Institut für Informatik
Henning Wenke, M.Sc.
Nico Marniok, B.Sc.
Sascha Kolodzey, B.Sc.

Universität Osnabrück, 23.05.2012
`http://www-lehre.inf.uos.de/~cg`
Testat bis 30.05.2012, 15:00 Uhr

Übungen zu Computergrafik

Sommersemester 2012

Blatt 5

Aufgabe 5.1 (15 Punkte)

Thema: View und Projection Matrix. Laden Sie sich das Programm von `/home/cg/2012/Uebung/Blatt5/Aufg/CG12Blatt5.zip` und richten Sie es in Ihrer bevorzugten Entwicklungsumgebung ein.

1. Machen Sie sich mit der Klasse `util.Util` vertraut, damit Sie fortan grundlegende befehlsintensive Operationen mit ihrer Hilfe durchführen können.

2. Implementieren Sie die Methoden `lookAtRH(...)` und `orthographicRH(...)` in der Datei `Util.java`, sodass sie die Funktionalität bereitstellen, die im zugehörigen JavaDoc beschrieben wird. Testen Sie sie mittels der Klasse `test.Aufg512.java`.

Musterlösung vom 30.05.2012:

Die Quellen der Musterlösung sind unter `/home/cg/2012/Uebung/Blatt5/Lsg/CG12Blatt5.zip`zu finden.

Aufgabe 5.2 (35 Punkte)

Thema: VertexArrayObject. Erweitern Sie das Projekt aus Aufgabe 1.

1. Implementieren Sie die Methode `createQuad()` inder Datei `Util.java`. Hier soll ein VertexArrayObject mit dem Vertexlayout

   $$\{v_x, v_y, v_z, c_r, c_g, c_b, c_a\}$$

   erzeugt werden, wobei die ersten drei Werte die Position des Vertex (ohne homogene Koordinate) und die hinteren vier seine Farbe sind. Die Positionen der Ecken sollen $\{(-1,-1,0),(1,-1,0),(1,1,0),(-1,1,0)\}$ sein. Die Farbwahl ist Ihnen überlassen, soalnge sie sich ausreichend vom Hintergrund abhebt. **Hinweis:** Sollten Sie nicht wissen, wie Sie Farben definieren, benutzen Sie einfach $\{(0,1,1,1),(1,0,1,1),(1,1,0,1),(1,1,1,1)\}$.

2. Starten Sie das Programm. Die Kanten des Quads sollten genau mit dem Bildschirmrand übereinstimmen.

3. Implementieren Sie die Methode `createTriangle()` in der Datei `Util.java`. Hier soll ein VertexArrayObject mit dem Vertexlayout

$$\{c_r, c_g, c_b, c_a, v_x, v_y, v_z\}$$

erzeugt werden, wobei die ersten vier Werte die Farbe des Vertex und die hinteren drei seine Position (ohne homogene Koordinate) sind. Die Positionen der Ecken sollen $\{(-1,-1,0),(1,-1,0),(0,1,0)\}$ sein. Die Farbwahl ist Ihnen überlassen, soalnge sie sich ausreichend vom Hintergrund abhebt und sich von der des Vierecks unterscheidet. **Hinweis:** Sollten Sie nicht wissen, wie Sie Farben definieren, benutzen Sie einfach $\{(1,0,0,1),(0,1,0,1),(0,0,1,1)\}$.

4. Starten Sie das Programm. Die untere Kante des Dreiecks sollte sich am unteren Bildschirmrand befinden, während die Spitze genau in der Mitte des oberen Bildschirmrandes liegt.

Musterlösung vom 30.05.2012:

Die Quellen der Musterlösung sind unter `/home/cg/2012/Uebung/Blatt5/Lsg/CG12Blatt5.zip`zu finden.

Aufgabe 5.3 (30 Punkte)

Thema: Modeling Transformation. Erweitern Sie das Projekt aus Aufgabe 1.

1. Ordnen Sie vier der Quadrate so an, dass sie einen Würfel ergeben. Dazu sind die vier Model Transformation Matrizen bereits als Klassenvariable `cubeModel` deklariert und initialisiert und die Methode `render()` sorgt bereits dafür, dass vier Kopien mit diesen Model Matrizen gezeichnet werden. Implementieren Sie die Methode `initCube()`, die diese vier Matrizen so setzt, dass ein Würfel der Kantenlänge $d = \frac{1}{4}$ entsteht. Verschieben Sie den Würfel außerdem um $t_x = \frac{1}{2}$ Einheiten in x Richtung. **Hinweis:** Da wir nur das Drahtgitter des Würfels zeichnen, genügen vier Kopien des Vierecks.

2. Ordnen Sie vier der Dreiecke so an, dass sie eine Pyramide ergeben. Dazu sind die vier Model Transformation Matrizen bereits als Klassenvariable `pyramidModel` deklariert und initialisiert und die Methode `render()` sorgt bereits dafür, dass vier Kopien mit diesen Model Matrizen gezeichnet werden. Implementieren Sie die Methode `initPyramid()`, die diese vier Matrizen so setzt, dass eine Pyramide der Kantenlänge $d = \frac{1}{4}$ entsteht (sowohl Länge der Kanten der Grundseite, als auch die der Schrägen). Verschieben Sie die Pyramide außerdem um $t_x = -\frac{1}{2}$ Einheiten in x Richtung. **Hinweis:** Da wir nur das Drahtgitter der Pyramide zeichnen, genügen vier Kopien des Dreiecks.

