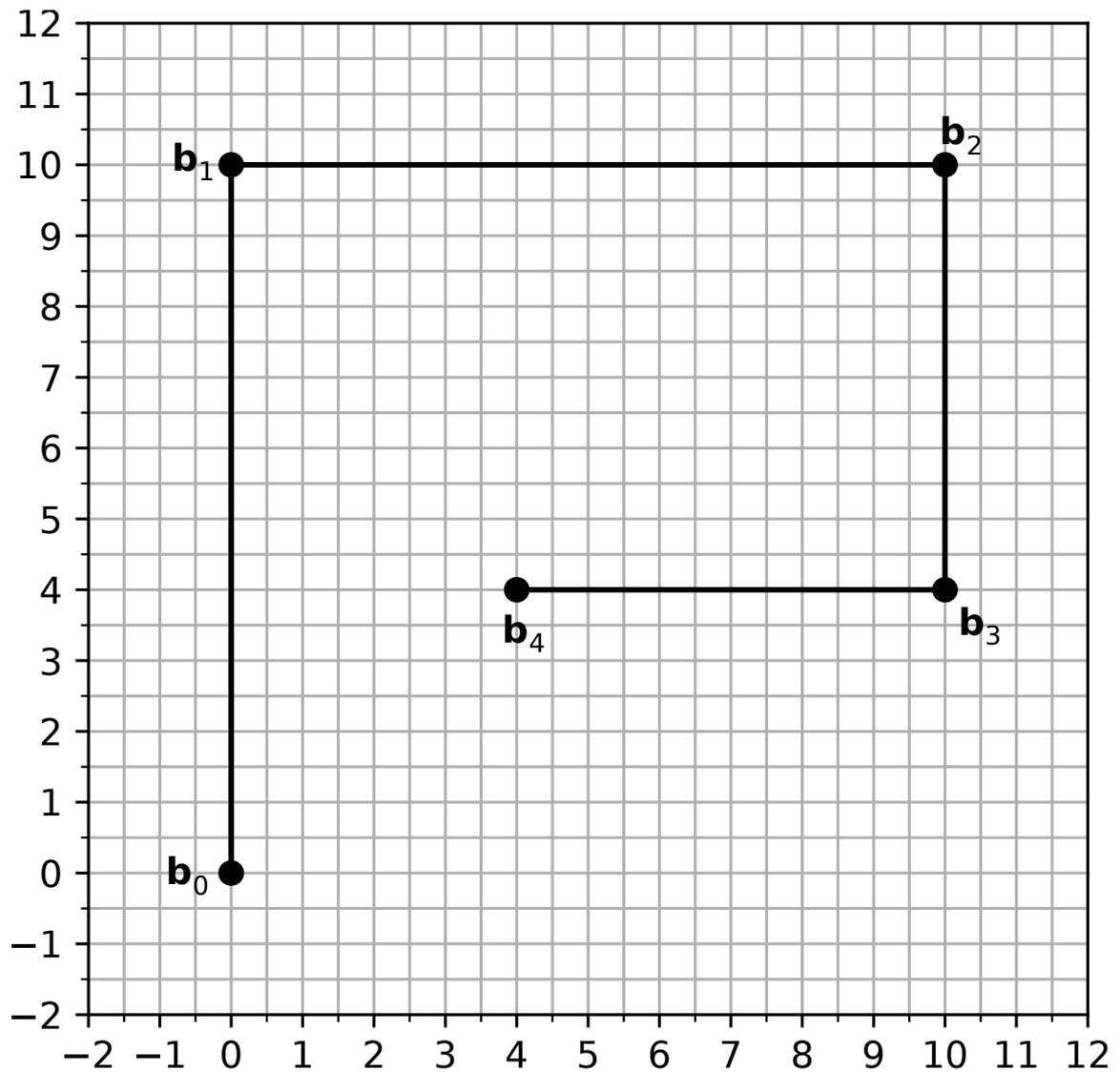
Bestimmen Sie den Punkt $F(0.5)$ in der Abbildung unten, indem Sie die Kurve *zeichnerisch* für $u = 0.5$ unterteilen. Machen Sie dabei die Kanten und Punkte der Kontrollpolygone der Teilkurven kenntlich!

Nutzen Sie anschließend die Kontrollpolygone der entstehenden Teilkurven, um $F(u)$ unter Ausnutzung der Eigenschaften von Bézier-Kurven zu skizzieren!



Zur Korrektur können Sie die zweite Zeichnung auf der nächsten Seite verwenden. Kennzeichnen Sie eindeutig, welche Zeichnung bewertet werden soll! (7 Punkte)





- b) Zusätzlich zur Bézier-Kurve $F(u)$ sei eine Gerade durch Punkt b_0 und Punkt $(11, 4)$ gegeben. Wie oft schneidet die Bézier-Kurve für $u \in [0, 1]$ die Gerade? Begründen Sie anhand der Eigenschaften von Bézier-Kurven, warum das so ist! (4 Punkte)

