

Aufgabe:	1	2	3	4	5	Σ
Punktzahl:	/15	/15	/15	/5	/10 Bonus	/50

Klausur: Computergraphik I (CG-1) Klausur	Prüfer: Kolb	Datum/Zeit: XX.XX.XXXX XX:XX Uhr
Semester: Probeklausur	Prüfungsdauer: 60 Minuten	Max. Punktzahl: 50 (+10 Bonus)
Hilfsmittel: • Schreibgeräte (kein Bleistift, keine Farbstifte (rot und grün)) • Lineal, Geo-Dreieck • Nicht-programmierbarer Taschenrechner		
Bitte beachten: • Tragen Sie auf jedes Blatt Ihren Namen und Ihre Matr.-Nr. in die entsprechenden Felder ein. • Benutzen Sie nur das ausgeteilte Papier – kein eigenes Papier. Notizen sind auf den Rückseiten der Klausurblätter zulässig und werden nicht als Lösungsbestandteil gewertet. Die Mitnahme von Klausurbögen aus dem Raum ist nicht gestattet. • Die Klammerung der Klausurbögen darf nicht gelöst werden. • Schreiben Sie deutlich! Doppelte, unleserliche oder mehrdeutige Lösungen sind ungültig. • Erscheint Ihnen eine Aufgabenstellung mehrdeutig, melden Sie sich bitte durch Handzeichen bei einer Aufsichtsperson. • Ein Täuschungsversuch (Abschreiben, Nutzung unerlaubter Hilfsmittel, Kommunikation mit anderen Klausurteilnehmern, etc.) führt umgehend zum Ausschluss und Nichtbestehen. Es erfolgt keine Vorwarnung.		

Bitte ausfüllen:

Name:	Matrikel-Nr.:
--------------	----------------------

Aufgabe 1 Transformationen und Szenengraphen (15 Punkte)

Abbildung 1 zeigt links ein geometrisches Objekt, das vollständig aus den geometrischen Primitiven A und B rechts in der Abbildung konstruiert werden soll. Alle Rechtecke des Objekts und der Kreis bewegen sich stets in der xy -Ebene. Der Kreis A ist in seinen lokalen Koordinaten um den Koordinatenursprung zentriert und hat den Radius 1. Der Arm links und die negative y -Achse schließen einen Winkel von 45° ein. Arme und Beine haben jeweils die Größe des Primitivs B .

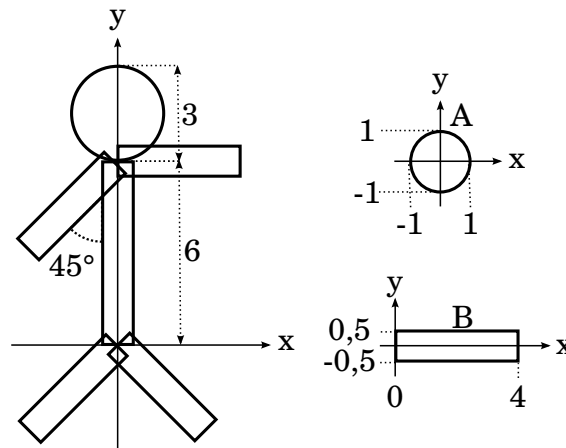


Abbildung 1: Links das zu konstruierende Objekt in Weltkoordinaten und rechts die beiden geometrischen Primitive A und B in deren lokalen Koordinatensystemen.

- 1.1. Geben Sie einen **Szenengraphen** an, der die Szene beschreibt. Für jedes gezeichnete Primitiv sollen dabei maximal zwei aufeinanderfolgende Transformationen durchgeführt werden. Verwenden Sie nur die folgenden Transformationen:

T_{t_x, t_y} = Translation in x, y Richtung um den Betrag (t_x, t_y)

R_ϕ = Rotation mit Winkel ϕ um die z -Achse

S_{s_x, s_y} = Skalierung in x, y Richtung um den Betrag (s_x, s_y)

