CS334: Fundamentals of Computer Graphics
(or ECE30834)

Instructor: Daniel G. Aliaga (aliaga@cs.purdue.edu)
Classroom: ARMS B071
Time: Lecture MWF @ 11:30-12:20pm
Office hours: by appt
TAs: Chris May (may5@purdue.edu) and Yizhi Song (song630@purdue.edu)
TA office hours: TBD
Class communication: Piazza

1. Introduction

Interested in computer graphics, visual computing, virtual reality, architecture, games? Does modeling objects interest you? Do you like rendering photorealistic imagery? Is doing animations fun to you? Are you intrigued by visual computing, machine learning, and deep learning? All this is part of computer graphics. This course teaches the fundamentals, at an undergraduate school level, for such activities and research projects. Major applications include:

- Virtual and Augmented Reality
- Video Games
- Film Special Effects
- 3D Rendering, Modeling, and Shading
- GPU Programming
- Colors and Perception
- Geometry
- Optimization
- Machine Learning, Deep Learning
- Image Processing

2. Prerequisites

Students are required to have previous C/C++ programming experience and are recommended to have previous computer graphics experience, such as OpenGL programming experience although OpenGL will be reviewed at the beginning of the semester).

3. Books

There is no mandatory book but here are some recommendations.

- Computer Graphics: Principles and Practice, 3rd Edition by Hughes, van Dam, McGuire, Sklar, Foley, Feiner, and Akeley: this is a good overview of fundamentals of graphics.
- OpenGL Programming Guide ("red book"): you can buy the latest version or an old version free at www.glprogramming.com/red, for example.
- OpenGL Reference Manual ("blue book"): you can buy the latest version or an old version free at www.glprogramming.com/blue, for example.

There are plenty of other good books and I will bring some to class so you can peek at them.

4. Course work

The course work is composed of programming assignments, two exams, and interactive class participation. The programming assignments consist of a warm-up assignment, four minor programming assignments, and a final project. The exams consist of a midterm and a final exam. The final project consists of a fast-forward presentation at beginning of final project time and a final project presentation at end of semester. Course work will be easier to manage if you keep a constant pace through the semester. This course is hard work but you will learn a lot and have fun!

5. Grading

Programming Assignments:	30% (assignments 0-4: 1%, 5%, 8%, 8%, 8%)
	25% (5% fast forward, 20% final presentation)
Exams:	20% (midterm)
	20% (final)
Participation:	5%

	100% TOTAL

6. Assignments

Assignments must be written in C/C++ -- the final project programming language may be selected by the student/students. Assignments are due before class time on the due date and must be sent to the TAs using BrightSpace including all source code, data files, and *an already compiled program*. The time-stamp will be used to verify on time submission. The grading for the assignment will consider *functionality* and *form*. All assignments must be polished products, with a well-designed user interface and clean, reliable functionality. A program that does not compile obtains 0 points.

Assignment #0 – Cook it! (1.25 weeks). The objective is to cook up your first compiling, though simple, graphics project. You can either use (i) OpenGL+FreeGLUT or (ii) OpenGL+Qt to implement a trivial program. You will be given a skeleton framework.

Assignment #1 – Project It! (1 week): The objective is to ensure you understand well camera models and perspective projection. The logic will be described in class and in the assignment.

Assignment #2 – GPU It! (2.5 weeks): You will write a GPU program. We will provide a skeleton program but you will have to use OpenGL shading language (GLSL) and CPU code.

Assignment #3 –Map-It! (2.5 weeks). The object is to write a graphics program making using of some mapping concepts as explained in class. This project can extend the previous one.

Assignment #4 – Procedural-It! (2.5 weeks). This objective is a short procedural modeling program.

Final Project (4 weeks). Projects will be presented on a publicly attended "demo day" at the end of the semester (last day of classes, details TBD based on enrollment). You may choose a project that builds upon suggested ones or you may provide a written proposal for an independent project. Team projects (of up to 2 students) are permitted. Grading: the final assignment must be a polished product, with a well-designed user interface and clean, reliable functionality.

7. Fast-Forward Presentation

Near the beginning of the final project time period, each project (individual or group), will give a short GRADED "fast forward" presentation about a background literature search of their proposed project. The presentation should include mostly a summary of the state of the art and a short preview of what your project will do. Each fast forward will be few-minute presentation (details TBD based on enrollment).

8. Exams

The midterm will cover material explained in class, stressing fundamentals. The final exam will cover material of the entire semester and will stress understanding of general interactive computer graphics and its fundamentals. Both are closed book and will require "understanding and imagination" rather than memorization of formulas.

