[CSCI 420] Computer Graphics

Fall 2020

Lectures:	Tue 3:30pm-6:50pm
Classroom:	Online/Zoom
Instructor:	Andy Nealen
Office:	Online
Office Hours:	Tuesday, 1:00pm-2:30pm, via Zoom by email appointment
E-mail:	nealen@usc.edu

TA: Cho-Ying Wu (choyingw@usc.edu). **Office hours**: Tuesday 9:00am-10:30am

Syllabus: This course is an introduction to three-dimensional computer graphics. Students will learn both theory of 3D computer graphics, and how to program it efficiently using OpenGL. Topics include 2D and 3D transformations, Bézier and B-Spline curves for geometric modeling, interactive 3D graphics programing, computer animation, kinematics, and rendering including ray tracing, shading, and lighting. There will be an emphasis on the mathematical and geometric aspects of computer graphics, This course is regularly offered every semester (the instructor may vary as well as the content). There will be 3 hours of lecture (by instructor), and 1 hour of discussion/office hour (by instructor or TA) every week.

Target Audience: This course is intended for students interested in the basics and math of computer generated imagery using OpenGL. It involves a lot of programming, and requires a certain degree of mathematical sophistication (in linear algebra, specifically). But it's also a lot of fun. No artistic skill is required, but it does come in handy.

Prerequisites: CSCI 104 and 1 from (MATH 225 or (EE 141 and MATH 126) or (EE 141 and MATH 127) or (EE 141 and MATH 129)). In other words, knowledge of C++ programming, and a familiarity with Linear Algebra and Calculus.

The most important prerequisite of all, however, is your interest in the class, motivation, and commitment to learning. If you are not sure whether this class is for you, come and talk to us

Readings: This course is based on the following textbooks:

Edward Angel, **Interactive Computer Graphics: A Top-Down Approach Using OpenGL**, Fifth edition, Addison Wesley, ISBN-10: 0321535863

Dave Shreiner, **OpenGL Programming Guide: The Official Guide to Learning OpenGL**, Versions 3.0 and 3.1, Seventh edition, Publisher: Addison-Wesley Professional, ISBN: 9780321552624

Lectures: Lecture attendance, or watching the recorded lecture is strongly encouraged; please do not take the course if you cannot make the time to keep up with and follow the lectures. If you miss a class, it is your responsibility to find out what we discussed in class, including which announcements we made in class. If there is something that you do not understand, feel free to interrupt with questions.Your active participation in class is crucial in making the class successful.

Use your fellow students as a resource, e.g., by forming study groups or posting questions on the discussion forum on Piazza that the TA and Graders monitor on a daily basis. We encourage you to participate actively on the discussion forum, by both asking and answering questions. If you need

additional help, please feel free to meet us during office hours. The TA and Graders are experienced and will be able to answer all of your technical questions, including about the textbook, lectures, and projects.

Assignments: Students will design, implement, and use interactive graphical applications (using the OpenGL API in C++). This amounts to three coding assignments using OpenGL and the OpenGL Shading Language (GLSL). **All assignments must be completed to pass the course**. The assignments have bonus credits.

Exams: There will be one midterm and one final, all of which are mandatory. The exams will be based on the material covered in class, and on what is learned from completing the assignments. The exam dates are listed on the schedule; please do not take the course if you cannot attend the exams. No makeups will be given. All exams will be comprehensive but with a focus on material not yet tested in a previous exam.

Grades: Assignments and exams have the following weights:

- Assignment 1: 16%
- Assignment 2: 17%
- Assignment 3: 17%
- Midterm Exam: 20%
- Final Exam: 30%

The instructors reserve the right to adjust the grading scale. There will always be some students who are very close to grade boundaries. There is nothing that we will do about that. Grades are based on performance, not need or personal circumstances, and the instructor does not negotiate grades. Thus, do not take CSCI420 (or take it at your own risk) if you need a certain grade, for example, because you are graduating or because you have been conditionally admitted.

