Computer Graphics
50:198:456/56:198:556 (Fall 2017)

Handout:	**1**	**Professor:**	**Suneeta Ramaswami**
Date:	**9/6/17**	**E-mail:**	`suneeta.ramaswami@rutgers.edu`
Office:	**323 BSB**	**URL:**	`http://crab.rutgers.edu/~rsuneeta`
		Phone:	**(856)-225-6439**

Course Outline

Course Summary:

This is an introductory course on computer graphics, in which you will learn about the mathematical foundations of the field, and also gain significant experience in graphics programming.

The goal of this course is to thoroughly understand the fundamentals of graphical systems and, at the same time, develop sophisticated graphics programs by using a powerful programming package. In particular, you will use OpenGL, a platform-independent state-of-the-art graphics library, and the industry standard for graphics programming. Since OpenGL is platform-independent, an interface with the windowing system is needed to display the results of the graphics program: GLFW (Graphics Library Framework) is a toolkit that provides window management and user-interaction facilities. Therefore, you will use GLFW along with OpenGL. **Note:** The textbook uses a library called GLUT instead of GLFW. However, GLUT does not work with OpenGL 3.2 and above on Mac OSX, whereas GLFW is a more modern, lightweight library that works on Linux, Mac OSX, as well as Windows.

The topics that will be covered during the course are the following (a more detailed schedule of topics to be covered appears at the end of this handout): Graphics systems and imaging principles, graphics programming using OpenGL and GLFW, input devices and interactive techniques, geometric transformations and modeling in two and three dimensions, viewing in 2D and 3D, lighting and shading, and fundamental graphics algorithms (clipping, hidden surface removal, scan conversion and anti-aliasing). Time permitting, we will also cover selected topics on curves and surfaces, and solid modeling. To understand the theoretical material in this course, you will need some background in discrete mathematics, data structures and basic algorithms. While we will learn the fundamentals of OpenGL and GLFW, we won't have time to cover all the features in great detail in class. It is expected that you will explore these libraries on your own as well.

All course material such as the syllabus, programs covered during lectures, homework assignments, and course announcements will be available on the Sakai site for this course. You are expected to check this site regularly. The course webpage contains some useful and interesting course-related links. In particular, links to the GLFW documentation and a free online OpenGL book can be found there: `http://crab.rutgers.edu/~rsuneeta/cs456/`.

Pre-requisites:

Programming with Data Structures (50:198:213). **It will be assumed** that you can program competently with data structures in C or C++. Please see me if you do not have the necessary pre-requisites.

Text Book and Useful References:

1. Required: *Interactive Computer Graphics: A Top-down Approach with Shader-Based OpenGL*, by Edward Angel and Dave Shreiner, **6th Edition**, Addison-Wesley (ISBN-10 0-13-254523-3).
2. A free online OpenGL book: openglbook.com
3. For all information related to GLFW: www.glfw.org.

Office hours:

Mondays and Wednesdays 2:00-3:30pm, or by appointment at a mutually agreeable time.

Coursework:

1. Five Programming Assignments, worth a total of 50% (4% for #1, 10% for #2, 12% for #3, 12% for #4, and 12% for #5)
2. Two Written Assignments, worth 5% total
3. In-class midterm, worth 20% (tentatively to be held in class on Monday, October 30, 2017).
4. Final Exam, worth 25% (to be held on Monday, December 18, 11:30am-2:20pm).

The written assignments cover mathematical aspects of the course. The programming assignments cover significant topics in the course and require you to use OpenGL.

Tentative dates for Assignments:

Prog #1:	Out 9/12, Due 9/27	Prog #2:	Out 9/27, Due 10/13
Prog #3:	Out 10/13, Due 11/3	Prog #4:	Out 11/3, Due 11/22
Prog #5:	Out 11/22, Due 12/13		
Written #1:	Out 10/9, Due 10/23	Written #2:	Out 11/20, Due 12/6

Grading Policy:

General discussion of homework assignments with fellow students is fine. However, *all work must be done independently.* Please respect this policy - this will allow each of you to receive full credit for the creativity I hope to see in your assignments (particularly the programming ones), while also removing undue burden from your performance on the exams.

Please make sure that your homework assignments (compile and) run successfully before submitting them. Homework assignments **must** be submitted on the due date. Late hand-ins will be subject to a 50% deduction per day after the due date.

Tentative Schedule of Topics to be Covered

(On next page)

Dates	Topics	Reading
9/6-9/13	**Overview of Graphics Systems** - Applications, Interactive Graphics - Hardware/Software, Graphics Architecture - Imaging Principles, Pinhole camera model - Graphics Pipeline	Ch. 1 of Angel & Shreiner
9/18-9/27	**Graphics Programming** - Introduction to OpenGL - Introduction to GLFW - Vertex and Fragment Shaders - Interaction, Events, Menus - Sample Programs	Ch. 2 of Angel & Shreiner GLFW documentation
10/2-10/9	**Geometry Fundamentals** - Elements of geometry required for computer graphics - Affine geometry - Coordinate frames	class notes
10/11-10/18	**Geometric Objects and Transformations** - Matrix Representation of 2D and 3D Transformations - Translation, Rotation and Scaling - Homogeneous Coordinates - Concatenation of Transformations OpenGL Implementation of Transformations, Sample Program	Ch. 3 of Angel & Shreiner
10/23-10/25	**Viewing** - Camera Positioning - Parallel and Perspective Projections	Ch. 4 of Angel & Shreiner
10/30	**Midterm Exam**	All material covered upto 10/23
11/1-11/6	**Viewing** (cont'd) - OpenGL Implementation - Perspective Projection Matrices - Normalization, Sample Program	Ch. 4 of Angel & Shreiner
11/8-11/13	**Programmable Shaders** - Legacy vs Modern OpenGL	class notes
11/15-11/29	**Lighting and Shading** - Light sources, Illumination models - Phong Reflection Model - Computation of Vectors - Flat/Gouraud/Phong Shading - Implementing a lighting model in OpenGL, Sample Program	Ch. 5 of Angel & Shreiner
12/4-12/13	**From Vertices to Fragments** - Clipping in 2D & 3D - Rasterization - Hidden-surface Removal - Scan Conversion and Anti-aliasing	Ch. 6 of Angel & Shreiner