

Klausurvorbereitung Computergrafik 2, WS 2011/2012

Superwichtige / sehr wichtige / wichtige Lernziele des Semesters:
(*alle* Lernziele können potentiell in der Klausur abgefragt werden!)

- Rendering-Gleichung und BRDF
- **Raytracing-Algorithmus: Hauptschleife, Schnittberechnung, Verdeckungsberechnung**
- **Richtungen und Punkte, Vektor von A nach B**
- **Bedeutung und Definition des Skalarprodukts**
- **Bedeutung des Kreuzprodukts**
- **Normalenvektor**
- Normalisierung eines Vektors
- Implizite Form der Kreis- und Kugelgleichung
- Ebenen-Gleichung: explizite und implizite Form, hessesche Normalform
- **Strahl: parametrische Liniengleichung**
- **Kameramodell, Generierung von Primärstrahlen**
- Schnittberechnung Strahl-Ebene, Strahl-Kugel,

- **Lokale vs. globale Beleuchtung**
- **Phong-Modell: ambient, diffuse, spekulär; was ist jeweils die Bedeutung / Anschauung; welche Größen / Winkel gehen in welche Phong-Terme ein? Was tut der Phong-Exponent?**
- **Was ist der Unterschied zwischen Lighting und Lightning?** (niemand, der diesen Kurs abschließt, soll jemals fälschlich Lightning sagen!!!)

- **Was ist Schatten, Soft Shadow, Halbschatten.**
- Integration von Schattenberechnungen in die Phong-Formel
- **Raytracing von Schatten mittels Schattenstrahlen**
- Mehrere Lichtquellen / Flächenlichtquellen
- **Auswirkung numerischer Ungenauigkeiten / Self Intersection**
- **Rekursives Raytracing: was ist da rekursiv? Was bestimmt die Rekursionstiefe?**
- Reflexion, Refraktion, Fresnel/Schlick
- **Was berechnen die Fresnel-Gleichungen bzw. die Schlick-Approximation?**
- Implementierung eines Raytracers: **wie funktionieren die wichtigsten Methoden? (`intersect()` mit der Szene, `intersect()` mit dem Shape, `shade()` mit einem Phong-Modell, ...)** Welche Sonderfälle muss man abfangen

(Objekt hinter Auge, Blocker-Objekt hinter Lichtquelle); wie erkennt man solche Fälle?

- **Matrizen und Transformationen: was ist der Zusammenhang? Warum 4x4-Matrizen? Was sind homogene Koordinaten? Welche Transformationen kann man mit einer 3x3 / 4x4-Matrix darstellen? Homogene Koordinaten, Unterschied Richtung / Punkt?**
- **Wie setzt man Transformationen zusammen? Welche Rolle spielt die Reihenfolge?**
- **Wie transformiere ich von einem Koordinatensystem in ein anderes? Beispiele dafür, welche wichtigen K-Systeme kennen Sie?**
- **Was muss ich beim Transformieren von Vertices und Normalen beachten? Mit welcher Matrix transformiere ich sie von O nach E?**
- Was ist ein Szenengraph? Wie stellt man ihn üblicherweise dar?
- Rekursive Traversierung des Szenengraphen; Kompositum

- **Rasterisierungs-Pipeline: 3 konzeptionelle Stufen**
- Einordnung DirectX / OpenGL / OpenGL ES / WebGL
- Klassische Pipeline erklären / in die richtige Reihenfolge bringen: Transform & Lighting, Projektion, Clipping, Rasterisierung, Texturierung, Blend & Merge
- **Unterschiede klassische / programmierbare Pipeline**
- Unterschiede OpenGL 3/4 vs. OpenGL ES / WebGL (was kann WebGL nicht)
- **Wie ist die WebGL-Pipeline strukturiert? (Aufgabe z.B.: die richtigen Dinge in die richtige Reihenfolge bringen!)**
- **Datenfluss? Wie und wo kommen welche Daten in die Pipeline?**
- Wo in der Pipeline befindet man sich in welchem Koordinatensystem?
- **Vertex Shader, was ist ein Vertex?**
- **Fragment Shader, was ist ein Fragment?**
- **Rasterisierung und Interpolation**
- Einfaches Shader-Beispiel – vom Vertex-Attribut zur finalen Fragment-Farbe (Aufgabe z.B. erklären / ergänzen / zuordnen / Fehler finden)
- **Was ist Tessellierung? Aus welchen Primitiven setzt sich in WebGL ein Rechteck / ein Würfel zusammen?**

