

Übungen zu Grafikprogrammierung in C++

Abgabetermin: 19. Juni 2006

Aufgabe 21 (H) OpenGL mit QT

Die Teekesselanimation aus Aufgabe 17 soll mittels QT implementiert werden. Neben einem OpenGL 3D rendering context soll ein widget mit Buttons und Eingabefeldern gezeichnet werden, mit denen man die Szene manipulieren kann. Der Code für den Teekessel kann weitgehend von Aufgabe 17 übernommen werden.

Es sollen zwei Widgets realisiert werden, die in einer Anwendung nebeneinander angeordnet sind. Das linke Fenster soll ein von `QGLWidget` abgeleitetes Widget sein, das rechte Fenster soll mindestens einen Button für das Starten der Animation und zwei Slider zum Verschieben der Kamera in x und y Richtung enthalten. Die Szene in dem von `QGLWidget` abgeleiteten Fenster soll via Maus intuitiv manipuliert werden können. Dafür müssen die entsprechenden Funktionen wie z.B. `mouseMoveEvent(QMouseEvent *e)` oder `mousePressEvent(QMouseEvent *e)` überladen werden.

Um eigene SLOTS und SIGNALS verwenden zu können, muss der QT moc-Kompiler verwendet werden. Im VS kann dies durch einen *Custom Build Step* realisiert werden. Dieser kann über die *Properties* des Headers der jeweiligen Klasse definiert werden. Die Parameter hierfür sind:

- *Command Line*, z.B.:
`..\..\..\aufgaben\Qt\bin\moc.exe "$(InputPath)" -o "$(InputDir)\moc_$(InputName).cpp"`
- *Description*:
Was auch immer in dem VS Output-Fenster dargestellt werden soll, wenn dieser Schritt ausgeführt wird, z.B.:
`MOCen von $(InputName)`
- *Outputs*, z.B.:
`"$(InputDir)\moc_$(InputName).cpp"`

Zudem muss das Macro `Q_OBJECT` eingebunden werden. Zuletzt muss die erzeugte („geMOCte“) Datei noch zu den Quelldateien im VS hinzugefügt werden.

Im Folgenden ist ein kurzes Codefragment für eine von `QGLWidget` abgeleitete Klasse. Diese erlaubt durch das Überladen von Funktionen, benutzerdefinierte Aktionen bei bestimmten Signalen auszuführen (siehe `mousePressEvent`).

```
class QustomGLThing : public QGLWidget {
    Q_OBJECT
public:
    QustomGLThing();
```

```
~QCustomGLThing();  
private:  
    void mousePressEvent(QMouseEvent *e);  
    void mouseReleaseEvent(QMouseEvent *e);  
public slots:  
    void startAnimation();  
};
```