

Die bestehende Sicht soll so geändert werden, dass farbige 3D-Objekte gezeichnet werden.

## 1 Vorbereitung

- Sie benötigen zusätzlich GLM für das folgende Programm. Windows: `pacman -S mingw-w64-x86_64-glm`, Linux `libglm-dev`. GLM ist eine für OpenGL geeignete Implementierung von Matrizen. Die Implementierung ist komplett in Header-Dateien verborgen. Es gibt kein Link-Library dazu.
- Laden Sie sich `viewer.zip` herunter.
- Übersetzen Sie das Programm mit `cmake`.
- Starten Sie es.
- Das Programm lädt innerhalb der `main`-Methode die angegebene Wavefront-Datei ein und zeigt sie an.

## 2 Objekte modellieren

Lesen Sie sich auf jeden Teil die nachfolgende Aufgabe durch, bevor Sie mit der Modellierung starten.

- Modellieren Sie, z.B. mit Blender, Raumschiff, Torpedo, fliegende Untertasse und einen der vier Asteroidtypen. Der Asteroid soll von oben betrachten etwa die Umrisse des Originals haben.
- Verwenden Sie Farbe bei den Objekten. Es werden später nur ambiente Farben ohne Alphakanal übernommen. Keine Texturen oder anderen Materialeigenschaften.
- Jedes Objekt darf höchstens aus 300 Dreieck bestehen.
- Sie müssen jedes Objekt einzeln im Wavefront-Obj-Format exportieren. Achten Sie darauf, dass beim Export die Flächen trianguliert werden. Material (die Farben) müssen ebenso mit exportiert werden.
- Betrachten Sie die Objekte mit dem `viewer`. Sie müssen die Datei in der `main`-Methode angeben und ggf. den Skalierungsfaktor ändern.

## 3 Asteroids mit 3D-Objekten implementieren

Sie können diesen Teil auch zuerst starten, in dem Sie erst einmal nur ein Objekt, z.B. einen Quader für alle zu rendernden Objekte verwenden.

- Ändern Sie Ihre bestehende Asteroids-Implementierung ab, so dass Ihre modellierten Objekte importiert werden. Verwenden Sie dazu den entsprechenden Quelltext `wavefront.cc` und `wavefront.h`. Alle Wavefront-Obj-Dateien müssen im Verzeichnis des Projekts liegen, sonst werden Sie nicht gefunden.
- Die Objekte müssen bei Start des Programms in Vertex-Buffer-Objekte transferiert werden. Sie können das gleiche Speicherlayout verwenden, wie beim `viewer`.

- Sie können ebenso den Vertex- und Fragmentshader des Viewers verwenden. Es wird Lambertsches Shading der Eckpunkte durchgeführt mit einer im Shader definierten Lichtquelle. Die Farbewerte werden bei Rasterung interpoliert, es handelt sich also um Gouraud-Shading.
- Ansonsten müssen Sie das Programm so abändern, dass Ihre 3D-Objekte verwendet werden. Achten Sie darauf, dass die Größe mit den gegebenen Ausdehnungen übereinstimmt.
- Die Explosion können Sie analog zur Bestehenden implementieren: Statt einzelne Punkte in verschiedene Richtungen zu verschieben, können Sie kleine Isokaeder oder ähnliches verwenden.
- Sie können das horizontale- und vertikale Scrolling aus der vorangehenden Aufgabe auch wieder rückgängig machen, falls es stört.
- Vergessen Sie nicht den Tiefentest generell anzustellen (`glEnable(GL_DEPTH_TEST);`) und die Tiefeninformation bei jedem erneuten Zeichnen zu löschen (`glClear ( GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT );`).
- GLM oder weitere zusätzliche Bibliotheken dürfen Sie nicht verwenden.