- 1.2. Für Rotationen, Translationen und Skalierungen lassen sich direkt deren **inverse Transformationen** angeben. Die Matrizen R_ϕ und T_{t_x, t_y} beschreiben eine Rotation mit Winkel ϕ um die z -Achse bzw. eine Translation um den Vektor $(t_x, t_y)^T$. Gegeben sei die Matrix $M = T_{t_x, t_y} \cdot R_\phi$. Stellen Sie R_ϕ und T_{t_x, t_y} zunächst als 3×3 -Matrizen auf und berechnen Sie die Matrix M . Zeigen Sie rechnerisch, dass für die Inverse der Matrix M folgendes gilt:

$$M^{-1} = (T_{t_x, t_y} \cdot R_\phi)^{-1} = R_\phi^{-1} \cdot T_{t_x, t_y}^{-1}$$

Hinweis: Zur Lösung können Sie die Eigenschaft $MM^{-1} = \mathbb{1}$ verwenden.

Aufgabe 2 Polygon-Rasterisierung (15 Punkte)

In dieser Aufgabe sollen Sie mit Hilfe des **Scanline-Algorithmus** ein Polygon rasterisieren.

- 2.1. Wie und wonach müssen Einträge in der Edge-Table und in der Active-Edge-Table für den Scanline-Algorithmus sortiert werden?
- 2.2. Es sei ein Polygon durch die Punkte **A**, **B**, **C** und **D** mit

$$\mathbf{A} = \begin{pmatrix} 3 \\ 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 4 \\ 2 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 3 \\ 4 \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

und durch die Kanten $\overline{AB}$, $\overline{BC}$, $\overline{CD}$ und $\overline{DA}$ gegeben. Bestimmen Sie die zugehörige initiale Edge-Table und tragen Sie sie in der folgenden Tabelle ein:

ID	x	k	y_{min}	y_{max}	dx	rx	Δy

- 2.3. Skizzieren Sie zunächst das Polygon, das durch die folgende Edge-Table definiert ist:

ID	x	k	y_{min}	y_{max}	dx	rx	Δy
$\overline{AE}$	2	0	0	1	-1	0	1
$\overline{AB}$	2	0	0	4	0	1	4
$\overline{ED}$	1	0	1	4	0	0	3
$\overline{CD}$	2	0	2	4	-1	1	2
$\overline{CB}$	2	0	2	4	0	1	2

Führen Sie anschließend den Scanline-Algorithmus durch und fügen Sie für jede Scanline Ihre berechneten Werte in die folgende Active-Edge-Table ein. Geben Sie außerdem unten die berechneten Spans an.

Scanline	ID	x	k	LR: pixel (x)					
$y = 0$									
$\vdots$									

Spans: _____

- 2.4. Welches Problem kann beim Rasterisieren von Polygonen mit spitzen Winkeln auftreten?

Aufgabe 3 Point-In-Polygon Test (15 Punkte)

Im Folgenden soll ein Point-in-Polygon-Test durchgeführt werden, um zu prüfen, ob ein Strahl ein Polygon $\mathcal{P}$ mit den folgenden 4 Eckpunkten trifft.

$$\mathbf{P}_1 = \begin{pmatrix} 3 \\ 1 \\ -1 \end{pmatrix}, \mathbf{P}_2 = \begin{pmatrix} 5 \\ 1 \\ 0 \end{pmatrix}, \mathbf{P}_3 = \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix}, \mathbf{P}_4 = \begin{pmatrix} \frac{3}{2} \\ -\frac{5}{6} \\ 1 \end{pmatrix}$$

1.1. Bestimmen Sie den **Schnittpunkt $\mathbf{K}$** der Polygonebene mit dem Strahl

$$\mathbf{G}(\alpha) = \begin{pmatrix} 0 \\ -1 \\ -1 \end{pmatrix} + \alpha \begin{pmatrix} 0 \\ 2 \\ 2 \end{pmatrix}, \alpha \geq 0.$$

Stellen sie hierfür zunächst die Ebenengleichung der von $\mathcal{P}$ definierten Ebene auf.

