



Aufgabenblatt 4

23.04.2018

Aufgabe 8: Rotation um eine beliebige Achse

Implementieren Sie die Rotation um eine beliebige Achse gemäß der Herleitung aus der Vorlesung.

- a) Erstellen Sie die notwendigen Einzel-Matrizen und multiplizieren Sie diese miteinander um die Gesamt-Rotationsmatrix zu erhalten. Sie sollen dies zum Verständnis der Funktionalität tatsächlich Schritt für Schritt von Hand machen, obwohl es eine fertige Rotationsmatrix in der glm-Library gibt. Die Rotationsachse und der Winkel soll per GUI einstellbar sein. Diese soll mit Hilfe der vorhandenen Funktionalität für Zylinder und Kegel als 3D Pfeil-Objekt gezeichnet werden.

Per Button-Klick soll ein aus einer obj-Datei geladenes Objekt zur Veranschaulichung entsprechend rotiert werden.

- b) Schreiben Sie einen Unit-Test, der die Funktionalität ihrer Matrix testet und dabei auch mit der glm-Implementierung vergleicht. Der Unit-Test soll per Button-Klick ausgeführt werden.

Aufgabe 9: Szene-Graph

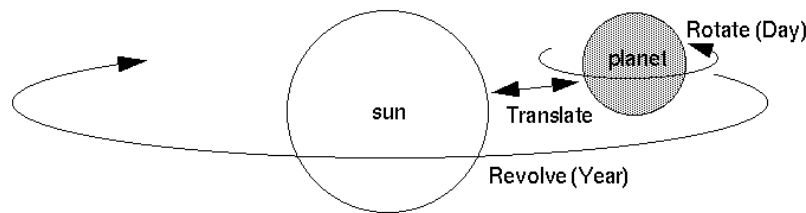
Implementieren Sie einen Szenegraph, der ihre Objekte und deren Transformationen enthält.

- a) Implementieren Sie die Klassen-Struktur gemäß der Vorlesung und entsprechende Methoden um Kind-Objekte oder Objekte innerhalb der Objekt-Liste hinzuzufügen oder zu löschen.
- b) Implementieren Sie eine Render-Funktionalität, die die beschriebene Matrix-Stack Funktionalität umsetzt und die Transformationsmatrix beim Render-Aufruf entsprechend an den Renderer übergibt.
- c) Es soll möglich sein, ein und dasselbe geometrische Objekt in unterschiedlicher Größe und Ausrichtung an unterschiedliche Position mehrfach zu zeichnen, obwohl es nur einmal existiert. Initialisieren Sie dazu das Objekt einmal mit seiner eindeutigen ID, beim Render Aufruf soll dieses Objekt dann nacheinander mit unterschiedlichen Transformationen an den Renderer übergeben werden.

Aufgabe 10: Implementierung eines Planetensystems

Testen Sie ihre Implementierung des Szenegraphen, indem Sie eine Animation eines Planetensystems schreiben. Zur Umsetzung soll also sowohl ihr Szenegraph als auch ihre selbst implementierte Rotation um eine beliebige Achse genutzt werden.

- Das Planetensystem soll eine Sonne und mindestens zwei unterschiedlich große Planeten enthalten, die mit verschiedenem Radius auf je einer unterschiedlich geneigten Kreisbahn unterschiedlich schnell um die Sonne kreisen und sich um ihre eigene Achse drehen.



Schemazeichnung Planetensystem (unvollständig)

- Einer der Planeten soll etwas in seiner Achse geneigt sein und einen Mond besitzen. Die Neigung kann zusätzlich z.B. durch die Ringe wie beim Saturn verdeutlicht sein (durch einige Kreise als Polyline).
- Der zweite Planet soll zwei Monde haben, die ihn auf der gleichen Kreisbahn gleich schnell umkreisen (d.h. ihr relativer Abstand ändert sich nicht).
- Führen Sie eine Variable `tag` ein, die die Dauer eines Umkreises um die Sonne (z.B. 365 Tage) bzw. die Dauer einer Rotation um die eigene Achse eines des Planeten beschreibt (z.B. 1 Tag).
- Der `tag` -Wert soll mit zwei Tasten um je einen festen float-Wert erhöht werden, so dass eine schnelle oder langsame Animation entsteht. Rechnen Sie dies in einen Winkel für die Rotation um, um die Position und Orientierung des Planeten zu beschreiben. Die Umlaufdauer des Mondes um den Planeten soll 28 Tage betragen.
- Die entsprechenden Werte der Umlaufzeit, Rotation und Mond-Umlaufzeit für den zweiten Planeten sollen ein beliebig zu wählendes Vielfaches (z.B. doppelte Zeit, per Gui einstellbar) der Werte für den ersten Planeten sein.
- Insgesamt soll durch die Veränderung des Werte für `tag` über die Tastatur das Planetensystem animiert werden.
- Nutzen Sie zur Darstellung eine selbst implementierte Kugel (analog zu Zylinder und Kegel).

Wichtiger Hinweis: Es handelt sich um ein einziges Geometrisches Objekt (Kugel), das an verschiedenen Stellen gezeichnet wird. Es soll also nur eine Geometrie geben, die immer wieder an unterschiedliche Positionen gezeichnet werden soll, ebenso wie es nur eine Polyline für mehrere Kreise geben soll.