

Übungen zu Computergraphik I

– Übungsblatt 4 –

Lehrstuhl für Computergraphik
und Multimediasysteme

Hendrik Hochstetter, Bianca Kretz, Rene Winchenbach

Abgabe: Bis spätestens 8. Mai 2014, 10 Uhr

Besprechung: Mittwoch 14. Mai und Freitag 16. Mai 2014

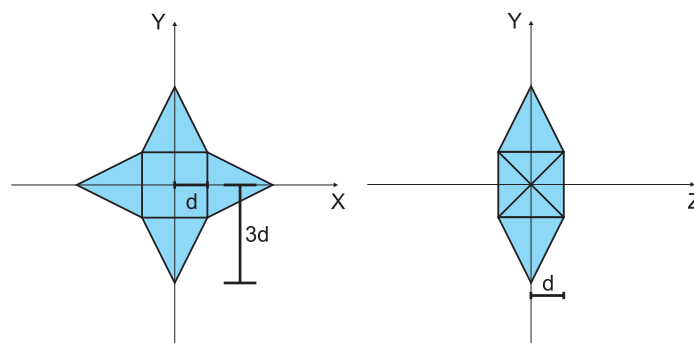
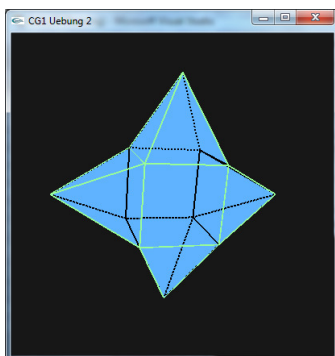
Hinweis: Die Programmieraufgaben müssen per E-Mail an Ihren Tutor eingereicht werden. Geben Sie dabei bitte immer Ihren Namen, Ihre Matrikelnummer, sowie Ihre Übungsgruppe an.

Aufgabe 1 Farbräume + Geometrie (2 Punkte)

In der folgenden Aufgabe soll in OpenGL ein sternenförmiges Objekt dargestellt werden. Nehmen Sie als Ausgangsbasis das auf unserer Website bereitgestellte Programmgerüst `ueb04.zip`. Um das Projekt zu kompilieren, folgen Sie der Anleitung auf der CG1 Übungsseite. In Kapitel 2 der Online Anleitung des „OpenGL Programming Guide“ finden Sie die nötigen OpenGL-Befehle: <http://www.glprogramming.com/red/chapter02.html>

Das Programm enthält bereits die Funktion `drawStar()`, in der folgende Aufgaben zu lösen sind:

- 1.1 Konvertieren Sie den CIE Farbvektor, gegeben durch die Parameter (X_CIE, Y_CIE, Z_CIE) , in den RGB Farbraum.
- 1.2 Definieren Sie nun das in der Zeichnung definierte sternenförmige Objekt mit der Längeneinheit d , die der Funktion `drawStar()` als Parameter übergeben wird. Entfernen Sie dazu zunächst das Rendering des vordefinierten Torus-Objekts `glutSolidTorus()`.



Aufgabe 2 Geometrie (1 Punkte)

In geschlossenen Objekten treten Eckpunkte meist in mehreren Primitiven gleichzeitig auf. Werden alle Primitive getrennt gezeichnet, ergibt sich ein sehr hoher Datentransferaufwand. Mit Hilfe von Dreiecksfächern (`GL_TRIANGLE_FAN`) können mehrere in einem gemeinsamen Eckpunkt zusammenhängende Dreiecke deutlich effizienter dargestellt werden.

- 2.1 Beschreiben Sie einen Würfel unter Verwendung von genau zwei Dreiecksfächern.
- 2.2 Wieviele Aufrufe von `glVertex3f` können oben durch Dreiecksfächer im Vergleich zum Zeichnen von Einzeldreiecken bzw. Einzelvierecken eingespart werden?