

The Art of Graphics Programming

Course Syllabus

Fall 2012

Patrick Hebron
patrick.hebron@nyu.edu

https://github.com/Hebali/AoGP_Fall_2012

Weekly Assignment:

Part 1: Create a software-based piece (an artwork, visualization, tool, etc) that utilizes only the platform components we have developed thus far. EMBRACE THE CONSTRAINT. Then, add one (or maybe a few) additional functions of your own that will enhance the piece in some way. You can combine functions or create new ones if you like. But you can't, for example, add a camera if we haven't yet covered them in class.

Part 2: Think about the aesthetic constraints that effect the pieces you created and critique the platform in its current state. What are some additional features that would enhance your creative output and the process of creating it?

Part 3: Look at the next week's syllabus topics. Begin to think about where these pieces should fit into the platform architecture. Create a diagram of the proposed architectural additions.

Class Sessions:

Week 1: September 6

Course Introduction: *The Evolution of the Medium*

- Historical Overview
- Aesthetic Decisions, Complete Representation and "The Myth of Total Cinema"
- Design Tools, Ahistorical Progress and *Piano Tops*

Technical Context

- Basic Anatomy: CPUs, GPUs, Programming Languages & APIs
- From `main()` to Processing: Basic Architectural Patterns in Graphics Software

Building a 3D Graphics Platform

- Developing the Project Roadmap
- Implementing a Scenegraph Node

Week 2: September 20

Building a 3D Graphics Platform

- Drawing Primitives and Custom Polygons
- Creating, Storing and Manipulating Complex Geometries
- Implementing a Generic Geometry Container within the Scenegraph Model

Week 3: October 4

Building a 3D Graphics Platform

- Working with Surface and Vertex Normals
- Working with Textures and UV Coordinates
- Optimized Rendering and GPU Storage

Theoretical Context

- Figurative Abstraction in Painting, Literature and Computer Graphics

Week 4: October 18

Building a 3D Graphics Platform

- Implementing Animation Properties
- Animating Geometries
- Overview of Cameras and Perspective Models in Computer Graphics
- Implementing Cameras

Week 5: November 1

Building a 3D Graphics Platform

- Overview of Lighting Models in Computer Graphics
- Implementing Lights
- Animating Cameras and Lights

Theoretical Context

- Bending the Laws of Perspective and Optics in Computer Graphics

Week 6: November 15

Building a 3D Graphics Platform

- Introducing Shaders
- Integrating GLSL Shaders into the Graphics Pipeline
- Manipulating Scenes with Vertex Shaders - Part I
- Manipulating Surfaces with Fragment Shaders - Part I

Week 7: November 29

Building a 3D Graphics Platform

- Manipulating Scenes with Vertex Shaders - Part II
- Manipulating Surfaces with Fragment Shaders - Part II

Building a 3D Graphics Platform in C++

- The Differences Between Java and C++
- The Differences Between Processing, oF and Cinder
- Overview of the C++ 3D Graphics Platform Code Samples