CS354 - Computer Graphics (Spring 2015)

Course:	Computer Graphics CS 354 Unique Number: 51943
Instructor:	Prof. Donald S. Fussell GDC 5.510 Phone: 471-9719 *fussell@cs.utexas.edu* Office Hours: MWF 10:00-11:00am
Teaching Assistant:	Doug Ilijev *doug.ilijev@utexas.edu* Office Hours: TTh 11:00-12:30 Location: TBD
Class Meetings:	MWF 12:00-1:00pm, GSB 2.126
Class Information:	Webpage: www.cs.utexas.edu/users/fussell/courses/cs354 Piazza: https://piazza.com/utexas/spring2015/cs354/home
Recommended Textbook:	*Interactive Computer Graphics: A Top-Down Approach with WebGL, 7/E*, by Edward Angel and Dave Shreiner, ISBN-10: 0133574849 ISBN-13: 9780133574845.
Prerequisites:	The following courses with a grade of at least C-: Computer Science 429 or 429H; Computer Science 105 (Topic: Computer Programming: C++), 371P, 373, or 378 (Topic: Generic Programming in the STL); Mathematics 408D, 408M, or 427L; and Mathematics 340L, 341, or Statistics and Data Sciences 329C (or Statistics and Scientific Computation 329C).

Course Objectives

This is an introductory course on the major topics in the areas of image synthesis, interactive techniques, geometric modeling, and computer-based animation. The material covered includes

- OpenGL programming
- principles of operation of raster graphics systems
- sampling and antialiasing
- homogeneous coordinate transformation techniques
- parallel and central projection and perspective transformations
- hidden surface removal
- light and reflectance models for local and global illumination
- shading techniques
- ray tracing
- basic object modeling techniques

- visual perception and basic color theory
- hierarchical modeling.
- basic animation

Upon course completion, you should have mastered both the mathematical principles of these techniques and their implementation. Implementation of these techniques will be demonstrated through a series of programming assignments in C++, in many cases using OpenGL. Your mastery of the mathematical fundamentals will be exercised through written homework and exams.

Student Evaluation

Your performance in this class will be evaluated through homework assignments, in class quizzes, programming assignments, and two midterm exams. The weights of each of these components is as follows:

- Exams (2): 30% (15% each)
- Homework and quizzes: 10%
- Programming Assignments: 60%

There are no makeup exams unless arranged explicitly in advance with the instructor. This will only be possible for valid non-academic reasons.

Homework assignments will be due at the beginning of lecture on the due date. Programming assignments will be submitted electronically by 11:59 on the due date and will require you to use your departmental Linux account. Turnin time will be the time recorded by the turnin program you will use to turn in the assignments. You may not email your assignments to the teaching staff.

Project submissions will each include a code listing and a written project description, which will include a description of the software architecture of the system and a user's guide. Grading also generally involves interactive demonstrations of the programs by the students. Programs will be graded on correctness, readability, style, and documentation. Program development may be done anywhere, but the version of the assignments that is turned in and demonstrated must be able to be built and executed on the publix Linux machines on which software for this course is supported.

You have a "late account" of 3 days for the term which you can spend any way you choose for your programming assigments. For example, you can turn in three programs one day late each, or one program 3 days late, without penalty. Once you have exhausted your late account, no late assignments will be accepted.

Academic Misconduct Policy

You are free to discuss the course material with your classmates and are encouraged to form study groups for the exams. However, collaboration on homework or programming assignments is **not** permitted unless explicitly stated by the instructor. Helping a friend understand the intent of a homework or programming assignment specification is permitted. Students who work together too closely (e.g. design their solution together) should be aware that this is a form of cheating called COLLUSION and is subject to academic penalties. Penalties for academic misconduct include a failing grade in this course.

The homework, programs, and exams must be the work of students turning them in. University policy (see Dean of Students' policies on academic integrity) will be followed strictly.

Acts that exceed the bounds defined by the approved collaboration practices will be considered cheating. Such acts include:

- Copying solutions, code, or programs from someone else or giving someone else your solutions, code, or programs
- Participation in a discussion group that develops a solution that everyone copies

We urge everyone in the class to take appropriate measures for protecting one's work. You should protect your files, homework solution sheets, etc. as deemed reasonable.

Studying for tests together is permitted and encouraged. Please come talk to us if you are unsure about how to work together with your friend in a legal, helpful manner. Remember, it is always ok to "work together" with your professor or TA!

Your Responsibilities in This Class

- You are expected to show up on time for lectures.
- You are responsible for all material posted to the web site and sent to the email list. Ignorance of such material is no excuse.
- You are responsible for obtaining a departmental Linux account, and learning the Linux operating system sufficiently well to meet the demands of the course. You can find on-line Linux guides - a place to start is here.
- You are responsible for all material presented in lecture and in the reading assignments.
- Your performance in this class will be determined by you! It will require a strong dedication to learning the material and may require a substantial time commitment to complete the programming assignments.
- You are responsible for turning in your own work on all assignments. Unauthorized collusion is not allowed and constitutes a violation of the university's policies on academic integrity. See above guidelines for more information on what is or is not allowed.
- You are responsible for protecting your work from being copied by others.

Programming Infrastructure

The programming assignments will require use of the Department of Computer Sciences instructional computing resources. All CS students who meet the prerequisites for CS354 are eligible for a CS departmental lab account. We will be doing projects using OpenGL and C++. User interfaces will be developed using GLUT or FLTK to make user interface building easier and to make the inferfaces portable across platforms.

Software needed for your projects is installed on the public Linux machines in the GDC labs. If you have trouble running the software, please report it to either the instructor or the TA. You can feel free to download and install the support software we are using on your own machines and work there, but you will need to turn in programs that work on the public Linux machines. Instructions for using software for each programming project on the lab machines will be given out with the assignments.

Important links:

- How to turn in assignments
- CS Computing Accounts
- UTCS Public Computer Laboratories

- UTCS Computing FAQ

Course Material Outline
Course Schedule

Last modified: 01/22/15 by Don Fussell fussell@cs.utexas.edu