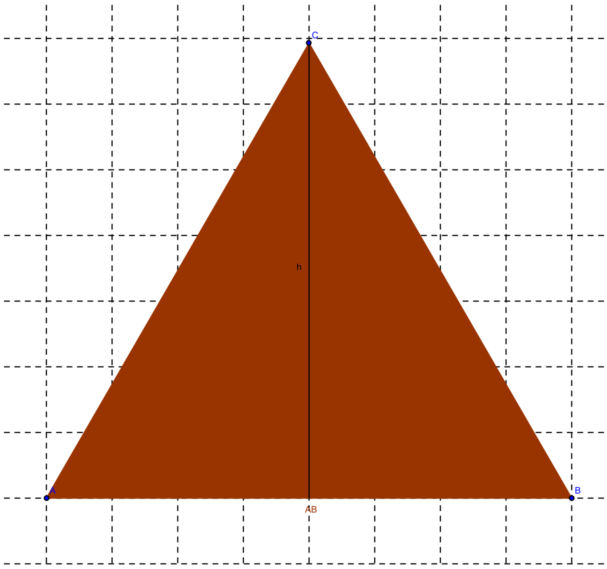
3. Halten Sie **schriftlich** fest, wie Sie den Winkel berechnen, um den Sie das Dreieck kippen müssen, um eine Pyramidenseite zu erhalten.

4. **Zusatzaufgabe:** Implementieren Sie die Möglichkeit den Würfel mittels der Tasten $\uparrow$, $\leftarrow$, $\downarrow$ und $\rightarrow$ zu drehen. Die Drehgeschwindigkeit soll erhöht sein, wenn man zusätzlich die rechte Shift-Taste gedrückt hält. Während des Drehens soll sich der Mittelpunkt des Würfels nicht ändern.

Musterlösung vom 30.05.2012:

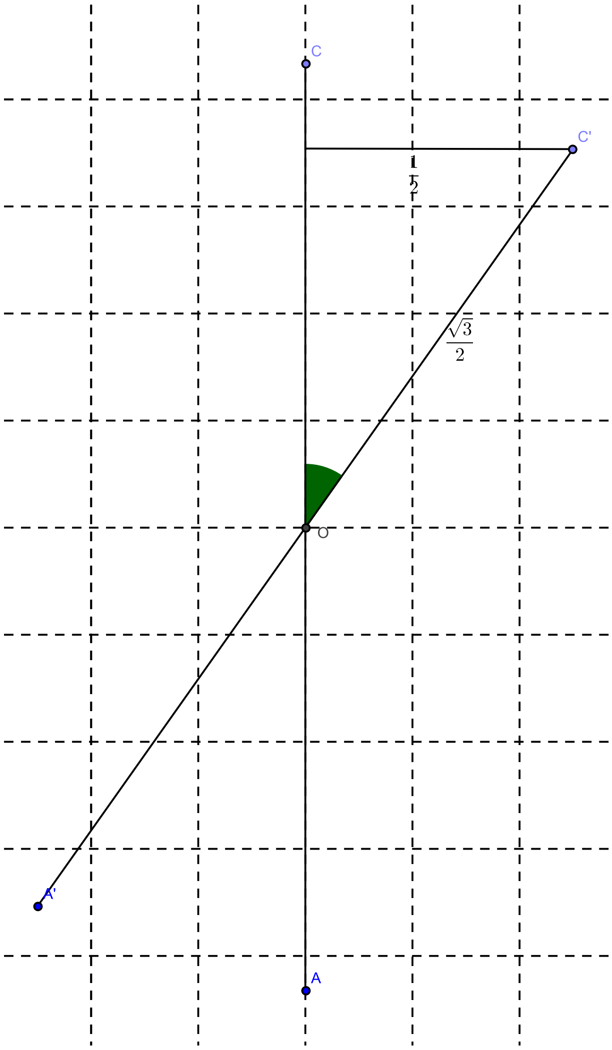
Die Quellen der Musterlösung sind unter `/home/cg/2012/Uebung/Blatt5/Lsg/CG12Blatt5.zip`zu finden.

- Das Dreieck besteht anfangs aus den Koordinaten $A = (-1,-1)$, $B = (1,-1)$ und $C = (0,1)$. Damit ergibt sich nach Pythagoras eine Schenkellänge von $s = \sqrt{5} \neq 2$. Wir skalieren das Dreieck zunächst in y-Richtung, sodass die Schenkel eine Länge von $s = 2$ erhalten. Die Höhe h eines gleichseitigen Dreiecks mit Seitenlänge s ist nach Pythagoras $h = \sqrt{s^2 - \left(\frac{s}{2}\right)^2} = \sqrt{\frac{3}{4}s^2} = \sqrt{3} \cdot \frac{s}{2}$. In unserem Fall mit $s = 2$ ist also $h = \sqrt{3}$. Da unsere aktuelle Höhe $h_0 = 2$ ist, müssen wir also um $\frac{\sqrt{3}}{2}$ in y-Richtung skalieren, um ein gleichseitiges Dreieck mit Seitenlänge $s = 2$ zu erhalten. Die Höhe des Dreiecks beträgt dann $h = \sqrt{3}$.



Skalierung des Dreiecks (Ansicht von vorne)

- Nun kippen wir das Dreieck, sodass die Entfernung von der Spitze zur unteren Kante in x-Richtung genau $t = 1$ beträgt. Aus der Zeichnung ergibt sich die Beziehung $\sin\alpha = \frac{1/2}{\sqrt{3}/2} = \frac{1}{\sqrt{3}}$. Also ist $\alpha = \arcsin\frac{1}{\sqrt{3}}$.



Kippen des Dreiecks (Ansicht von der Seite)

Aufgabe 5.4 (20 Punkte)

Beantworten Sie Ihrem Tutor Fragen zu den Inhalten der Vorlesung.