- 1.2. Im nachfolgenden soll der Schnittpunkt $\mathbf{S} = (2, 0, 0)^T$ eines anderen Strahls mit der Polygonebene betrachtet werden. Projizieren sie das Polygon $\mathcal{P}$ sowie den Schnittpunkt $\mathbf{S}$ zweidimensional entlang der treibenden Achse der Normale der Polygonebene. Geben sie die **projizierten Punkte $\mathbf{P}'_i$** und $\mathbf{S}'$ an.
- 1.3. Verschieben sie anschließend die projizierten Punkte $\mathbf{P}'_i$, so dass der Schnittpunkt $\mathbf{S}$ im Ursprung liegt. Führen sie für die Kanten $\overline{\mathbf{P}'_i \mathbf{P}'_{i+1}}$ ($i = 1, \dots, 4$, $\mathbf{P}'_5 = \mathbf{P}'_1$) des verschobenen Polygons die **Trivial-Reject-Prüfung** entlang der x' -Achse durch.
- 1.4. Bestimmen sie für die verbleibende(n) Kante(n) die Schnittpunkte mit der x' -Achse und überprüfen sie, ob $\mathbf{S}$ **inner- oder außerhalb des Polygons** liegt.
- 1.5. Wie muss vorgegangen werden, wenn eine Kante vollständig auf der x' -Achse verläuft?

Aufgabe 4 Multiple Choice (5 Punkte)

In der folgenden Aufgabe sind zehn **Multiple Choice Fragen** zu beantworten. Setzen Sie ihr Kreuz entweder bei *korrekt* oder *falsch* oder lassen Sie die Zeile frei, wenn Sie sich unsicher sind.

Pro korrekter Antwort werden 0,5 Punkte vergeben, pro falscher Antwort -0,5 Punkte. In Summe sind für die gesamte Aufgabe nur nicht-negative Punktzahlen erreichbar.

Aussage	korrekt	falsch
Matrixmultiplikationen sind im Allgemeinen nicht kommutativ.	☐	☐
Der diffuse Beleuchtungsterm des Phong-Modells hängt nicht von der Lichtquellen- oder Beobachterposition ab.	☐	☐
Der Reflexionsvektor wird für die Berechnung des spekulären Phong-Beleuchtungsterms benötigt.	☐	☐
Szenengraphen dürfen keine Zyklen enthalten.	☐	☐
Die Beleuchtungsberechnung erfolgt in Normalized Device Coordinates (NDC).	☐	☐
Double-Buffering wird eingesetzt, um Flimmern bei der Bildwiedergabe zu verhindern.	☐	☐
Mit dem Sutherland-Hodgeman-Algorithmus können beliebige Strecken rasterisiert werden.	☐	☐
Das Ergebnis der Scanline-Interpolation allgemeiner Polygone hängt von deren Lage ab.	☐	☐
Mit Gouraud-Shading können visuell glatte Farbübergänge auf Flächen und an Kanten erzeugt werden.	☐	☐
Bei aktivem Standard-z-Test kann ein Objekt auch hinter einem bereits gezeichneten transparenten Objekt gezeichnet werden.	☐	☐

Aufgabe 5 Verallgemeinerte Reflexion (10 Bonuspunkte)

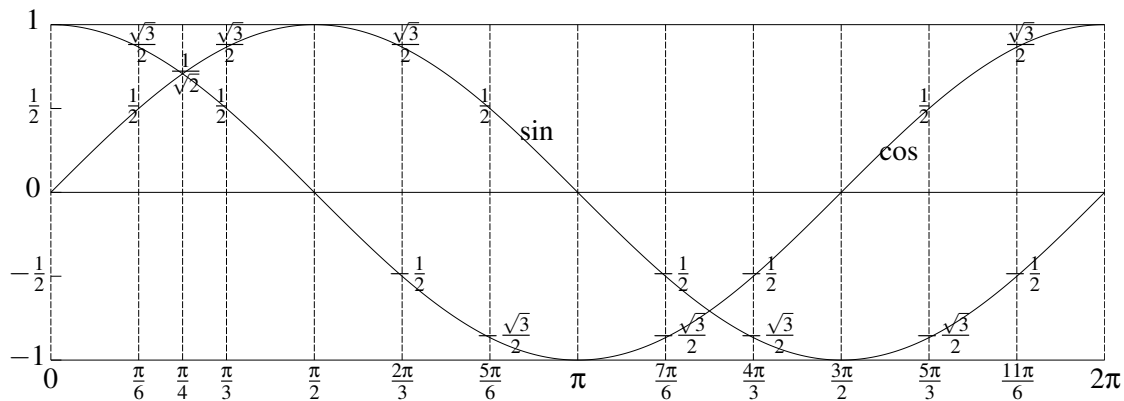
Im Folgenden sollen Sie die **Matrix** $R \in \mathbb{R}^{3 \times 3}$ bestimmen, die einen beliebigen Vektor $\vec{v} \in \mathbb{R}^3$ an einer Normalen $\hat{n} \in \mathbb{R}^3$ reflektiert.

