

Übungen zu Grafikprogrammierung in C++

Abgabetermin: 13. Juni 2005 (zwei Wochen Bearbeitungszeit!)

Aufgabe 18 (H) Texture Mapping

Sie können nun 3D Objekte zeichnen, Szenen modellieren und mit Licht versehen. Jedoch erscheinen die 3D Objekte in der Szene immer noch sehr unrealistisch, da den Objekten selbst bei guter Lichtgebung lediglich Farben einer Farbpalette zugewiesen werden. Um die Szene noch realistischer wirken zu lassen, kann man Texturen verwenden.

Lesen Sie sich Kapitel 9, 'Texture Mapping', im OpenGL Programming Guide (Red Book) durch. Texturen sind im Wesentlichen Bilder, die auf die Oberfläche von Strukturen (z.B. Polygonen) abgebildet werden.

Um Bilder (Texturen) laden zu können, verwenden Sie die 'imagelib', indem Sie das VS.NET Project-File `aufgaben/imagelib/imagelib.vcproj` einbinden. Wie schon aus einer früheren Aufgabe bekannt, können Sie mit der Klasse `Bitmap` eine `.bmp`-Datei laden und verwenden.

- Erweitern Sie die Bibliothek `polylib` so, dass shapes auch Texturen auf ihre Oberflächen abbilden können. Versuchen Sie, den Code möglichst überschaubar zu halten und durch Vererbung das Kopieren von Code zu vermeiden. Um Konstruktoren von Superklassen aufzurufen, verwenden Sie *Initializer Lists*. Stellen Sie sicher, dass alle Membervariablen initialisiert werden.
- Bilden Sie verschiedene Texturen Ihrer Wahl auf einen Tetraeder und Quader ab, die auf dem Tisch neben der Teekanne aus dem vorigen Ausgabenblatt stehen. Bitte vergessen sie nicht die Bilder ins *SVN*-Repository zu stellen.
- Da man diese 3D Objekte skalieren kann, verwenden Sie nach Möglichkeit *mipmapping*, um bei bestimmten Skalierungsgrößen auf geringere oder höhere Auflösung der Texturen umzuschalten.
- Versuchen Sie, mit geeigneten Texturen die Teapot-Szene realistischer zu machen (z.B. Holztisch). Eine Teekanne kann durchsichtig sein - versuchen Sie dies (mittels *blending*) zu realisieren.

Aufgabe 19 (H) Das obligatorische „Hello World“ mit QT

QT ist ein plattformunabhängiges Toolkit fuer graphische Benutzeroberflächen (GUI). In einem einfachen Programm soll ein `QPushButton` mit der Aufschrift „Hello World!“ in einem einfachen Fenster implementiert werden. Alle benötigten Pfade sollen relativ eingebunden werden. Die include-Dateien, sowie die benötigten Libraries sind compiliert im Verzeichnis `aufgaben/qt` zu finden.

(Hinweis: Die benötigten Libraries sind `QtGui4.lib` und `QtCore4.lib`, der `PATH` muss um das Verzeichnis mit den DLLs erweitert werden.)

Das Fenster soll nun um einen weiteren Button („beenden“) erweitert. Das Drücken dieses Buttons soll die Applikation beenden.

Detaillierte Hilfe und Tutorials zu QT sind unter <http://doc.trolltech.com/4.1/index.html> zu finden.

Aufgabe 20 (H) Ein Text Editor mit QT

Es soll ein kleiner Texteditor implementiert werden. Mittels einer `QComboBox` soll die Schriftart auswählbar sein, mittels `QSpinBox` soll die Schriftgröße auswählbar sein. Bei Änderungen in einem der beiden Eingabefenster soll unmittelbar die Schriftart bzw. die Schriftgröße des Textfeld geändert werden.

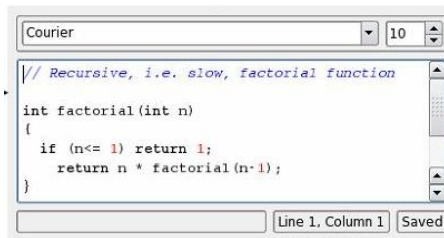


Abbildung 1: Exemplarischer Screenshot des zu implementierenden einfachen Texteditors.

Optional kann eine Menuleiste mit Funktionen zum Laden und Speichern von simplen Textdokumenten (ohne Formatierungen), sowie zum Beenden des Texteditors implementiert werden.