- **Besonderheiten von Javascript** (keine Deklarationen; wenige Typen; Typprüfung zur Laufzeit; keine Klassen; Objekte können zur Laufzeit um Methoden und Attribute erweitert werden;)
- **In was und wie ist WebGL eingebettet?**
- **Was passiert mit Shader-Sourcecode in einem WebGL-Programm? Was ist das Kontext-Objekt, was ist das Programm-Objekt?**

- **Wie ist ein WebGL-Programm prinzipiell aufgebaut: Kontext erzeugen, Programm erzeugen, Buffer erzeugen, Szene malen.**
- Was sind **die beiden wesentlichen Dinge**, die die Applikation beim Malen einer Szene tun muss? 1. Zustandsvariablen setzen; 2. Geometrie malen / draw()-Befehle aufrufen.
- **Unterschiede attribute - uniform - varying**
- **Was ist ein Vertex Buffer Object?**
- Wie setzt man den Wert einer Uniform-Variablen von der Applikation aus? (zwei Schritte: Location bestimmen, Wert an dieser Location setzen)
- Wieviele verschiedene WebGL-Befehle zum Malen von Geometrie haben wir kennengelernt? Wie lauten sie? Wie funktionieren sie?
- Welche Geometrie-Primitiven gibt es bei WebGL (Point, Line, Triangles, TriangleStrip, TriangleFan) – **wurde in VL nicht so ausführlich erklärt.**
- **Was ist der Unterschied zwischen einem Aufruf mit TRIANGLES und TRIANGLE_STRIP?**
- **Wird nicht abgefragt: sonstige WebGL-Befehle.**

- **Input und Output eines Vertex-Shaders; was tut ein Vertex-Shader?**
- **Input und Output eines Fragment-Shaders; wie funktioniert ein Fragment-Shader?**
- **gl_Position**
- **gl_FragColor**
- Was macht die folgende GLSL-Codezeile? In welchem Shader-Type wäre diese Zeile erlaubt? (z.B.: uniform sampler2D dayTex; oder attribute vec3 vNormal; oder gl_Position = mvMatrix * vPosition;)
- Wofür könnte das ec in varying vec3 ecPosition stehen?
- Was macht folgendes Shader-Paar:
- **Wird nicht abgefragt: sonstige Sprachelemente / exakte Syntax der GL Shading Language**

- Wie realisiere ich Phong-Beleuchtung mittels Shadern? Welche Input-Daten benötige ich, wie strukturiere ich die?
- **Unterschiede Vertex-/Fragment-Beleuchtung?**
- Verschiedene Lichtquellen-Typen; wodurch wird ein Spotlight definiert / charakterisiert?
- Was ist der Unterschied zwischen Phong und Blinn-Phong? **Wie heißt der Vektor, der bei Blinn-Phong eine wichtige Rolle spielt?** Wie berechnet man ihn?

- **Welche Daten muß ein Kugel-Objekt besitzen / an OpenGL senden, damit eine beleuchtete und texturierte Kugel gemalt werden kann?**
- **Wie wird eine parametrische Fläche spezifiziert?**
- **Was ist der Unterschied zwischen Objektraum, Parameter-Raum und Texturraum?**
- **Wie rechne ich vom Parameter-Raum in den Texturraum um? (z.B. einfache Umrechnungs-Formel vervollständigen)**
- Welche Textur-Wrapping-Modi gibt es, welcher macht was?
- Was machen die Befehle `glBindTexture()`, `glTexImage2D()`?
- Wie funktioniert Multi-Texturing?
- **Was ist Texture Minification / Magnification? Probleme? Mip-Map?**

- **Was ist eine Cube Map?**
- **Was ist Environment Mapping?**
- **Was ist ein Höhenfeld, was ist eine Bump Map?**
- **Was sind Tangente und Bitangente eines Vertex?**
- **Was sind Unterschiede zwischen Shadow Mapping und Shadow Volumes?**
- **Was macht ein Stencil Buffer?**
- **Was bedeutet Multi Pass Rendering?**