CS 4300/CS 5310: Computer Graphics

Instructor: Dr. Amit Shesh
Office: Nightingale 132C
Email: ashesh [at] ccs.neu.edu
Office hours: Tues 10am-12 noon, Thurs 10am-11:30am **or by appointment**

Course Details:

Teaching Assistant: David Heyman
TA email: heyman.d [at] husky.neu.edu
TA Office hours: Mon and Wed 2pm-4pm
TA Office room: 164 WVH

Teaching Assistant: Nishant Anupam Gupta
TA email: gupta.nisha [at] husky.neu.edu
TA Office hours: Wed 4:30pm-8:30pm, Thurs 5pm-7pm, Fri 5pm-6pm
TA Office room: 164 WVH

Credit hours: 4
Class schedule: TF 3:25pm-5:05pm
Classroom: Shillman 425
Required text:

1. *Fundamentals of Computer graphics* by Peter Shirley and Steve Marschner by AK Peters.

Recommended text:

1. *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.3 (8th Edition)* by Shreiner et al. by Addison-Wesley
2. OpenGL 4 reference pages: http://www.opengl.org/sdk/docs/man/
3. OpenGL 4 tutorials: http://www.opengl-tutorial.org/
4. OpenGL 4 online book (incomplete, but a good start): http://openglbook.com/the-book/
5. JOGL: http://jogamp.org/jogl/www/
6. JOML: https://github.com/JOML-CI/JOML
7. GLM: http://glm.g-truc.net/0.9.7/api/index.html

Piazza: https://piazza.com/class/j796lmm7y2wsg

Course Description

Welcome to CS 4300/CS 5310! In this course, we will study the fundamentals of computer graphics. We will study primitive 2D graphics and related algorithms. We will also study how various kinds of 3D models are created for a variety of applications. We will then examine the graphics "rendering pipeline" in detail, focussing on 2D/3D modeling, viewing and projective transformations and their matrix implementations, rendering and visibility algorithms. We will also study the technique of ray-tracing that is known to produce some of the most stunning pictures in computer graphics.

Computer graphics is as much engineering as it is math and science, and throughout this course we will see examples of how mathematics governs almost everything in computer graphics, and how it is judiciously compromised to provide visually pleasing results at far less computational costs to make interactive graphics applications a reality.

We will use OpenGL 3.3+ (shader-based OpenGL) in this course. You will have a choice between Java and C++ for all the programming components of this course and will write shader programs in GLSL. This course is intended for undergraduate and graduate students who want to learn the basics of computer graphics and familiarize themselves with its concepts in breadth rather than depth.

Course Goals

The goals of this course are to:

- Familiarize students with the conventional rendering pipeline in computer graphics and its various components
- Familiarize students with various algorithms related to modeling and rendering in computer graphics
- Provide hands-on experience in writing simple and complex computer graphics applications
- Familiarize students with shader programming
- Develop an understanding of the science, mathematics and art behind practical computer graphics

Course Outcomes

After taking this course, students will be able to:

- Write simple as well as fairly complex graphics applications and shaders using OpenGL 3.3+ and GLSL with Java and/or C++ that perform 2D and 3D interactive graphics.
- Explain the overall process of displaying static and dynamic graphical content on a screen
- Explain and implement several algorithms related to computer graphics
- Implement various processes and graphical operations such as modeling, viewing and rendering in 2D and 3D
- Use vector and matrix algebra, coordinate and solid geometry and physics to solve problems in computer graphics
- Implement a ray tracing application

Course Expectations

Pre-requisites

All of the pre-requisites below must be met:

1. **One of** CS 1500, CS 2510, CS 3500 with a minimum grade of D-
2. **One of** MATH 1260, MATH 2331, MATH 2341 with a minimum grade of D-
3. **Graduate students:** Knowledge of linear algebra

Expected skill sets (as a result of the above courses): At least one semester's worth of programming experience in either Java or C++, basic knowledge of linear algebra (vectors, matrices, etc. most probably at a high-school level) and some physics. Prior exposure to calculus is a plus.

I will not spend any class time teaching Java or C++, except the first couple of examples to explain the structure and library/IDE usage. **No prior knowledge of OpenGL is required.** A short overview of the relevant Math and physics will be provided in class wherever appropriate. Above all, a high level of curiosity and excitement about graphics is essential!