To receive a good grade, you will need to perform well in the exams **and** the assignments. Please check the correctness of the grading and the posted scores immediately after we announce the availability of the scores. **You will need to let us know about any grading issue within 7 days of us posting the score**. After that time, we will no longer respond to requests for changes to your score. If you have a grading issue, you will need to discuss the issue first with the TA. If you cannot reach consensus, you can appeal the grading issue to the instructor.

Late Policy: Projects handed in up to 24 hours late are worth 50%. After that, they will not be given any credit. Of course, exceptions will be made in extreme circumstances (let us know in advance).

Statement for Students with Disabilities: Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to me (or to TA) as early in the semester as possible. DSP is located in STU 301 and is open 8:30 a.m.–5:00 p.m., Monday through Friday. The phone number for DSP is (213) 740-0776.

Tentative Schedule: The instructor may adjust this schedule during the semester

Week	Date	Topics
1	8/18	Introduction Basic Graphics Programming
2	8/25	Input and Interaction Transformations
3	9/1	Viewing and Projection Hierarchical Modeling
	9/4	Assignment 1 Out
4	9/8	Polygonal Meshes, Curves, and Surfaces Splines
5	9/15	Lighting and Shading Shading in OpenGL + Assignment 1 Q&A
6	9/22	Texture Mapping Bump Mapping and Clipping
	9/25	Assignment 1 Due
	9/25	Assignment 2 Out
7	9/29	Rasterization Review for Midterm
8	10/6	Midterm Exam
9	10/13	Ray Tracing Geometric Queries
	10/16	Assignment 2 Due
	10/16	Assignment 3 Out
10	10/20	Spatial Data Structures Global Illumination
11	10/27	Animation Principles Computer Animation
12	11/3	Quaternions and Rotations Programmable Graphics Hardware
	11/6	Assignment 3 Due
13	11/10	Physically-Based Simulation Review for Finals
	11/19	**Final Exam** (2pm-4pm)

Statement on Academic Integrity: USC seeks to maintain an optimal learning environment. General principles of academic honesty include the concept of respect for the intellectual property of others, the expectation that individual work will be submitted unless otherwise allowed by an instructor, and the obligations both to protect one's own academic work from misuse by others as well as to avoid using another's work as one's own. All students are expected to understand and abide by these principles. *We will strictly enforce the student conduct code and refer students to the Office of Student Judicial Affairs and Community Standards for further review, should there be any suspicion of academic dishonesty, and suggest that they follow the recommended sanctions in case they should find that there was academic dishonesty*. We typically suggest an F as overall class grade as penalty, if asked. Scampus, the Student Guidebook, contains the student conduct code and the academic review process: `https://policy.usc.edu/scampus-part-b/`.

Problems and Concerns: At some point, you will have questions. For example, you might not be able to get code to run, there might be something in the textbook that you do not understand, and so on. In this case, we encourage you to post the question on the discussion forum and see whether someone can help you. If this approach does not generate the desired result, then the TAs and CPs will be happy to help you in person during their office hours. TAs do answer email but, unfortunately, often will not manage to answer it on the same day. (Sometimes, they will be out of town and it will take them even longer. Also, they are typically overloaded with questions on exam days or directly before.) It is very important to us that you voice your concerns about any aspect of the class as soon as they arise. Please send an e-mail to the instructors or talk to us in person.

Computer Graphics is a fun topic, and we hope that all of us will have lots of fun!

Title IX: Harassment, sexual misconduct, interpersonal violence, and stalking are not tolerated by the university. All faculty and most staff are considered Responsible Employees by the university and must forward all information they receive about these types of situations to the Title IX Coordinator. The Title IX Coordinator is responsible for assisting students with supportive accommodations, including academic accommodations, as well as investigating these incidents if the reporting student wants an investigation. The Title IX office is also responsible for coordinating supportive measures for transgender and nonbinary students such as faculty notifications, and more. If you need supportive accommodations you may contact the Title IX Coordinator directly (titleix@usc.edu or 213-821-8298) without sharing any personal information with me. If you would like to speak with a confidential counselor, Relationship and Sexual Violence Prevention Services (RSVP) provides 24/7 confidential support for students (213-740-9355 (WELL); press 0 after hours).