Bearbeiten Sie hierzu nacheinander die folgenden Fragen/Schritte:

- 5.1. Überlegen Sie sich zunächst, wie die Reflexion von $\vec{v}$ an $\hat{n}$ mit Hilfe einfacher Vektoroperationen realisiert wird. Fertigen Sie dazu eine aussagekräftige Skizze an.
- 5.2. Zeigen Sie rechnerisch, dass allgemein $(\vec{v} \cdot \hat{n})\hat{n} = (\hat{n}\hat{n}^T)\vec{v}$ gilt. Stellen Sie dazu zunächst die Matrix $\hat{n}\hat{n}^T \in \mathbb{R}^{3 \times 3}$ auf.
- 5.3. Leiten Sie nun eine allgemeine Formel für die Matrix R her.
- 5.4. Prüfen Sie exemplarisch Ihre Matrix R , indem Sie den Vektor $\vec{v} = (1, 0, 3)^T$ an der Normalen $\hat{n} = (0, 1, 0)^T$ zweimal spiegeln. Was stellen Sie bezüglich der Inversen Matrix R^{-1} fest?

Anhang

Sinus und Cosinus



Skalarprodukt und Kreuzprodukt

$$(\vec{u} \cdot \vec{v}) = \sum_{i=1}^n u_i v_i, \quad \vec{u} \times \vec{v} = \begin{pmatrix} u_y v_z - u_z v_y \\ u_z v_x - u_x v_z \\ u_x v_y - u_y v_x \end{pmatrix}$$

$$(\vec{u} \cdot \vec{v}) = \cos \alpha \|\vec{u}\| \|\vec{v}\|, \quad \alpha \text{ einschließender Winkel}$$

Norm und normierte Vektoren

$$\|\vec{u}\| = \sqrt{(\vec{u} \cdot \vec{u})} = \sqrt{\sum_{i=1}^n u_i^2}$$

$\hat{u}$ ist normierter Vektor, falls $\|\hat{u}\| = 1$

Viewing Transformation

Rechtshändiges, orthonormales Beobachter-Koordinatensystem $\{\mathbf{V}, \hat{\mathbf{v}}_x, \hat{\mathbf{v}}_y, \hat{\mathbf{v}}_z\}$ in WC liefert

$$T_V = \begin{bmatrix} A^T & -A^T \mathbf{V} \\ 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \quad \text{mit} \quad A = (\hat{\mathbf{v}}_x, \hat{\mathbf{v}}_y, \hat{\mathbf{v}}_z)$$

Perspektivische Transformation

n und f sind die nahe bzw. ferne Cliepbene, b und h sind Breite bzw. Höhe auf der nahen Cliepbene.

$$T_P = \begin{bmatrix} \frac{2}{b} & 0 & 0 & 0 \\ 0 & \frac{2}{h} & 0 & 0 \\ 0 & 0 & \frac{n+f}{n(n-f)} & \frac{2f}{n-f} \\ 0 & 0 & -\frac{1}{n} & 0 \end{bmatrix}$$

Liang-Barsky-Algorithmus

Kante sei $\mathbf{P}_1 \mathbf{P}_2$. $\alpha_i = \frac{\text{wec}_i(\mathbf{P}_1)}{\text{wec}_i(\mathbf{P}_1) - \text{wec}_i(\mathbf{P}_2)}$, $i \in \{L, R, B, T\}$

Bresenham-Algorithmus

$$d_{\text{init}} = 2\Delta y - \Delta x, \quad \text{inc}_1 = 2\Delta y, \quad \text{inc}_2 = 2(\Delta y - \Delta x)$$

Scanline-Algorithmus

$$\frac{\Delta x}{\Delta y} = dx + \frac{rx}{\Delta y}, \quad dx \in \mathbb{Z}, \quad rx \in [0, \Delta y - 1] \subset \mathbb{N}_0$$