Graphics hardware

We will use OpenGL 3.3 and higher for all the assignments in this course. This requires that you have a computer with a graphics card that supports OpenGL 3.3 or higher. If you are planning to use your own computer to complete assignments, please confirm that it has the necessary GPU and drivers. You will be given a demo program to test your machine. Labs 210 and 212 in the West Village H building are equipped with the necessary hardware.

Role of the instructor

My primary method of teaching for this course will be in the form of lectures and programming examples. I will regularly demonstrate several programs as part of my lectures, so you are highly encouraged to attend all lectures to gain full benefit from this course. Apart from office hours, you can talk to me about any course-related issues before or after each class. I am also available for students outside class (by appointment) to discuss any individual course-related issues or brainstorm ideas about computer graphics. **I will try my best to be available to help you succeed in this course, but I have to see you try just as much to learn.**

Course Details

Assignments and Grading

The grade is divided as follows:

1 small programming assignment	8%
1 medium programming assignment	13%
2 group programming assignments	52%
Quizzes	5%
Midterm exam	10%
Final exam	12%

Programming assignments will usually ask you to implement the concepts that you have learnt about in class. Many assignments will be done in groups of 2 students. Unless otherwise specified in the assignment descriptions, the efficiency of your program will normally not be a criterion for grading. The exams will ask you to apply learned concepts to problems that may not have been directly discussed in class.

Upon my discretion, one or more assignments *may* contain some amount of extra credit to make up for any points lost in previous assignments. **Please do not bank on such a chance, the course may end up having no extra credit at all.**

For undergraduate students, grades will be awarded as follows:

Undergraduate		Graduate	
$\geq$ 93% and $\leq$100%	A	$\geq$ 93% and <100%	A
$\geq$ 90% and $\leq$92.9%	A-	$\geq$ 90% and $\leq$92.9%	A-
$\geq$ 87% and $\leq$89.9%	B+	$\geq$ 87% and $\leq$89.9%	B+
$\geq$ 83% and $\leq$86.9%	B	$\geq$ 83% and $\leq$86.9%	B
$\geq$ 80% and $\leq$82.9%	B-	$\geq$ 80% and $\leq$82.9%	B-
$\geq$ 77% and $\leq$79.9%	C+	$\geq$ 77% and $\leq$79.9%	C+
$\geq$ 73% and $\leq$76.9%	C	$\geq$ 73% and $\leq$76.9%	C
$\geq$ 70% and $\leq$72.9%	C-	$\geq$ 70% and $\leq$72.9%	C-
$\geq$ 67% and $\leq$69.9%	D+		
$\geq$ 63% and $\leq$66.9%	D		
$\geq$ 60% and $\leq$62.9%	D-		
$\geq$ 0% and $\leq$59.9%	F	$\geq$ 0% and $\leq$69.9%	F

Grading will be on an absolute basis, which means your final letter grade will depend only on your performance. It is possible that I will move the B-C and/or C-D boundaries by a small amount, but the criterion to get an A and A- is strict. Your overall weighted score will be rounded to the nearest 10th of a percent.

All assignments are to be submitted on time, and unless specified otherwise, all assignments will be due at 8:59pm on the due date. **There is no grace period for any assignment.** Assignments that are submitted late will not be graded. Under extreme circumstances, I will allow an assignment to be submitted later than required. However you must meet me personally to explain the circumstances and take my permission well before the actual deadline. I will evaluate this on a case-by-case basis.

Class policies

Cheating/Plagiarism

Cheating helps no one, and if caught, will cause you only trouble. **All work for this course has to be completed individually, except the group assignments.** You are allowed to discuss only about what must be achieved in an assignment, not how. Sharing code with another student, helping another student to write code, receiving code from another student only to modify it yourself, sharing written work in any form are all forms of cheating that are equal to each other in seriousness. For group assignments, you are allowed to share code only with your group partner. You are not allowed to look at code of another student (other than your group partner) under any circumstances. You can look at articles or code online, but do not directly use it as part of your project even if it is licensed to be used that way. If you are "inspired" from articles or code that you found online, please cite it in your code clearly at the top of the file that contains it. Use of anything that wasn't your brainchild and is not cited is plagiarism. A reference should be detailed enough for me to access it to the extent that you were able to. If you are unsure about a particular method of working, **don't assume it is legitimate, check with me.** If you need help I am available to answer your questions and work with you. I have the liberty to ask you questions about your submitted work to verify that you have indeed worked on it by yourself. Penalties for cheating will range from a zero for the entire assignment in question to an "F" for the course. Irrespective of the size of the offense, you will be officially reported and the incident will go on your academic record.