9. COVID-19 Policy (see https://protect.purdue.edu)

We realize this is a difficult and unique time due to the pandemic, however we are making all efforts to keep some level of normalcy. Yet, all students are REQUIRED to comply with Purdue-stipulated personal-protection-equipment (i.e., were masks at all times during class and in interaction with other students in the class as well as professor and TA) and to maintain social distancing whenever possible.

If you do have symptoms, or test-positive, please notify instructors ASAP, as well as Purdue, and follow Purdue guidelines – and DO NOT ATTEND class. We will devise on a case-by-case basis how to continue progress in the course.

9. Administrative Issues

EMAIL policy
Please start the subject line of ALL emails to professor, or TAs, with "CS334" – IF NOT THERE IS NO GUARANTEE your email will be read. Your professor receives lots of emails each day. For example, an email with a subject of "hello" will likely be buried or filtered-out.

Late policy
Assignments are due before class on due date. First time late – no penalty for up to one week, but instructors (professor and TAs) must be notified via email BEFORE deadline (*if instructor not notified via email before assignment due date, late pass cannot be used and assignment will be late*). Second and subsequent times -- grade reduction of 33% per day. All assignments required by demo day at end of course.

Collaboration
All assignments, exams, and review presentations must be done individually. Final projects may be done in teams upon approval by the instructor. Copying or plagiarism will give you a failing grade in the course and you will be subject to departmental and University policies. Code obtained from the Internet, books, or other sources may *not* be used for any assignment/project. Exceptions allowed only under explicit instructor approval.

10. Tentative Lecture Schedule (most current one is on webpage)

(1) Jan 10	State of Graphics, History, Organization, Final Projects
Jan 12	Math Review: geometric operations, vector math, matrix transformations I
Jan 14	Math Review: geometric operations, vector math, matrix transformations II **Assignment #0 out: Cook-it! (1.25 weeks)**
(2) Jan 17	**No classes**
Jan 19	Graphics Pipeline: Rasterization, Shading, and Lighting
Jan 21	Cameras and Projections I
(3) Jan 24	Cameras and Projections II **Assignment #0 due** **Assignment #1 out: Project-it! (2 weeks)**
Jan 26	Ray Tracing: Reflections, Refraction, and more I
Jan 28	Ray Tracing: Reflections, Refraction, and more II
(4) Jan 31	GPU Programming I
Feb 2	GPU Programming II
Feb 4	GPU Programming III
(5) Feb 7	**Assignment #1 due** **Assignment #2 out: GPU-it! (2 weeks)** Texture Mapping: projection, environment, bump, and shadow mapping I
Feb 9	Texture Mapping: projection, environment, bump, and shadow mapping II
Feb 11	Texture Mapping: projection, environment, bump, and shadow mapping III
(6) Feb 14	Data Structures and Spatial Hierarchies I
Feb 16	Data Structures and Spatial Hierarchies II
Feb 18	TBD
(7) Feb 21	**Assignment #2 due** **Assignment #3 out: Sketch-it! (2.5 weeks)** Procedural Modeling I
Feb 23	Procedural Modeling II
Feb 25	Procedural Modeling III
(8) Feb 28	**Review**
Mar 2	**Midterm**
Mar 4	**Solutions**

(9) Mar 7	Deep Visual Computing I
Mar 9	Deep Visual Computing II **Assignment #3 due** **Assignment #4 out: Procedural-it! (2.5 weeks)**
Mar 11	Deep Visual Computing III
(10) Mar 14	Spring Break!
Mar 16	Spring Break!
Mar 18	Spring Break!
(11) Mar 21	Voronoi Diagrams, Triangulation
Mar 23	Surface Triangulation
Mar 25	TBD
(12) Mar 28	**Assignment #4 due** **Final Project out (4 weeks)** Colors and Perception
Mar 30	Colors and Perception
Apr 1	TBD
(13) Apr 4	Final Project Fast Forward I
Apr 6	Final Project Fast Forward II
Apr 8	Final Project Fast Forward III
(14) Apr 11	Global Illumination I
Apr 13	Global Illumination II
Apr 15	TBD
(15) Apr 18	Image processing: morphing and warping I
Apr 20	Image processing: morphing and warping II
Apr 22	Review
(16) Apr 25	**Final Project Demo Prep**
Apr 27	**Final Project Demo Day I / TBA**
Apr 29	**Final Project Demo Day II / TBA**
Final Exam Wk	**Final Exam**