CSE167
Introduction to Computer Graphics

Jürgen P. Schulze, Ph.D.
University of California, San Diego
Fall Quarter 2009
Course staff

Instructor

• Jürgen Schulze (jschulze@ucsd.edu), Research Scientist at Calit2

Teaching Assistants

• Jason Kimball (jkimball@ucsd.edu), CSE graduate student

• Daniel Tenedorio (dtenedor@ucsd.edu), CSE graduate student
Today

- Course organization
- Course overview
Today

• Course organization

• Course overview
Course organization

Lecture

- Tue/Thu, 2:00pm-3:20pm, HSS 1330

Lab hours

- Wed Sept 30, 1pm-4pm
- Future hours TBA in class and on web site
Course organization

Class web site

- [http://graphics.ucsd.edu/twiki/bin/view.pl/Classes/CSE167Fall2009](http://graphics.ucsd.edu/twiki/bin/view.pl/Classes/CSE167Fall2009)
- Schedule, slides, reading, homework, project descriptions, grading, etc.
Course organization

WebCT

- Go to 
  http://webct.ucsd.edu
  and select CSE167
- Log in with your Active Directory account
- Find discussion board, chat, etc.
Textbooks


Programming Projects

- Find assignments and schedule on class website
- Base code (for Windows and Linux) and documentation on class webpage
- Use EBU3B 2xx labs or your own PC
- Individual assistance by TAs during lab hours
- Turn in by demonstration to TA during lab hours or homework grading hours
Programming Projects

Build your own 3D rendering engine

• **Project 1**: Matrices, Vectors, and Coordinate Transformations

• **Project 2**: Interactive Viewing

• **Project 3**: Rasterization

• **Project 4**: Lighting and Texturing

• **Project 5**: Scene Graphs

• **Project 6**: Shader Programming

• **Final Project**
Tests

Two in-class written tests.

Closed book, single sided handwritten index card size sheet is permitted.

Midterm exam:

• Thu 10/29, 2:00pm-3:20pm, HSS 1330

Final exam:

• Thu 12/03, 2:00pm-3:20pm, HSS 1330
Grading

- Homework Projects 1-6: 10% each
- Midterm exam: 10%
- Final exam: 15%
- Final project: 15%

Late submission policy for homework projects:

75% of original grade if you present your project within seven days of the due date
**Prerequisites**

Basic familiarity with

- Linear algebra
- C++ (if you know Java you’ll be able to adapt)
- Object oriented programming
Questions?
Today

• Course organization

• Course overview
What is computer graphics

Applications

• Movie, TV special effects
• Video games
• Scientific visualization
• GIS (Geographic Information Systems)
• Medical visualization
• Industrial design
• Simulation
• Communication
• Etc.
What is computer graphics?

- Rendering
- Modeling
- Animation
Rendering

- Synthesis of a 2D image from a 3D scene description
  - Rendering algorithm interprets data structures that represent the scene in terms of geometric primitives, textures, and lights
- 2D image is an array of pixels
  - Red, green, blue values for each pixel
- Different objectives
  - Photorealistic
  - Interactive
  - Artistic
Photorealistic rendering

- Physically-based simulation of light, camera
- Shadows, realistic illumination, multiple light bounces
- Slow, minutes to hours per image
- Special effects, movies
- CSE168: Rendering Algorithms
Photorealistic rendering
Interactive rendering

- Produce images within milliseconds
- Using specialized hardware, graphics processing units (GPUs)
- Standardized APIs (OpenGL, DirectX)
- Often “as photorealistic as possible”
- Hard shadows, fake soft shadows, only single bounce of light
- Games
- CSE167
Interactive rendering
Artistic rendering

- Stylized
- Artwork, illustrations, data visualization
Modeling

• Creating 3D geometric data
  - The “model” or the “scene”
• By hand
  - Autodesk (Maya, AutoCAD), LightWave 3D, ...
• Free software
  - Blender
• Not as easy to use as Notepad...
Modeling

- Basic 3D models consist of array of triangles
- Each triangle stores 3 vertices
- Each vertex contains
  - xyz position
  - Color
  - Etc.
Modeling

- Procedural: by writing programs
- Scanning real-world objects
Animation

- Deforming or editing the data
- Change over time
- Faces, articulated characters, ...
- CSE169: Computer Animation
Animation
Physics simulation
In this class

The Basics...

• Rendering 3D models
  - Camera simulation
  - Interactive viewing
  - Lighting, shading

• Modeling
  - Triangle meshes
  - Smooth surfaces

• Experience with linear algebra, C++, OpenGL

• Background for advanced topics (CSE168, CSE169)
Questions?
Next Lecture

- Tue 9/29 at 2pm
- Vectors, matrices and transformations
- Lab session next Wed at 1pm: introduction to the base code by Jason Kimball
  - If you can’t find Jason in lab 250, make sure to check in all labs EBU3B 2xx