Please review the Northeastern University policy on academic integrity: `http://www.northeastern.edu/osccr/academic-integrity-policy/`.

Student Conduct

Attendance for this course is highly recommended if you wish to gain maximum benefit from this course. Students attending the lectures are expected to maintain a positive learning environment. Students whose behavior is disruptive either to the instructor or to other students will be asked to leave. **Laptops:** Laptops are allowed in class. However please confine its use to course-related material, and try not to distract others by the loud taps on your keyboard.

Accommodations for Students With Disabilities

If you have a disability-related need for reasonable academic accommodations in this course and have not yet met with a Disability Specialist, please visit www.northeastern.edu/drc and follow the outlined procedure to request services. After the Disability Resource Center has approved you for an academic accommodation in this class, please present your "Professor Notification Letter" to the instructor, ideally during the first week of the semester, so that we can address your specific needs and set up appropriate accommodations as early as possible.

End-of-Course Evaluation Surveys (TRACE)

Your feedback regarding your educational experience in this class is very important to the CCIS. Your comments will make a difference in the future planning and presentation of our curriculum.

TRACE (Teacher Rating and Course Evaluation) is a required part of every course. Your participation is needed and encouraged, as it is one way to enhance the quality of the course. Your voice matters!

At the end of this course, please take the time to complete the evaluation survey at `https://neu.evaluationkit.com`. Your survey responses are completely anonymous and confidential. For this class, surveys will be open for two weeks. An email will be sent to your HuskyMail account notifying you when surveys are available.

Course Calendar (approximate)

Monday	Tuesday	Thursday	Friday
Sep 4th	Sep 5th	Sep 7th	Sep 8th Introduction to graphics
Sep 11th	Sep 12th Introduction to OpenGL and shaders	Sep 14th	Sep 15th Introduction to OpenGL and shaders **Assignment 1 out**
Sep 18th	Sep 19th Shaders, 2D/3D modeling	Sep 21st	Sep 22nd Basic and Composite Transformations

Monday	Tuesday	Thursday	Friday
Sep 25th **Assignment 1 due**	Sep 26th Basic and Composite Transformations **Assignment 2 out**	Sep 28th	Sep 29th Transformations and Coordinate Systems
Oct 2nd	Oct 3rd Hierarchical Modeling	Oct 5th	Oct 6th Hierarchical Modeling **Assignment 2 due** **Assignment 3 out**
Oct 9th Columbus Day	Oct 10th Projections and Visibility	Oct 12th	Oct 13th Space partitioning
Oct 16th	Oct 17th Space partitioning **Assignment 3 due**	Oct 19th	Oct 20th Exam **Midterm Exam** **Assignment 4 out**
Oct 23rd	Oct 24th Lighting	Oct 26th	Oct 27th Lighting **Assignment 4 due** **Assignment 5 out**
Oct 30th	Oct 31st Lighting, Texture mapping	Nov 2nd	Nov 3rd Texture mapping
Nov 6th **Assignment 5 due**	Nov 7th Bump, environment mapping **Assignment 6+7 out**	Nov 9th	Nov 10th Ray tracing
Nov 13th	Nov 14th Ray tracing	Nov 16th	Nov 17th Ray tracing
Nov 20th	Nov 21st Curves and surfaces **Assignment 6 due**	Nov 23rd	Nov 24th Thanksgiving
Nov 27th	Nov 28th Curves and surfaces	Nov 30th	Dec 1st Particle systems **Assignment 7 due**

Monday	Tuesday	Thursday	Friday
Dec 4th	Dec 5th Particle systems/Review **Final exam goes out**	Dec 7th	Dec 8th
Dec 11th	Dec 12th **Final exam due**	Dec 14th	Dec 15th