Emergency Preparedness/Course Continuity in a Crisis/during COVID-19: In case of a declared emergency if travel to campus is not feasible, USC executive leadership will announce an electronic way for instructors to teach students in their residence halls or homes using a combination of Blackboard, teleconferencing, and other technologies. Please activate your course in Blackboard with access to the course syllabus. Whether or not you use Blackboard regularly, these preparations will be crucial in an emergency. USC's Blackboard learning management system and support information is available at blackboard.usc.edu.

Resources:

Textbook support by the author, Edward Angel: https://www.cs.unm.edu/~angel/

Real-Time Rendering Resources: http://www.realtimerendering.com/

OpenGL homepage: www.opengl.org

OpenGL Red Book online, Version 1.1: http://www.glprogramming.com/red/

OpenGL SDK documentation:
https://www.khronos.org/registry/OpenGL-Refpages/

OpenGL tutors for windows by Nate Robins:
http://users.polytech.unice.fr/~buffa/cours/synthese_image/DOCS/www.xmission.com/Nate/tutors.html

Supplemental Books:

Steven J. Gortler. Foundations of 3D Computer Graphics. First Edition, The MIT Press, 2012, ISBN-10: 0262017350, ISBN-13: 978-0262017350.

Real-Time Rendering. Thomas Akenine-Möller and Eric Haines. 2nd edition, AK Peters, 2002. ISBN 1-56881-182-9.

Computer Graphics: Principles and Practice. James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. 2nd edition in C, Addison-Wesley, 1996. ISBN 0201848406.

Fundamentals of Computer Graphics. Peter Shirley, Steve Marschner. 3rd edition, A K Peters, 2009. ISBN 1568814690.

An Introduction to Ray Tracing. Andrew S. Glassner, editor, Academic Press, 1989. ISBN 0-12-286160-4.

Advanced Animation and Rendering Techniques, Theory and Practice. Alan Watt and Mark Watt, ACM Press and Addison-Wesley, 1992. ISBN 0-201-54412-1.

Pre-requisite-related Resources:

Introduction to Linear Algebra. Gilbert Strang. Wellesley-Cambridge Press, 1998. ISBN 0-9614088-5-5.

Gilbert Strang's Linear Algebra class on MIT Open Courseware: https://ocw.mit.edu/courses/mathematics/18-06-linear-algebra-spring-2010/video-lectures/

(Math for graphics) Appendices B and C of the course textbook (Angel)

(Math for graphics) Chapter 2, "Miscellaneous Math" of Shirley and Marschner, Fundamentals of Computer Graphics

Calculus: Early Transcendentals. James Stewart. Fourth edition, Brooks/Cole Publishing Company, 1999. ISBN 0-534-36298-2.

Khan Academy, free math instructional videos: https://www.khanacademy.org/

The C Programming Language. Brian W. Kernighan and Dennis M. Ritchie. Second edition, Prentice Hall, 1988. ISBN 0-13-110362-8 (paperback).

Acknowledgment: I would like to thank Hao Li, Jernej Barbič, and Sathyanaraya Raghavachary for providing the lecture material. This lecture is based on the computer graphics courses taught by Frank Pfenning and Jessica Hodgins from CMU. This course has also been influenced by computer graphics courses at Cornell, MIT, UC Berkeley, EPFL, and Stanford.

$$x = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$x' = \begin{bmatrix} -2 \\ 2 \\ 6 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$\mathbf{n} \cdot (\mathbf{x} - \mathbf{p}) = 0$ (orthogonality)

$\mathbf{n} \cdot \mathbf{x} - \mathbf{n} \cdot \mathbf{p} = 0$

$2(\mathbf{l} \cdot \mathbf{n})\mathbf{n} - \mathbf{l}$

$$\begin{bmatrix} \sqrt{0.5} & 0 & -\sqrt{0.5} \\ 0 & -1 & 0 \\ \sqrt{0.5} & 0 & \sqrt{0.5} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$\begin{bmatrix} \sqrt{0.5} & 0 & \sqrt{0.5} \\ 0 & -1 & 0 \\ -\sqrt{0.5} & 0 & \sqrt{0.5} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$